

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**  
**CÔNG NGHỆ THÔNG TIN I**  
**BỘ MÔN NHẬP MÔN CÔNG NGHỆ PHẦN MỀM**



**Báo cáo bài tập lớn**  
**Nhập môn Công nghệ phần mềm**

<b>Tên đề tài:</b>	<b>Phát triển một phần mềm quản lý việc gọi món trong một nhà hàng</b>
<b>Mã đề tài</b>	<b>: 16</b>
<b>Họ và tên sinh viên</b>	<b>: LÊ MINH PHÚC</b>
<b>Mã sinh viên</b>	<b>: B21DCCN098</b>

*Hà Nội – 2024*

## Nội dung đề tài:

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý việc gọi món trong một nhà hàng, được mô tả như sau:

- Nhà hàng có nhiều bàn (Mã bàn, tên, số lượng khách tối đa, mô tả). Nhiều bàn nhỏ có thể gộp lại thành một bàn lớn khi có yêu cầu từ đoàn khách có số lượng lớn.
- Mỗi bàn, có thể bị đặt nhiều lần khác nhau trong ngày, hoặc khác ngày.
- Mỗi khách hàng (Mã, tên, số ĐT, email, địa chỉ) có thể đặt bàn nhiều lần, mỗi lần có thể đặt nhiều bàn (trường hợp này sẽ bị gộp thành đặt 1 bàn)
- Nhà hàng có thể lên combo dạng kết hợp sẵn một số món ăn đủ cho 1 bữa ăn cho một người ăn. Khách hàng có thể gọi combo có sẵn như thế này.
- Khách hàng ở mỗi bàn có thể gọi nhiều món ăn (Mã, loại, tên, mô tả, giá hiện tại) hoặc combo. Mỗi món ăn (combo) có thể bị gọi với số lượng khác nhau.
- Khi thanh toán, hóa đơn ghi đầy đủ thông tin: mã bàn, tên và mã nhân viên thanh toán, tên khách hàng nếu có, sau đó là một bảng, mỗi dòng chứa thông tin một món (combo) đã dùng: id, tên, đơn giá, số lượng, thành tiền. Dòng cuối cùng ghi tổng số tiền của hóa đơn.

Anh/chị hãy thực hiện modul "Đặt bàn" với các bước sau đây: Nhân viên chọn chức năng đặt bàn khi khách hàng gọi đến → giao diện tìm bàn trống hiện ra → NV nhập ngày + giờ đặt + số lượng khách và bấm tìm → kết quả hiện ra gồm danh sách các bàn còn trống vào ngày giờ đấy: mã, tên, số lượng khách tối đa, mô tả → NV chọn 1 bàn theo yêu cầu của KH → Giao diện nhập thông tin KH hiện ra → NV hỏi khách hàng và nhập mã, tên, số ĐT, email, địa chỉ và click tìm → Hệ thống hiện danh sách các khách hàng có cùng tên vừa nhập, mỗi khách hàng trên 1 dòng: mã, tên, số ĐT, email, địa chỉ → NV click vào dòng đúng với KH đag đặt (nếu không có thì lick thêm KH mới) → Hệ thống hiện lên giao diện xác nhận có đầy đủ thông tin bàn + thông tin KH + ngày giờ đặt → NV xác nhận với KH và click xác nhận → Hệ thống lưu thông tin vào CSDL.

## Mục lục

I. Pha lấy yêu cầu .....	3
1. Danh sách từ chuyên môn trong lĩnh vực của ứng dụng (glossary) .....	3
2. Mô tả thông tin chi tiết bằng ngôn ngữ tự nhiên: .....	3
3. Tác nhân trong hệ thống.....	4
4. Sơ đồ use case tổng quát của toàn hệ thống.....	5
5. Sơ đồ use case chi tiết module “đặt bàn” .....	6

II.	Pha phân tích .....	6
1.	Scenario và ngoại lệ.....	6
2.	Trích lớp thực thể.....	8
3.	Trích lớp điều khiển.....	9
4.	Trích lớp biên.....	9
5.	Mô hình hóa các lớp.....	10
6.	Mô hình hoạt động.....	10
7.	Scenario cuối pha phân tích .....	11
8.	Sơ đồ tuần tự.....	14
III.	Pha thiết kế.....	15
1.	Sơ đồ lớp thực thể pha thiết kế của hệ thống.....	15
2.	Thiết kế Cơ sở dữ liệu.....	16
3.	Thiết kế mô hình MVC cho module với thực thể thuần. ....	16
3.1.	Định nghĩa các thuộc tính và kiểu thuộc tính của mỗi lớp. ....	16
4.	Sơ đồ tuần tự.....	18
IV.	Pha cài đặt.....	19

## I. Pha lấy yêu cầu

### 1. Danh sách từ chuyên môn trong lĩnh vực của ứng dụng (glossary)

Từ ngữ	Giải nghĩa
Bàn	Nơi khách hàng ngồi ăn và gọi món.
Nhân viên	Là người hay nhóm người phục vụ khách hàng giúp khách hàng gọi món và thanh toán
Quản lí	Là người có chức năng quản lí các nhân viên, món ăn trong nhà hàng và có khả năng xem thống kê doanh thu hay món ăn bán chạy.
Khách hàng	Là người hay nhóm người có khả năng đặt bàn, gọi và sử dụng món ăn trong nhà hàng
Món ăn	Là sản phẩm được khách hàng tiêu dùng.
Combo	Là kết hợp sẵn một số món ăn đủ cho 1 bữa ăn cho một người ăn.

### 2. Mô tả thông tin chi tiết bằng ngôn ngữ tự nhiên:

Mục đích của hệ thống: Hệ thống ứng dụng giúp nhà hàng việc quản lí các món ăn, phục vụ cho việc đặt bàn và gọi món của khách hàng.

Nhà hàng có nhiều bàn (Mã bàn, tên, số lượng khách tối đa, mô tả). Nhiều bàn nhỏ có thể gộp lại thành một bàn lớn khi có yêu cầu từ đoàn khách có số lượng lớn.

- Mỗi bàn, có thể bị đặt nhiều lần khác nhau trong ngày, hoặc khác ngày.
- Mỗi khách hàng (Mã, tên, số ĐT, email, địa chỉ) có thể đặt bàn nhiều lần, mỗi lần có thể đặt nhiều bàn (trường hợp này sẽ bị gộp thành đặt 1 bàn)
- Nhà hàng có thể lên combo dạng kết hợp sẵn một số món ăn đủ cho 1 bữa ăn cho một người ăn. Khách hàng có thể gọi combo có sẵn như thế này.
- Khách hàng ở mỗi bàn có thể gọi nhiều món ăn (Mã, loại, tên, mô tả, giá hiện tại) hoặc combo. Mỗi món ăn (combo) có thể bị gọi với số lượng khác nhau.
- Khi thanh toán, hóa đơn ghi đầy đủ thông tin: mã bàn, tên và mã nhân viên thanh toán, tên khách hàng nếu có, sau đó là một bảng, mỗi dòng chứa thông tin một món (combo) đã dùng: id, tên, đơn giá, số lượng, thành tiền. Dòng cuối cùng ghi tổng số tiền của hóa đơn.

### 3. Tác nhân trong hệ thống

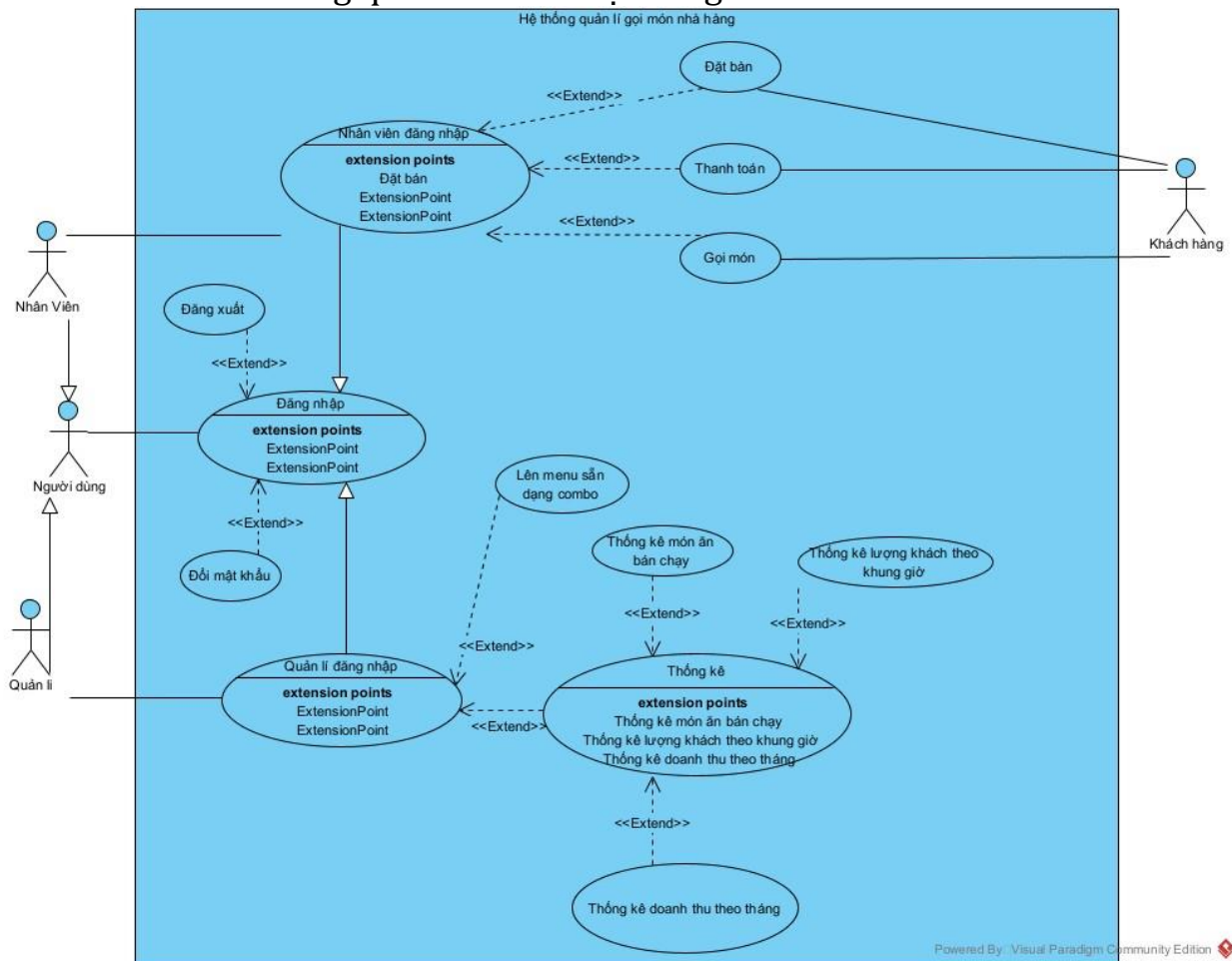
Nhân viên được phép:

- Đặt bàn khi khách hàng gọi đến
- Gọi món được khách hàng yêu cầu
- Thanh toán cho khách hàng

Quản lí được phép:

- Lên menu sẵn dạng combo, sửa, xóa, thêm thông tin cho combo
- Thống kê lượng khách theo khung giờ
- Thống kê món ăn bán chạy
- Thống kê doanh thu theo tháng

#### 4. Sơ đồ use case tổng quát của toàn hệ thống

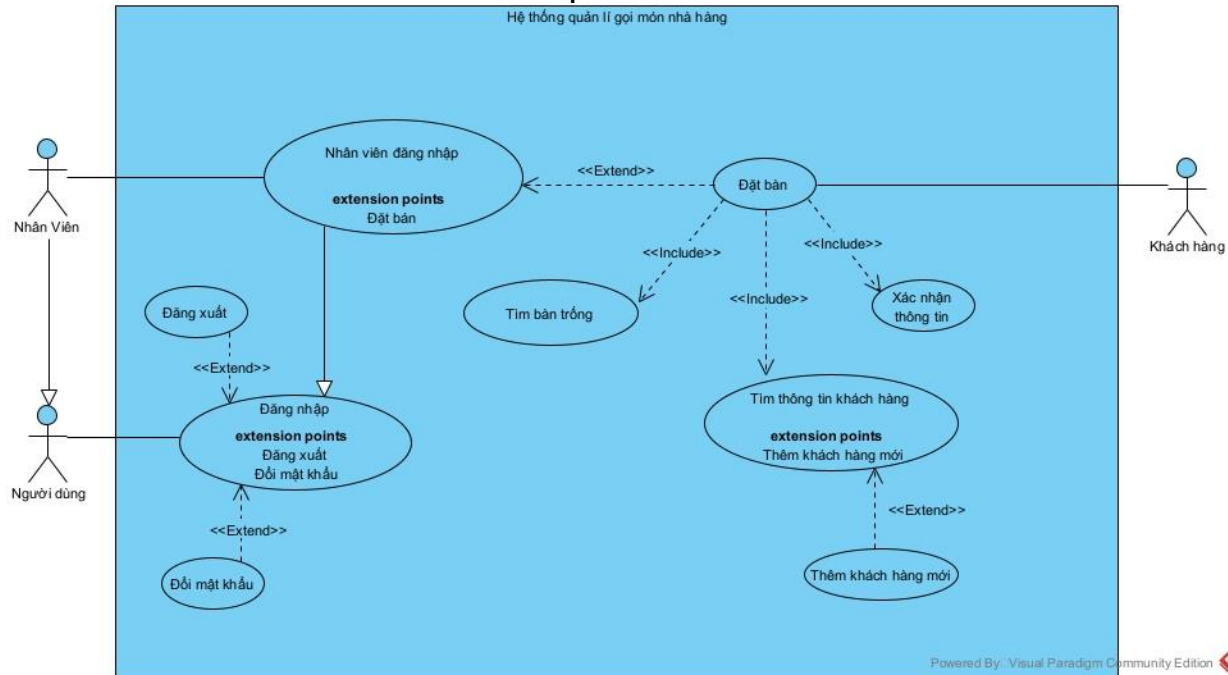


Chi tiết các usecase:

- Usecase Đăng nhập: giúp người đăng nhập vào ứng dụng.
- Usecase Đăng xuất: đăng xuất khỏi ứng dụng.
- Usecase Đổi mật khẩu: đổi mật khẩu của tài khoản.
- Usecase Nhân Viên đăng nhập: đăng nhập dưới quyền nhân viên có các chức năng sử dụng cho nhân viên.
- Usecase Quản lí đăng nhập: đăng nhập dưới quyền của quản lí.
- Usecase Đặt bàn: đặt bàn cho khách hàng.
- Usecase Thanh Toán: giúp khách hàng thanh toán sau khi ăn.
- Usecase Gọi món: giúp khách hàng gọi món.
- Usecase Thống kê: giúp quản lí xem các thống kê.
- Usecase Thông kê món ăn bán chạy: giúp quản lí xem thống kê các món ăn bán chạy.
- Usecase Thống kê doanh thu theo tháng: giúp quản lí xem doanh thu theo tháng.

- Usecase Thống kê lượng khách theo khung giờ: giúp quản lý xem lượng khách theo khung giờ.

## 5. Sơ đồ use case chi tiết module “đặt bàn”



Usecase Tìm bàn trống: tìm và chọn các bàn để đặt cho khách hàng.

Usecase Tìm thông tin khách hàng: tìm thông tin khách hàng có trong cơ sở dữ liệu

Usecase Xác nhận thông tin: xác nhận lại thông tin bàn và khách hàng đặt bàn

Usecase Thêm khách hàng mới: thêm khách hàng mới vào cơ sở dữ liệu

## II. Pha phân tích

### 1. Scenario và ngoại lệ

Usecases	Đặt bàn
Actor	Nhân viên, khách hàng
Tiên điều kiện	Nhân viên đăng nhập thành công và có khách hàng đặt bàn gọi đến
Hậu điều kiện	Nhân viên đặt bàn xong cho khách hàng.
Kịch bản chính	<ol style="list-style-type: none"> <li>1. Sau khi đăng nhập, từ giao diện chính nhân viên chọn chức năng đặt bàn.</li> <li>2. Giao diện tìm bàn trống hiện ra bao gồm các ô trống để nhập ngày, giờ, số lượng khách và một nút bấm tìm kiếm.</li> <li>3. Nhân viên hỏi khách hàng ngày, giờ và số lượng khách hàng.</li> <li>4. Khách hàng nói với nhân viên ngày, giờ và số lượng khách hàng.</li> <li>5. Nhân viên nhập ngày, giờ, số lượng khách hàng và bấm nút tìm kiếm.</li> <li>6. Màn hình hiện ra kết quả danh sách bàn còn trống vào ngày,</li> </ol>

giờ đã chọn. Mỗi hàng tương ứng một dòng bao gồm các thông tin: mã, tên, số lượng khách hàng tối đa, mô tả.

Mã	Tên	Số lượng khách hàng tối đa	Mô tả
B01	Bàn một	8	Bàn tối đa tám người

7. Nhân viên cho khách hàng xem danh sách phòng trống.
8. Khách hàng chọn một bàn trong danh sách phòng trống.
9. Nhân viên bấm chọn bàn khách hàng yêu cầu.
10. Giao diện nhập thông tin khách hàng hiện ra gồm ô nhập mã, tên, số điện thoại, email, địa chỉ và một nút tìm kiếm.
11. Nhân viên hỏi khách hàng tên, số điện thoại, email, địa chỉ khách hàng.
12. Khách hàng nói cho nhân viên tên, số điện thoại, email, địa chỉ của mình.
13. Nhân viên nhập tên khách hàng và bấm nút tìm kiếm.
14. Màn hình hiện ra danh sách các khách hàng có cùng tên vừa nhập bao gồm các trường mã, tên, số điện thoại, email, địa chỉ và nút thêm khách hàng mới.

	Mã	Tên	Số điện thoại	Email	Địa chỉ
	KH01	Nguyễn Văn A	0123456789	<a href="mailto:A@mail.com">A@mail.com</a>	Hà Nội
	KH02	Nguyễn Văn A	0398746546	<a href="mailto:B@mail.com">B@mail.com</a>	Huế

	Thêm khách hàng mới	
--	---------------------	--

15. Nhân viên click vào một dòng đúng với tên, số điện thoại, email, địa chỉ khách hàng đang đặt.
16. Giao diện xác nhận hiện ra có đầy đủ thông tin bàn, thông tin khách hàng và ngày giờ đặt cùng với nút xác nhận.
17. Nhân viên xác nhận với khách hàng.
18. Khách hàng xác nhận lại thông tin.
19. Nhân viên click nút xác nhận.
20. Một cửa sổ thông báo đặt bàn thành công hiện ra và quay về trang chủ.

Ngoại lệ
----------

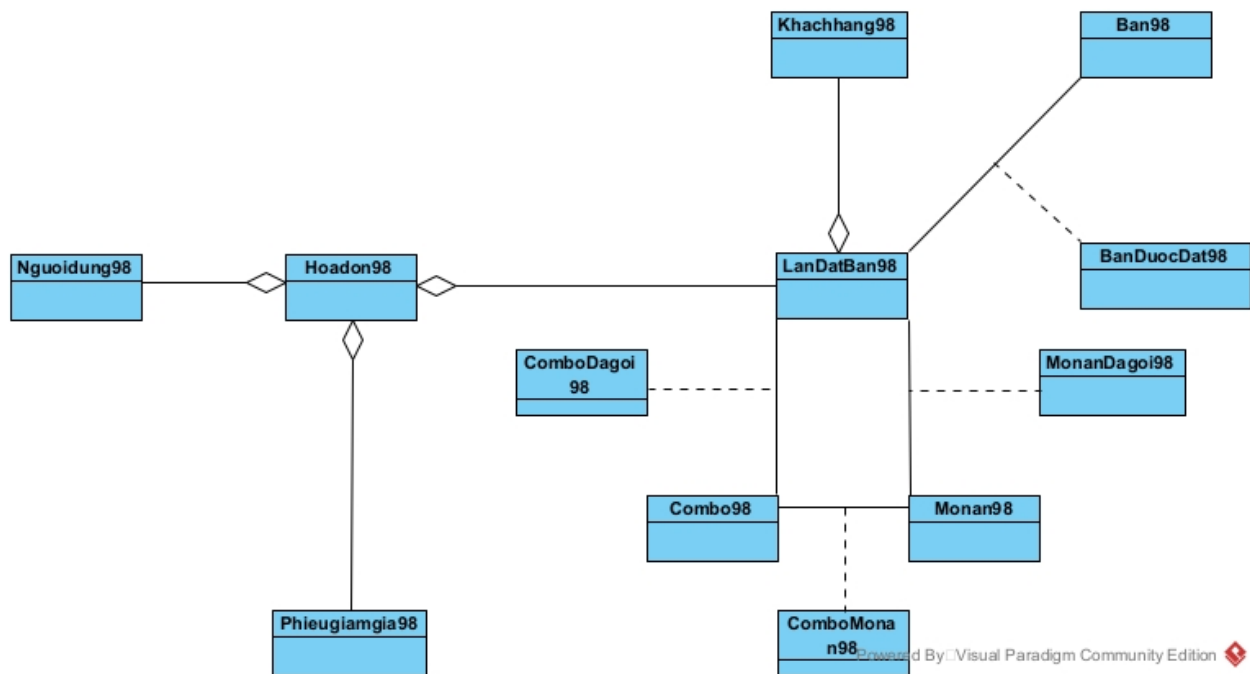
- |          |   |
|----------|---|
| Ngoại lệ | <p>6. Không có bàn nào trống vào ngày, giờ đã chọn nên danh sách hiện ra trống</p> <p>6.1. Nhân viên thông báo cho khách hàng đã hết bàn trống và yêu cầu khách hàng đặt bàn vào khung giờ khác</p> <p>6.2. Khách hàng nói cho nhân viên ngày, giờ khác</p> |
|----------|---|

	6.3. Quay lại bước 5 trong kịch bản chính 14. Không có dòng nào chứa thông tin tên đúng với khách hàng đang đặt nên danh sách hiện ra trống 14.1. Nhân viên click nút thêm khách hàng mới. 14.2. Màn hình hiện ra các ô trống để nhập tên khách hàng, email, số điện thoại, địa chỉ. 14.3. Nhân viên nhập tên khách hàng, email, số điện thoại, địa chỉ và click nút thêm 14.4. Tiếp tục bước 16 trong kịch bản chính
--	--

## 2. Trích lớp thực thể.

- Các danh từ: Nhà hàng, bàn, đoàn khách, lần, ngày, khách hàng, combo, món ăn, bữa ăn, người ăn, hóa đơn, nhân viên thanh toán, nhân viên, quản lí.
- Đánh giá lựa chọn các danh từ làm thực thể:
  - Các danh từ trừu tượng hoặc không nằm trong phạm vi hệ thống: Nhà hàng, bữa ăn, ngày, lần.
  - Các danh từ liên quan đến người: Khách hàng, nhân viên, quản lí.
  - Các danh từ liên quan đến thông tin: Combo, món ăn, bàn, hóa đơn, phiếu giảm giá.
- Lựa chọn các danh từ làm lớp thực thể:
  - Người dùng hệ thống → lớp Nguoidung98.
  - Nhân viên → lớp Nhanvien kế thừa lớp Nguoidung98.
  - Quản lí → lớp Quanli kế thừa lớp Nguoidung98.
  - Khách hàng → lớp Khachhang98.
  - Món ăn → lớp Monan98.
  - Combo → lớp Combo98.
  - Bàn → lớp Ban98.
  - Hóa đơn → lớp Hoadon98.
  - Phiếu giảm giá → lớp Phieugiamgia98.
- Quan hệ giữa các lớp thực thể:
  - Một khách hàng có thể đặt nhiều bàn một lần, một bàn có thể đặt bởi nhiều khách hàng tại các thời điểm khác nhau → Đề xuất lớp LanDatBan98
  - Một lần đặt bàn có thể đặt được nhiều bàn. Một bàn có thể đặt nhiều lần tại các thời điểm khác nhau → đề xuất thêm lớp BanDuocDat98 phụ thuộc vào lớp LanDatBan98 và Ban98.
  - Một món ăn có thể nằm trong nhiều combo, một combo có thể có nhiều món ăn → Đề xuất thêm lớp ComboMonan98.
  - Một nhân viên có thể thanh toán nhiều hóa đơn nhưng một hóa đơn chỉ thanh toán bởi một nhân viên.
  - Một hóa đơn chỉ bao gồm một phiếu giảm giá, một phiếu giảm giá chỉ nằm trong một hóa đơn
  - Một LanDatBan98 có thể bao gồm nhiều Monan98 hoặc Combo98 khác nhau, một món ăn Combo có thể được nằm trong nhiều LanDatban98 → đề xuất thêm các lớp MonanDagoi98 và ComboDagoi98.
  - Một Hoadon có một Datban, một đặt bàn nằm trong một hóa đơn.





### 3. Trích lớp điều khiển.

- Lớp điều khiển cho module đặt bàn: DieukhienDatban98

### 4. Trích lớp biên.

Sau khi nhân viên đăng nhập thành công giao diện chính xuất hiện để chọn chức năng Đặt bàn nên đề xuất lớp Giao diện chính: GDChinhNhanVien98.

Đề xuất các lớp biên cho module đặt bàn:

- Giao diện tìm kiếm bàn: GDTimBan98.
- Chức năng tìm kiếm khách hàng: GDTimKH98.
- Chức năng thêm khách hàng mới: GDThemKH98.
- Giao diện xác nhận thông tin: GDXacnhan98.
- Các dialog và cửa sổ con đều là thành phần của các form chính.

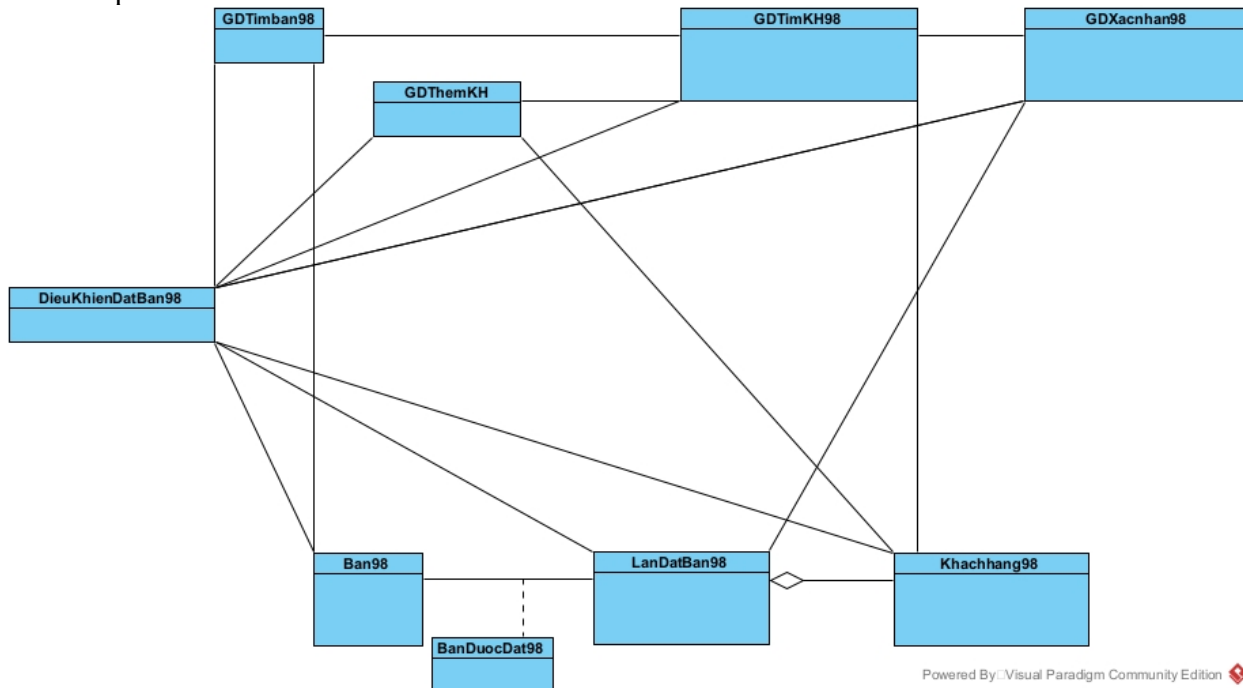
## 5. Mô hình hóa các lớp.

Thẻ CRC cho lớp điều khiển DieukhienDatBan:

DieukhienDatBan	
Responsibilities:	
Name	Collaborator
Yêu cầu lớp GDTimBan98 hiển thị giao diện tìm bán trống	GDTimBan98
Yêu cầu lớp GDTimBan98 hiển thị giao diện kết quả tìm kiếm phòng	GDTimBan98
Yêu cầu lớp GDTimKH98 hiển thị giao diện tìm khách hàng	GDTimKH98
Yêu cầu lớp GDTimKH98 hiển thị kết quả tìm kiếm khách hàng	GDTimKH98
Yêu cầu lớp GDxacNhan98 hiển thị giao diện xác nhận thông tin đặt	GDxacnhan98
Yêu cầu lớp GDThemKH98 hiển thị giao diện thêm khách hàng mới	GDThemKH98
Tìm kiếm bán trống trong cơ sở dữ liệu theo ngày, giờ	
Tìm kiếm khách hàng trong cơ sở dữ liệu theo tên khách hàng	
Thêm khách hàng mới vào cơ sở dữ liệu	
Cập nhật thông tin đặt bán vào cơ sở dữ liệu	
Yêu cầu lớp Khachhang đóng gói đối tượng Khachhang98	Khachhang98
Yêu cầu lớp LanDatban đóng gói đối tượng LanDatban98	LanDatBan98
Yêu cầu lớp Ban đóng gói đối tượng Ban98	Ban98

Powered By Visual Paradigm Community Edition

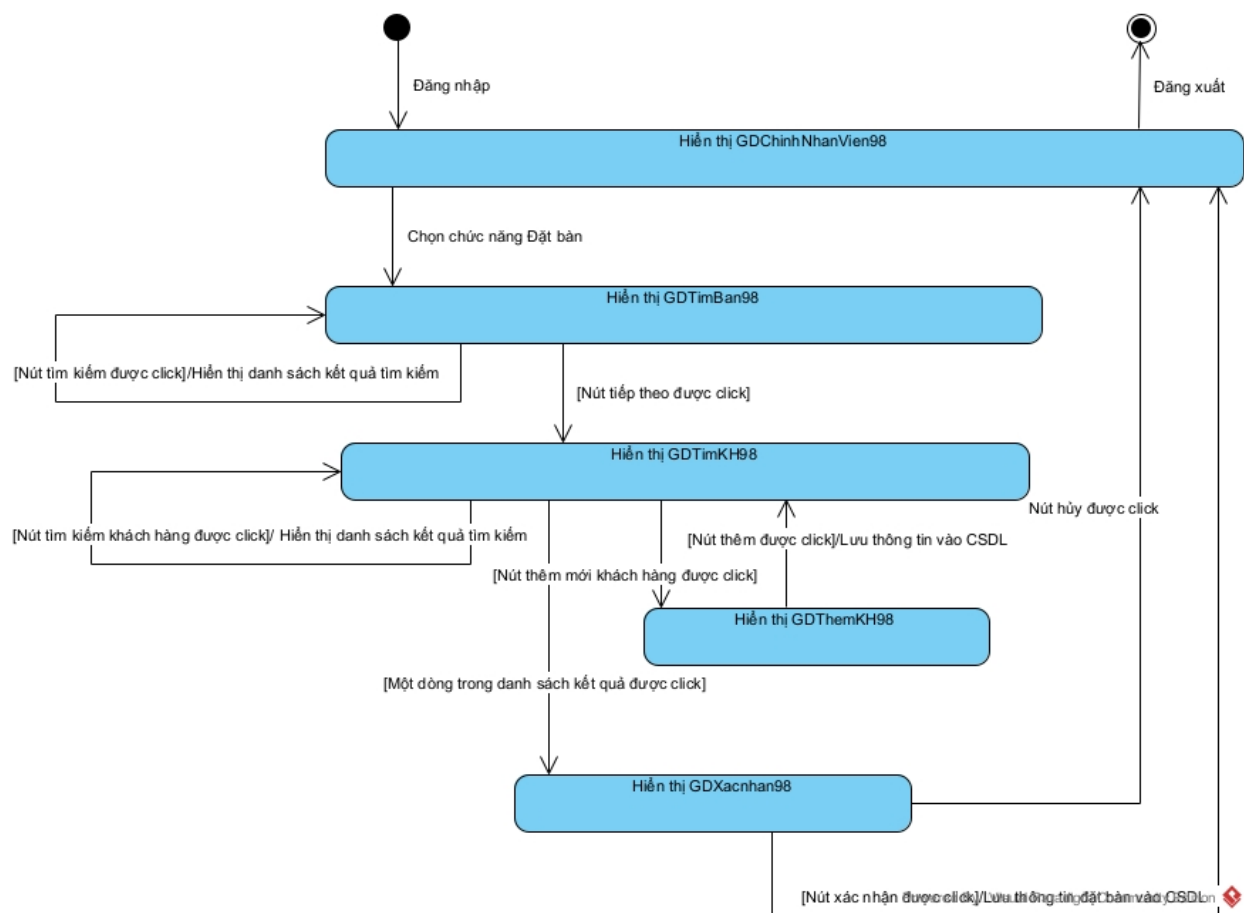
Sơ đồ lớp cho module Datban:



Powered By Visual Paradigm Community Edition

## 6. Mô hình hoạt động.

Sơ đồ statechart cho module đặt bàn:



## 7. Scencario cuối pha phân tích

Usecases	Đặt bàn
Actor	Nhân viên, khách hàng
Tiên điều kiện	Nhân viên đăng nhập thành công và có khách hàng đặt bàn gọi đến
Hậu điều kiện	Nhân viên đặt bàn xong cho khách hàng.
Kịch bản chính	<ol style="list-style-type: none"> <li>1. Sau khi đăng nhập, từ GDChinhNhanVien98 nhân viên chọn chức năng đặt bàn.</li> <li>2. GDChinhNhanVien98 yêu cầu hiển thị lớp GDTimBan98.</li> <li>3. Lớp GDTimBan98 hiện ra bao gồm các ô trống để nhập ngày, giờ, số lượng khách và một nút bấm tìm kiếm.</li> <li>4. Nhân viên hỏi khách hàng ngày, giờ và số lượng khách hàng.</li> <li>5. Khách hàng nói với nhân viên ngày, giờ và số lượng khách hàng.</li> <li>6. Nhân viên nhập ngày, giờ, số lượng khách hàng và bấm nút tìm kiếm.</li> <li>7. Lớp GDTimBan98 gửi thông tin ngày, giờ, số lượng khách hàng cho lớp DieuKienDatban98</li> </ol>

8. Lớp DieuKienDatban98 tìm kiếm các bàn trống vào ngày, khung giờ và số lượng khách tối đa lớn hơn số lượng khách đã nhập trong cơ sở dữ liệu
9. Lớp DieuKienDatban98 gửi kết quả cho lớp Ban98 đóng gói thành danh sách các đối tượng Ban98
10. Lớp DieuKienDatban98 gửi danh sách các đối tượng Datban98 cho lớp GDTimBan98 để hiển thị
11. GDTimBan98 hiện ra kết quả danh sách bàn còn trống vào ngày, giờ đã chọn. Mỗi hàng tương ứng một dòng bao gồm các thông tin: mã, tên, số lượng khách hàng tối đa, mô tả.

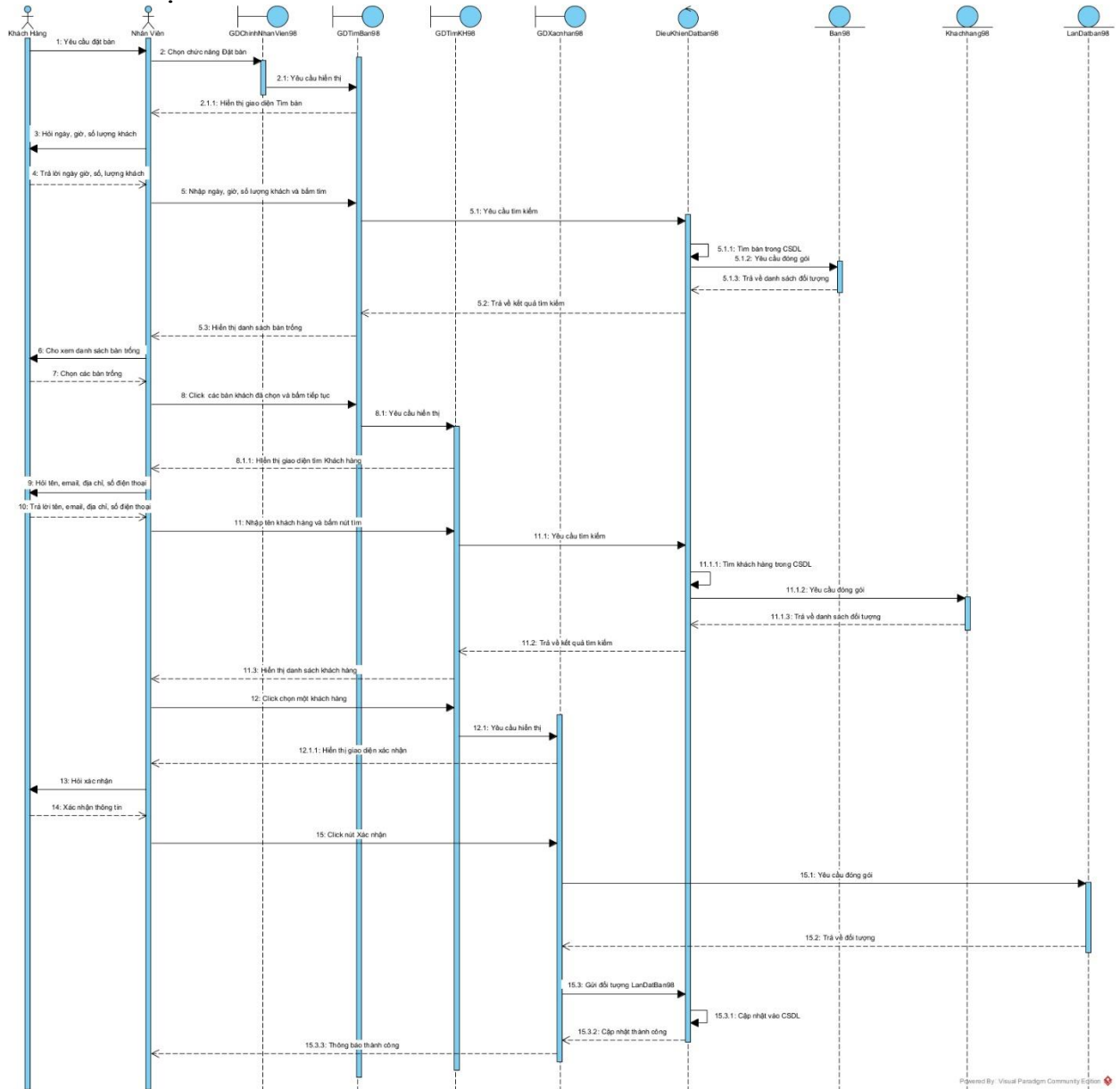
Mã	Tên	Số lượng khách hàng tối đa	Mô tả
B01	Bàn một	8	Bàn tối đa tám người

12. Nhân viên cho khách hàng xem danh sách phòng trống.
13. Khách hàng chọn một bàn trong danh sách phòng trống.
14. Nhân viên click chọn bàn khách hàng yêu cầu và click nút tiếp theo.
15. Lớp GDTimBan98 gọi lớp GDTimKH98 để hiển thị
16. lớp GDTimKH98 hiện ra gồm ô nhập mã, tên, số điện thoại, email, địa chỉ và một nút tìm kiếm.
17. Nhân viên hỏi khách hàng tên, số điện thoại, email, địa chỉ khách hàng.
18. Khách hàng nói cho nhân viên tên, số điện thoại, email, địa chỉ của mình.
19. Nhân viên nhập tên khách hàng và bấm nút tìm kiếm.
20. Lớp GDTimKH98 gửi thông tin tên khách hàng đến lớp DieuKienDatban98
21. Lớp DieuKienDatban98 tìm thông tin khách hàng có tên đã nhập trong cơ sở dữ liệu
22. Lớp DieuKienDatban98 gửi kết quả đến lớp Khachhang98 để đóng gói thành danh sách các đối tượng Khachhang98
23. Lớp Khachhang98 gửi trả cho lớp DieuKienDatban98 danh sách các đối tượng Khachhang98
24. Lớp DieuKienDatban98 gửi trả danh sách các đối tượng cho lớp GDTimKH98 để hiển thị
25. Lớp GDTimKH98 hiện ra danh sách các khách hàng có cùng tên vừa nhập bao gồm các trường mã, tên, số điện thoại, email, địa chỉ và nút thêm khách hàng mới.

Mã	Tên	Số điện thoại	Email	Địa chỉ
KH01	Nguyễn Văn A	0123456789	<a href="mailto:A@mail.com">A@mail.com</a>	Hà Nội

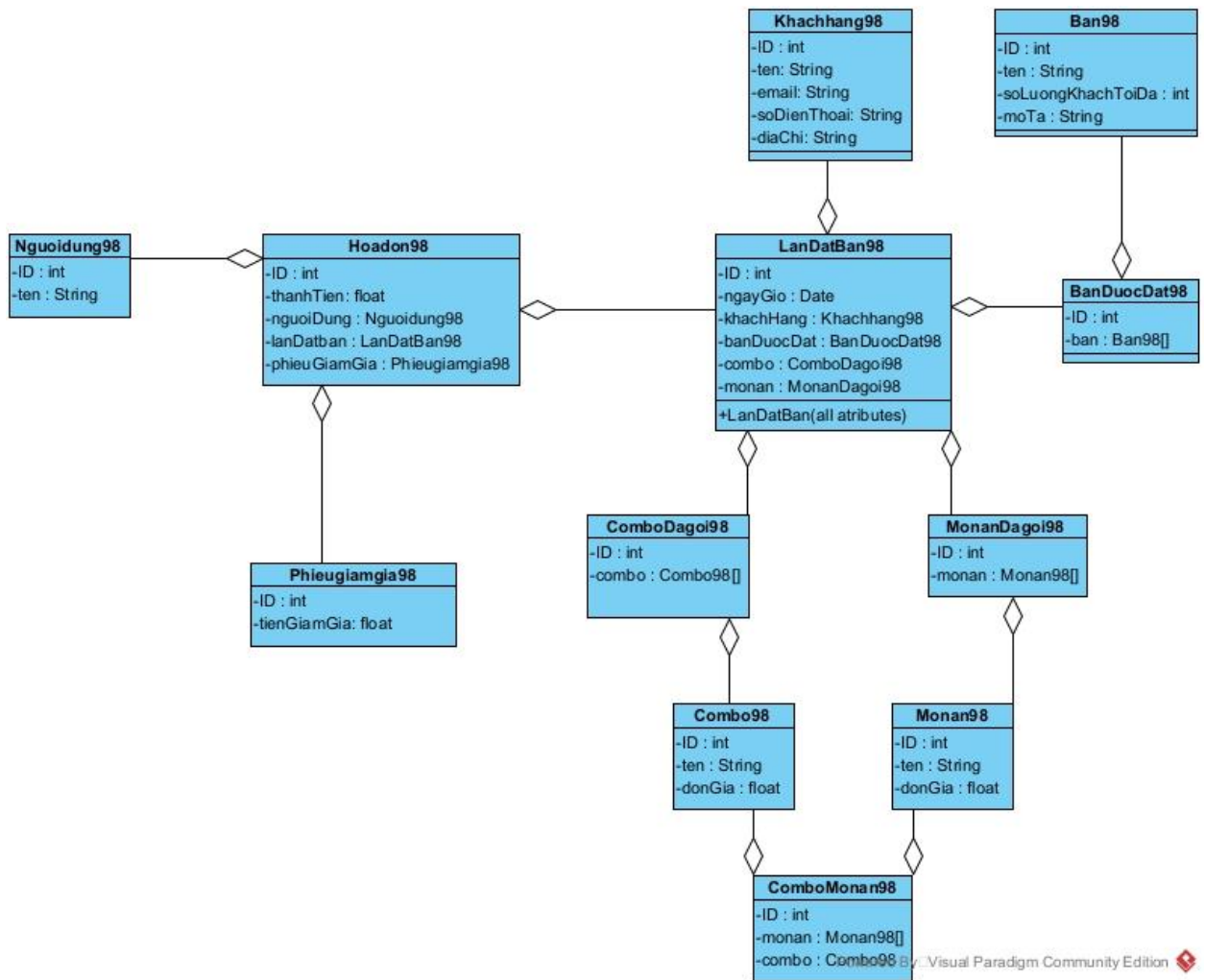
		KH02	Nguyễn Văn A	0398746546	<a href="mailto:B@mail.com">B@mail.com</a>	Huế
		Thêm khách hàng mới		<p>26. Nhân viên click vào một dòng đúng với tên, số điện thoại, email, địa chỉ khách hàng đang đặt.</p> <p>27. Lớp GDTimKH98 gửi yêu cầu hiển thị đến lớp GDxacnhan98</p> <p>28. Lớp GDxacnhan98 gửi thông tin lần đặt bàn bao gồm thông tin khách hàng và danh sách các bàn được đặt đến lớp LanDatBan98 để đóng gói thành đối tượng LanDatBan98.</p> <p>29. Lớp LanDatBan98 trả lại đối tượng LanDatBan98 cho lớp GDxacnhan98 để hiển thị.</p> <p>30. Lớp GDxacnhan98 hiện ra đầy đủ thông tin bàn, thông tin khách hàng và ngày giờ đặt cùng với nút xác nhận.</p> <p>31. Nhân viên xác nhận với khách hàng.</p> <p>32. Khách hàng xác nhận lại thông tin.</p> <p>33. Nhân viên click nút xác nhận.</p> <p>34. Lớp GDxacnhan98 gửi đối tượng LanDatBan98 cho lớp DieuKienDatBan98.</p> <p>35. Lớp DieuKienDatBan98 lưu thông tin vào cơ sở dữ liệu.</p> <p>36. Một cửa sổ thông báo đặt bàn thành công hiện ra.</p>		

## 8. Sơ đồ tuần tự.

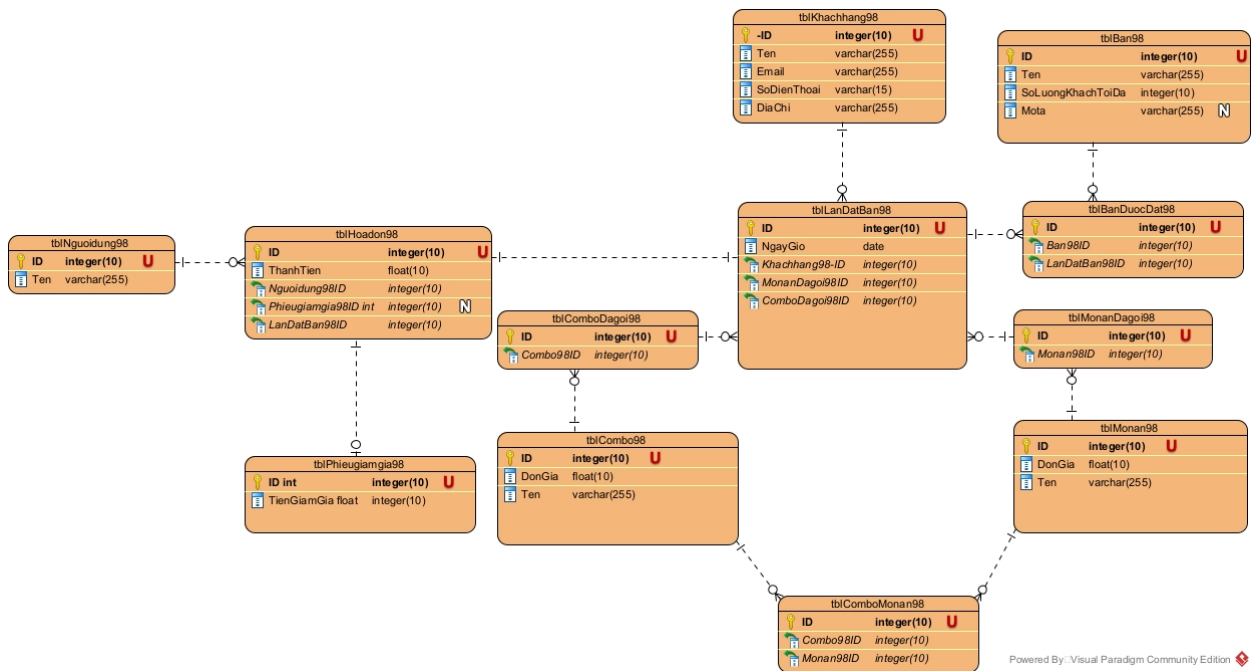


### III. Pha thiết kế.

#### 1. Sơ đồ lớp thực thể pha thiết kế của hệ thống

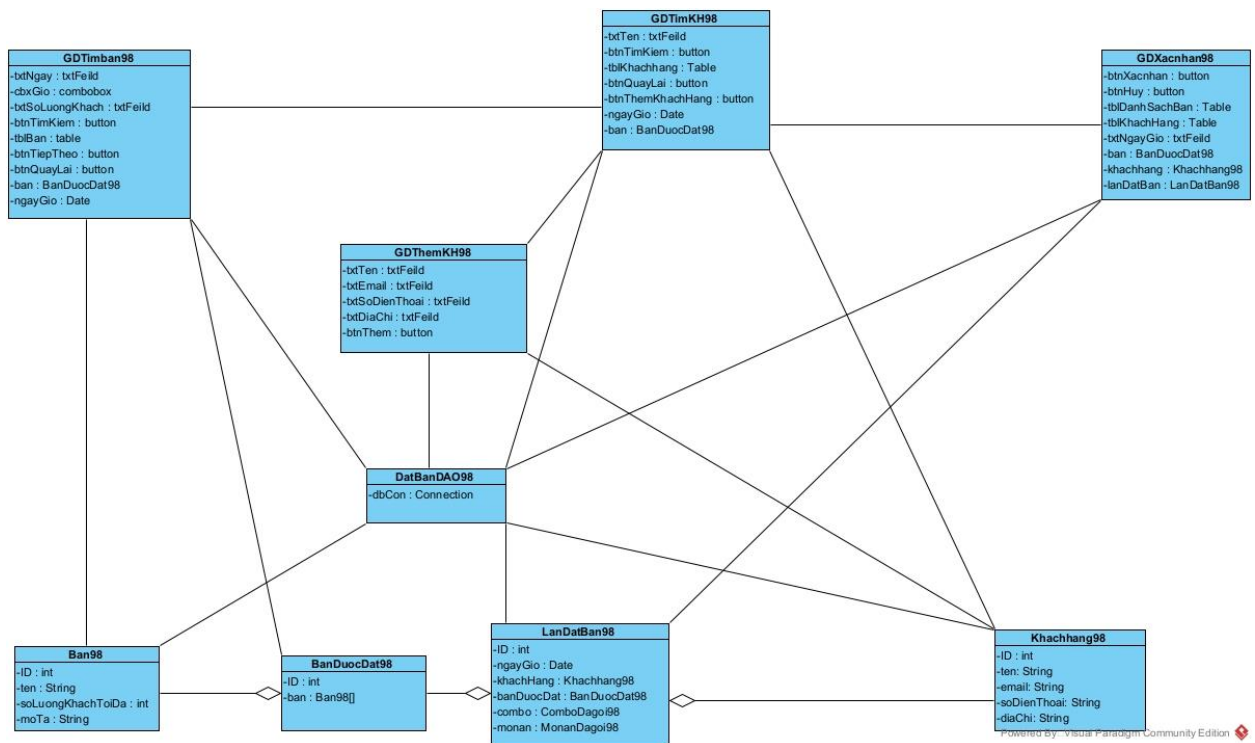


## 2. Thiết kế Cơ sở dữ liệu



## 3. Thiết kế mô hình MVC cho module với thực thể thuần.

### 3.1. Định nghĩa các thuộc tính và kiểu thuộc tính của mỗi lớp.





### 3.2. Thiết kế phương thức để gán các phương thức cho các lớp

#### 3.2.1. Thẻ CRC cho các lớp

<table> <tr><th colspan="2">DatBanDAO98</th></tr> <tr><th colspan="2">Responsibilities:</th></tr> <tr> <th>Name</th><th>Collaborator</th></tr> <tr><td>Yêu cầu lớp GDTimBan98 hiển thị giao diện tìm kiếm</td><td>GDTimBan98</td></tr> <tr><td>Tìm kiếm thông tin bàn trống trong CSDL</td><td></td></tr> <tr><td>Yêu cầu lớp Ban98 đóng gói thông tin tìm kiếm thành danh sách đối tượng</td><td>Ban98</td></tr> <tr><td>Yêu cầu lớp GDTimKH98 hiển thị giao diện tìm khách hàng</td><td>GDTimKH98</td></tr> <tr><td>Yêu cầu lớp Khachhang98 đóng gói thông tin tìm kiếm được thành danh sách đối tượng</td><td>Khachhang98</td></tr> <tr><td>Yêu cầu lớp GDThemKH98 hiển thị giao diện thêm mới khách hàng</td><td>GDThemKH98</td></tr> <tr><td>Thêm thông tin khách hàng mới vào CSDL</td><td></td></tr> <tr><td>Yêu cầu lớp GDThemKH98 thông báo thêm thành công</td><td>GDThemKH98</td></tr> <tr><td>Yêu cầu lớp GDXacnhan98 hiển thị giao diện xác nhận</td><td>GDXacnhan98</td></tr> <tr><td>Thêm thông tin đặt bàn vào CSDL</td><td></td></tr> <tr><td>Yêu cầu lớp GDXacnhan98 hiển thị thông báo đặt bàn thành công</td><td></td></tr> </table>	DatBanDAO98		Responsibilities:		Name	Collaborator	Yêu cầu lớp GDTimBan98 hiển thị giao diện tìm kiếm	GDTimBan98	Tìm kiếm thông tin bàn trống trong CSDL		Yêu cầu lớp Ban98 đóng gói thông tin tìm kiếm thành danh sách đối tượng	Ban98	Yêu cầu lớp GDTimKH98 hiển thị giao diện tìm khách hàng	GDTimKH98	Yêu cầu lớp Khachhang98 đóng gói thông tin tìm kiếm được thành danh sách đối tượng	Khachhang98	Yêu cầu lớp GDThemKH98 hiển thị giao diện thêm mới khách hàng	GDThemKH98	Thêm thông tin khách hàng mới vào CSDL		Yêu cầu lớp GDThemKH98 thông báo thêm thành công	GDThemKH98	Yêu cầu lớp GDXacnhan98 hiển thị giao diện xác nhận	GDXacnhan98	Thêm thông tin đặt bàn vào CSDL		Yêu cầu lớp GDXacnhan98 hiển thị thông báo đặt bàn thành công		<table> <tr><th colspan="2">GDTimBan98</th></tr> <tr><th colspan="2">Responsibilities:</th></tr> <tr> <th>Name</th><th>Collaborator</th></tr> <tr><td>Hiển thị giao diện form nhập thông tin tìm kiếm bàn trống</td><td></td></tr> <tr><td>Yêu cầu lớp DatBanDAO98 tìm kiếm thông tin bàn trống trong CSDL</td><td>DatBanDAO98</td></tr> <tr><td>Xử lý sự kiện khi nút Tìm Kiếm được click</td><td></td></tr> <tr><td>Xử lý sự kiện khi các dòng trong bảng kết quả được chọn</td><td></td></tr> <tr><td>Xử lý sự kiện khi nút tiếp theo được click</td><td></td></tr> <tr><td>Yêu cầu lớp BanDuocDat98 đóng gói đối tượng</td><td>BanDuocDat98</td></tr> </table> <table> <tr><th colspan="2">GDTimKH98</th></tr> <tr><th colspan="2">Responsibilities:</th></tr> <tr> <th>Name</th><th>Collaborator</th></tr> <tr><td>Hiển thị giao diện nhập thông tin tìm kiếm khách hàng</td><td></td></tr> <tr><td>Yêu cầu lớp DatBanDAO98 tìm kiếm thông tin khách hàng trong CSDL</td><td>DatBanDAO98</td></tr> <tr><td>Xử lý sự kiện khi nút Tìm Kiếm được click</td><td></td></tr> <tr><td>Xử lý sự kiện khi một dòng trong bảng khách hàng được click</td><td></td></tr> <tr><td>Xử lý sự kiện khi nút Thêm mới khách hàng được click</td><td></td></tr> </table>	GDTimBan98		Responsibilities:		Name	Collaborator	Hiển thị giao diện form nhập thông tin tìm kiếm bàn trống		Yêu cầu lớp DatBanDAO98 tìm kiếm thông tin bàn trống trong CSDL	DatBanDAO98	Xử lý sự kiện khi nút Tìm Kiếm được click		Xử lý sự kiện khi các dòng trong bảng kết quả được chọn		Xử lý sự kiện khi nút tiếp theo được click		Yêu cầu lớp BanDuocDat98 đóng gói đối tượng	BanDuocDat98	GDTimKH98		Responsibilities:		Name	Collaborator	Hiển thị giao diện nhập thông tin tìm kiếm khách hàng		Yêu cầu lớp DatBanDAO98 tìm kiếm thông tin khách hàng trong CSDL	DatBanDAO98	Xử lý sự kiện khi nút Tìm Kiếm được click		Xử lý sự kiện khi một dòng trong bảng khách hàng được click		Xử lý sự kiện khi nút Thêm mới khách hàng được click	
DatBanDAO98																																																															
Responsibilities:																																																															
Name	Collaborator																																																														
Yêu cầu lớp GDTimBan98 hiển thị giao diện tìm kiếm	GDTimBan98																																																														
Tìm kiếm thông tin bàn trống trong CSDL																																																															
Yêu cầu lớp Ban98 đóng gói thông tin tìm kiếm thành danh sách đối tượng	Ban98																																																														
Yêu cầu lớp GDTimKH98 hiển thị giao diện tìm khách hàng	GDTimKH98																																																														
Yêu cầu lớp Khachhang98 đóng gói thông tin tìm kiếm được thành danh sách đối tượng	Khachhang98																																																														
Yêu cầu lớp GDThemKH98 hiển thị giao diện thêm mới khách hàng	GDThemKH98																																																														
Thêm thông tin khách hàng mới vào CSDL																																																															
Yêu cầu lớp GDThemKH98 thông báo thêm thành công	GDThemKH98																																																														
Yêu cầu lớp GDXacnhan98 hiển thị giao diện xác nhận	GDXacnhan98																																																														
Thêm thông tin đặt bàn vào CSDL																																																															
Yêu cầu lớp GDXacnhan98 hiển thị thông báo đặt bàn thành công																																																															
GDTimBan98																																																															
Responsibilities:																																																															
Name	Collaborator																																																														
Hiển thị giao diện form nhập thông tin tìm kiếm bàn trống																																																															
Yêu cầu lớp DatBanDAO98 tìm kiếm thông tin bàn trống trong CSDL	DatBanDAO98																																																														
Xử lý sự kiện khi nút Tìm Kiếm được click																																																															
Xử lý sự kiện khi các dòng trong bảng kết quả được chọn																																																															
Xử lý sự kiện khi nút tiếp theo được click																																																															
Yêu cầu lớp BanDuocDat98 đóng gói đối tượng	BanDuocDat98																																																														
GDTimKH98																																																															
Responsibilities:																																																															
Name	Collaborator																																																														
Hiển thị giao diện nhập thông tin tìm kiếm khách hàng																																																															
Yêu cầu lớp DatBanDAO98 tìm kiếm thông tin khách hàng trong CSDL	DatBanDAO98																																																														
Xử lý sự kiện khi nút Tìm Kiếm được click																																																															
Xử lý sự kiện khi một dòng trong bảng khách hàng được click																																																															
Xử lý sự kiện khi nút Thêm mới khách hàng được click																																																															
<table> <tr><th colspan="2">GDXacnhan98</th></tr> <tr><th colspan="2">Responsibilities:</th></tr> <tr> <th>Name</th><th>Collaborator</th></tr> <tr><td>Hiển thị giao diện Xác nhận đặt bàn</td><td></td></tr> <tr><td>Yêu cầu lớp LanDatBan98 đóng gói thông tin đặt bàn thành đối tượng</td><td>LanDatBan98</td></tr> <tr><td>Yêu cầu lớp DatBanDAO98 lưu thông tin vào CSDL</td><td>DatBanDAO98</td></tr> </table>	GDXacnhan98		Responsibilities:		Name	Collaborator	Hiển thị giao diện Xác nhận đặt bàn		Yêu cầu lớp LanDatBan98 đóng gói thông tin đặt bàn thành đối tượng	LanDatBan98	Yêu cầu lớp DatBanDAO98 lưu thông tin vào CSDL	DatBanDAO98	<table> <tr><th colspan="2">GDThemKH98</th></tr> <tr><th colspan="2">Responsibilities:</th></tr> <tr> <th>Name</th><th>Collaborator</th></tr> <tr><td>Hiển thị giao diện nhập thông tin khách hàng</td><td></td></tr> <tr><td>Yêu cầu lớp Khachhang98 đóng gói thông tin đã nhập thành đối tượng</td><td>Khachhang98</td></tr> <tr><td>Yêu cầu lớp DatBanDAO98 lưu thông tin vào CSDL</td><td>DatBanDAO98</td></tr> <tr><td>Xử lý sự kiện khi nút Thêm bị click</td><td></td></tr> </table>	GDThemKH98		Responsibilities:		Name	Collaborator	Hiển thị giao diện nhập thông tin khách hàng		Yêu cầu lớp Khachhang98 đóng gói thông tin đã nhập thành đối tượng	Khachhang98	Yêu cầu lớp DatBanDAO98 lưu thông tin vào CSDL	DatBanDAO98	Xử lý sự kiện khi nút Thêm bị click																																					
GDXacnhan98																																																															
Responsibilities:																																																															
Name	Collaborator																																																														
Hiển thị giao diện Xác nhận đặt bàn																																																															
Yêu cầu lớp LanDatBan98 đóng gói thông tin đặt bàn thành đối tượng	LanDatBan98																																																														
Yêu cầu lớp DatBanDAO98 lưu thông tin vào CSDL	DatBanDAO98																																																														
GDThemKH98																																																															
Responsibilities:																																																															
Name	Collaborator																																																														
Hiển thị giao diện nhập thông tin khách hàng																																																															
Yêu cầu lớp Khachhang98 đóng gói thông tin đã nhập thành đối tượng	Khachhang98																																																														
Yêu cầu lớp DatBanDAO98 lưu thông tin vào CSDL	DatBanDAO98																																																														
Xử lý sự kiện khi nút Thêm bị click																																																															

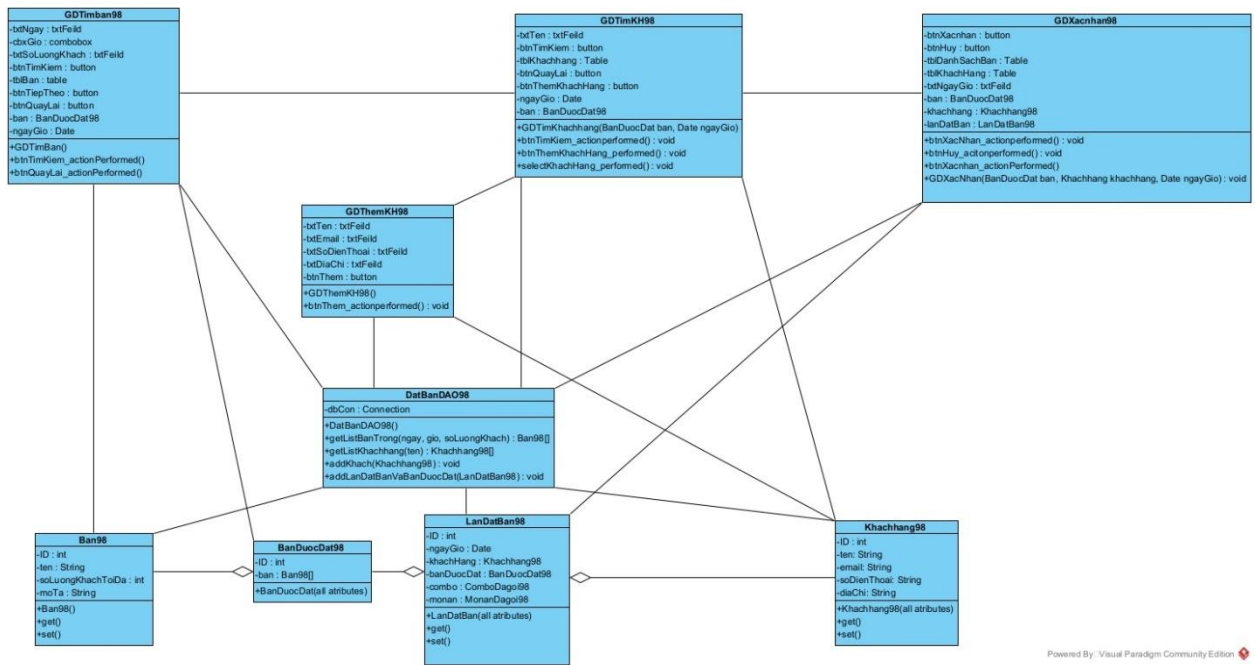
Powered By  UML Paradigm Community Edition

#### 3.2.2. Định nghĩa khuôn mẫu cho các phương thức

- Lớp GDTimBan98 phải có các phương thức: Hiển thị form (hàm khởi tạo), xử lý sự kiện các nút “Tìm kiếm” và nút “Tiếp theo” được click, xử lý sự kiện khi chọn các dòng trong bảng kết quả tìm kiếm.
- Lớp GDTimKH98 phải có các phương thức: Hiển thị form (hàm khởi tạo), xử lý sự kiện các nút “Tìm kiếm”, “Thêm mới khách hàng” được click, xử lý sự kiện khi chọn một dòng trong bản kết quả tìm kiếm.
- Lớp GDThemKH98 phải có các phương thức: Hiển thị form(hàm khởi tạo), xử lý sự kiện khi nút “Thêm” được click.
- Lớp GDXacnhan98 phải có các phương thức: Hiển thị form(hàm khởi tạo), xử lý sự kiện khi nút “Xác nhận”, “Hủy” được click.
- Lớp DatBanDAO phải có các phương thức: Tìm kiếm thông tin bàn trống, tìm kiếm thông tin khách hàng, lưu thông tin khách hàng vào CSDL, lưu thông tin lần đặt bàn vào CSDL

- Lớp Khachhang98 phải có các phương thức đóng gói thông tin (hàm khởi tạo và phương thức set)
- Lớp Ban98 phải có các phương thức đóng gói thông tin (hàm khởi tạo và phương thức set)
- Lớp LanDatBan98 phải có các phương thức đóng gói thông tin (hàm khởi tạo và phương thức set)

### 3.2.3. Sơ đồ lớp



## 4. Sơ đồ tuần tự



## 1. Chuẩn bị kiểm thử.

### 1.1.1. Chức năng tìm kiếm bản:

- Tìm bản đã có trong cơ sở dữ liệu:
  - Dữ liệu hiện thời:

Ban	ID	SoLuongKhachToiDa	Mota	Ten
	1	12	VIP	B01
	2	12	NORMAL	B02
	3	12	NORMAL	B03
	4	12	VIP	B04
	5	12	VIP	B05
	6	12	VIP	B
	7	12	NORMAL	B7

BanDuocDat	ID	LanDatBanID	BanID
	2	3	5
	3	4	2
	4	4	3
	5	4	4
	6	5	1
	7	5	5
	8	5	7
	9	6	2
	10	6	5
	11	6	7
	12	7	4
	13	7	5
	14	7	6

LanDatBan	ID	KhachhangID	NgayGio
	3	3	12/2/2024 0:00
	4	2	12/2/2024 8:00
	5	1	12/2/2024 8:00
	6	3	12/3/2024 1:00
	7	3	12/3/2024 8:00

○ Các thao tác và kết quả mong đợi:

Các bước thao tác	Kết quả mong đợi																				
1. Nhân viên click chức năng Đặt bàn từ giao diện chính	<div>GDTimBan98 hiện ra gồm ô nhập ngày, danh sách sổ xuống chọn khung giờ , số lượng khách hàng, một nút tìm kiếm và một bảng thể hiện kết quả gồm các trường như sau:</div> <div><div>Ngày<div>8:00</div><div>Số lượng khách</div><div>Tìm kiếm</div></div><table><tr><th>ID</th><th>Tên</th><th>Số lượng khách tối đa</th><th>Mô tả</th></tr></table></div>	ID	Tên	Số lượng khách tối đa	Mô tả																
ID	Tên	Số lượng khách tối đa	Mô tả																		
2. Nhân viên nhập: Tên = 2024-12-3 Giờ = 8:00 Số lượng khách = 20	<div>Bảng trong giao diện thể hiện kết quả sau:</div> <table><tr><th>ID</th><th>Tên</th><th>Số lượng khách tối đa</th><th>Mô tả</th></tr><tr><td>1</td><td>B01</td><td>12</td><td>VIP</td></tr><tr><td>2</td><td>B02</td><td>12</td><td>NORMAL</td></tr><tr><td>3</td><td>B03</td><td>12</td><td>NORMAL</td></tr><tr><td>7</td><td>B7</td><td>12</td><td>NORMAL</td></tr></table>	ID	Tên	Số lượng khách tối đa	Mô tả	1	B01	12	VIP	2	B02	12	NORMAL	3	B03	12	NORMAL	7	B7	12	NORMAL
ID	Tên	Số lượng khách tối đa	Mô tả																		
1	B01	12	VIP																		
2	B02	12	NORMAL																		
3	B03	12	NORMAL																		
7	B7	12	NORMAL																		

1.1.2. Chức năng tìm kiếm khách hàng:

- Tìm khách hàng đã có trong cơ sở dữ liệu
  - Dữ liệu hiện thời:

ID	Ten	Email	SoDienThoai	DiaChi
1	Nguyen Van A	nva@m.com	0912321332	Ha Dong
2	Tran Van B	tvbsd@n.com:	09124564543	Thanh Tri
3	Nguyen Van A	ndfva@m.com	0912321332	Ha Dong
4	Nguyen Van A	nvacx@m.com	0912321332	Ha Dong
5	Tran Van B	tvaab@n.com:	0912324543	Thanh Tri

○ Các thao tác và kết quả mong đợi

▪ Tìm một khách hàng đã có trong CSDL

Các bước thao tác	Kết quả mong đợi				
1. Nhân viên click nút tiếp theo từ GDTimBan98	GDTimKH98 hiện ra gồm một ô nhập tên khách hàng ,một nút tìm kiếm và một bảng thể hiện kết quả gồm các trường như sau:				
	ID	Tên	Email	Số điện thoại	Địa chỉ
2. Nhân viên nhập: Tên = Nguyen Van A	Bảng trong giao diện thể hiện kết quả sau:				
	ID	Tên	Email	Số điện thoại	Địa chỉ
	1	Nguyen Van A	nva@m.com	Ha Dong	0912321332
	3	Nguyen Van A	ndfva@m.com	Ha Dong	0912321332
	4	Nguyen Van A	nvacx@m.com	Ha Dong	0912321332

- Tìm khách hàng chưa có trong cơ sở dữ liệu:

○ Dữ liệu hiện thời:

ID	Ten	Email	SoDienThoai	DiaChi
1	Nguyen Van A	nva@m.com	0912321332	Ha Dong
2	Tran Van B	tvbsd@n.com:	09124564543	Thanh Tri
3	Nguyen Van A	ndfva@m.com	0912321332	Ha Dong
4	Nguyen Van A	nvacx@m.com	0912321332	Ha Dong
5	Tran Van B	tvaab@n.com:	0912324543	Thanh Tri

○ Các thao tác và kết quả mong đợi

▪ Tìm một khách hàng đã có trong CSDL

Các bước thao tác	Kết quả mong đợi				
1. Nhân viên click nút tiếp theo từ GDTimBan98	GDTimKH98 hiện ra gồm một ô nhập tên khách hàng ,một nút tìm kiếm và một bảng thể hiện kết quả gồm các trường như sau:				
	ID	Tên	Email	Số điện thoại	Địa chỉ
2. Nhân viên nhập: Tên = Nguyen Van A	Bảng trong giao diện thể hiện kết quả sau:				
	ID	Tên	Email	Số điện thoại	Địa chỉ

- Thêm khách hàng đã có trong cơ sở dữ liệu

○ Dữ liệu hiện thời:

ID	Ten	Email	SoDienThoai	DiaChi
1	Nguyen Van A	nva@m.com	0912321332	Ha Dong
2	Tran Van B	tvbsd@n.com:	09124564543	Thanh Tri
3	Nguyen Van A	ndfva@m.com	0912321332	Ha Dong
4	Nguyen Van A	nvacx@m.com	0912321332	Ha Dong
5	Tran Van B	tvaab@n.com:	0912324543	Thanh Tri

○ Các thao tác và kết quả mong đợi

▪ Tìm một khách hàng đã có trong CSDL

Các bước thao tác	Kết quả mong đợi
1. Nhân viên click nút thêm khách hàng từ GDTimKH98	GDThemKH98 hiện ra gồm một ô nhập tên khách hàng ,một nút tìm kiếm và một bảng thể hiện kết quả gồm các trường như sau: <div> <div>Tên</div> <div>Email</div> <div>Số điện thoại</div> <div>Địa chỉ</div> <div>Thêm</div> </div>
2. Nhân viên nhập: Tên = Nguyen Van B Email = <a href="mailto:x@gmail.com">x@gmail.com</a> Số điện thoại = 029393923 Địa chỉ = “Ha Dong”	Một cửa sổ thông báo thành công

○ Cơ sở dữ liệu được cập nhật:

1 Nguyen Van A	nva@m.com	0912321332	Ha Dong
2 Tran Van B	tvbsd@n.com:	09124564543	Thanh Tri
3 Nguyen Van A	ndfva@m.com	0912321332	Ha Dong
4 Nguyen Van A	nvacx@m.com	0912321332	Ha Dong
5 Tran Van B	tvaab@n.com:	0912324543	Thanh Tri
6 Nguyen Van B	<a href="mailto:x@gmail.com">x@gmail.com</a>	029393923	Ha Dong

- Tìm khách hàng chưa có trong cơ sở dữ liệu:

○ Dữ liệu hiện thời:

1 Nguyen Van A	nva@m.com	0912321332	Ha Dong
2 Tran Van B	tvbsd@n.com:	09124564543	Thanh Tri
3 Nguyen Van A	ndfva@m.com	0912321332	Ha Dong
4 Nguyen Van A	nvacx@m.com	0912321332	Ha Dong
5 Tran Van B	tvaab@n.com:	0912324543	Thanh Tri
6 Nguyen Van B	<a href="mailto:x@gmail.com">x@gmail.com</a>	029393923	Ha Dong

○ Các thao tác và kết quả mong đợi

▪ Thêm một khách hàng đã có trong CSDL

Các bước thao tác	Kết quả mong đợi
1. Nhân viên click nút tiếp theo từ GDTimBan98	GDTimKH98 hiện ra gồm một ô nhập tên khách hàng ,một nút tìm kiếm và một bảng thể hiện kết quả gồm các trường như sau: <div> <div>ID</div> <div>Tên</div> <div>Email</div> <div>Số điện thoại</div> <div>Địa chỉ</div> </div>
2. Nhân viên nhập: Tên = Nguyen Van B Email = <a href="mailto:x@gmail.com">x@gmail.com</a> Số điện thoại = 029393923	Một dialog thông báo khách hàng đã tồn tại hiện ra và cơ sở dữ liệu không được cập nhật

Địa chỉ = “Hà Đông”	
------------------------	--

- Chức năng thêm một lần đặt bàn và bàn được đặt vào cơ sở dữ liệu:
  - o Dữ liệu hiện thời:

Ban	ID	SoLuongKhachToiDa	Mota	Ten
	1	12	VIP	B01
	2	12	NORMAL	B02
	3	12	NORMAL	B03
	4	12	VIP	B04
	5	12	VIP	B05
	6	12	VIP	B
	7	12	NORMAL	B7

BanDuocDat	ID	LanDatBanID	BanID
	2	3	5
	3	4	2
	4	4	3
	5	4	4
	6	5	1
	7	5	5
	8	5	7
	9	6	2
	10	6	5
	11	6	7
	12	7	4
	13	7	5
	14	7	6

LanDatBan	ID	KhachhangID	NgayGio
	3	3	12/2/2024 0:00
	4	2	12/2/2024 8:00
	5	1	12/2/2024 8:00
	6	3	12/3/2024 1:00
	7	3	12/3/2024 8:00

ID	Ten	Email	SoDienThoai	DiaChi
1	Nguyen Van A	nva@m.com	0912321332	Hà Đông
2	Tran Van B	tvbsd@n.com:	09124564543	Thanh Trì
3	Nguyen Van A	ndfva@m.com	0912321332	Hà Đông
4	Nguyen Van A	nvacx@m.com	0912321332	Hà Đông
5	Tran Van B	tvaab@n.com:	0912324543	Thanh Trì

- o Các thao tác và kết quả mong đợi:

Các bước thao tác	Kết quả mong đợi
1. Nhân viên click chức năng Đặt bàn	GDTimBan98 hiện ra gồm ô nhập ngày, danh sách sổ xuống chọn khung giờ , số lượng khách hàng, một nút tìm kiếm, một nút tiếp

từ giao diện chính	theo và một bảng thể hiện kết quả gồm các trường như sau:																						
2. Nhân viên nhập: Tên = 2024-12-3 Giờ = 8:00 Số lượng khách = 20	<div>Bảng trong giao diện thể hiện kết quả sau:</div> <table><tr><th>ID</th><th>Tên</th><th>Số lượng khách tối đa</th><th>Mô tả</th></tr><tr><td>1</td><td>B01</td><td>12</td><td>VIP</td></tr><tr><td>2</td><td>B02</td><td>12</td><td>NORMAL</td></tr><tr><td>3</td><td>B03</td><td>12</td><td>NORMAL</td></tr><tr><td>7</td><td>B7</td><td>12</td><td>NORMAL</td></tr></table>	ID	Tên	Số lượng khách tối đa	Mô tả	1	B01	12	VIP	2	B02	12	NORMAL	3	B03	12	NORMAL	7	B7	12	NORMAL		
ID	Tên	Số lượng khách tối đa	Mô tả																				
1	B01	12	VIP																				
2	B02	12	NORMAL																				
3	B03	12	NORMAL																				
7	B7	12	NORMAL																				
3. Khách hàng chọn các bàn B01, B02. Nhân viên click chọn các bàn B01, B02 và bấm nút tiếp theo	<div>GDTimKH98 hiện ra gồm một ô nhập tên khách hàng ,một nút tìm kiếm và một bảng thể hiện kết quả gồm các trường như sau:</div> <table><tr><th>ID</th><th>Tên</th><th>Email</th><th>Số điện thoại</th><th>Địa chỉ</th></tr></table>	ID	Tên	Email	Số điện thoại	Địa chỉ																	
ID	Tên	Email	Số điện thoại	Địa chỉ																			
4. Nhân viên nhập: Tên = Nguyen Van A	<div>Kết quả trong bảng hiện ra như sau:</div> <table><tr><th>ID</th><th>Tên</th><th>Email</th><th>Số điện thoại</th><th>Địa chỉ</th></tr><tr><td>1</td><td>Nguyen Van A</td><td>nva@m.com</td><td>Ha Dong</td><td>0912321332</td></tr><tr><td>3</td><td>Nguyen Van A</td><td>ndfva@m.com</td><td>Ha Dong</td><td>0912321332</td></tr><tr><td>4</td><td>Nguyen Van A</td><td>nvacx@m.com</td><td>Ha Dong</td><td>0912321332</td></tr></table>	ID	Tên	Email	Số điện thoại	Địa chỉ	1	Nguyen Van A	nva@m.com	Ha Dong	0912321332	3	Nguyen Van A	ndfva@m.com	Ha Dong	0912321332	4	Nguyen Van A	nvacx@m.com	Ha Dong	0912321332		
ID	Tên	Email	Số điện thoại	Địa chỉ																			
1	Nguyen Van A	nva@m.com	Ha Dong	0912321332																			
3	Nguyen Van A	ndfva@m.com	Ha Dong	0912321332																			
4	Nguyen Van A	nvacx@m.com	Ha Dong	0912321332																			
5. Khách hàng chọn dòng thứ 2 sau đó nhân viên click vào dòng thứ 2	<div>Một giao diện xác nhận kết quả hiện ra như sau:</div> <table><tr><th>ID Bàn</th><th>Tên Bàn</th><th>Số lượng khách tối đa</th><th>Mô tả</th></tr><tr><td>1</td><td>B01</td><td>12</td><td>VIP</td></tr><tr><td>2</td><td>B02</td><td>12</td><td>NORMAL</td></tr></table> <table><tr><th>ID</th><th>Tên</th><th>Email</th><th>Số điện thoại</th><th>Địa chỉ</th></tr><tr><td>3</td><td>Nguyen Van A</td><td>ndfva@m.com</td><td>Ha Dong</td><td>0912321332</td></tr></table> <div>Giờ đặt bàn: 2024-12-03 08:00</div>	ID Bàn	Tên Bàn	Số lượng khách tối đa	Mô tả	1	B01	12	VIP	2	B02	12	NORMAL	ID	Tên	Email	Số điện thoại	Địa chỉ	3	Nguyen Van A	ndfva@m.com	Ha Dong	0912321332
ID Bàn	Tên Bàn	Số lượng khách tối đa	Mô tả																				
1	B01	12	VIP																				
2	B02	12	NORMAL																				
ID	Tên	Email	Số điện thoại	Địa chỉ																			
3	Nguyen Van A	ndfva@m.com	Ha Dong	0912321332																			
6. Khách hàng xác nhận sau đó nhân viên click nút xác nhận	Thông báo thành công và cơ sở dữ liệu sẽ có như sau:																						

○ Kết quả trong cơ sở dữ liệu:

LanDatBan	ID	KhachhangID	NgayGio
	3	3	12/2/2024 0:00
	4	2	12/2/2024 8:00
	5	1	12/2/2024 8:00
	6	3	12/3/2024 1:00
	7	3	12/3/2024 8:00
	8	3	12/3/2024 8:00



BanDuocDat	ID	LanDatBanID	BanID
	2	3	5
	3	4	2
	4	4	3
	5	4	4
	6	5	1
	7	5	5
	8	5	7
	9	6	2
	10	6	5
	11	6	7
	12	7	4
	13	7	5
	14	7	6
	15	8	1
	16	8	2

## 2. Cài đặt theo thiết kế MVC.

### 2.1. Lớp Ban98.

```

1 package model;
2
3 public class Ban98 {
4     private int ID;
5     private String ten;
6     private int soLuongKhachToiDa;
7     private String moTa;
8     public Ban98() {
9     }
10    public Ban98(int ID, String ten, int soLuongKhachToiDa, String moTa) {
11        this.ID = ID;
12        this.ten = ten;
13        this.soLuongKhachToiDa = soLuongKhachToiDa;
14        this.moTa = moTa;
15    }
16    public int getID() {
17        return ID;
18    }
19    public void setID(int ID) {
20        this.ID = ID;
21    }
22    public String getTen() {
23        return ten;
24    }
25    public void setTen(String ten) {
26        this.ten = ten;
27    }
28    public int getSoLuongKhachToiDa() {
29        return soLuongKhachToiDa;
30    }
31    public void setSoLuongKhachToiDa(int soLuongKhachToiDa) {

```

```

32         this.soLuongKhachToiDa = soLuongKhachToiDa;
33     }
34     public String getMoTa() {
35         return moTa;
36     }
37     public void setMoTa(String moTa) {
38         this.moTa = moTa;
39     }
40     public Object[] toObject(){
41         return new Object[]{ID, ten, soLuongKhachToiDa, moTa};
42     }
43

```

## 2.2.Lớp LanDatBan

```

1  package model;
2  import java.util.Date;
3  import model.BanDuocDat98;
4  import model.Khachhang98;
5  public class LanDatBan98 {
6      private int ID;
7      private Date ngayGio;
8      private Khachhang khachHang;
9      private BanDuocDat banDuocDat;
10     public LanDatBan98(Date ngayGio, Khachhang khachHang, BanDuocDat
        banDuocDat) {
11         this.ngayGio = ngayGio;
12         this.khachHang = khachHang;
13         this.banDuocDat = banDuocDat;
14     }
15     public int getID() {
16         return ID;
17     }
18     public Date getNgayGio() {
19         return ngayGio;
20     }
21     public Khachhang getKhachHang() {
22         return khachHang;
23     }
24     public BanDuocDat getBanDuocDat() {
25         return banDuocDat;
26     }
27     public void setID(int ID) {
28         this.ID = ID;
29     }
30     public void setNgay(Date ngayGio) {
31         this.ngayGio = ngayGio;
32     }
33     public void setKhachHang(Khachhang khachHang) {
34         this.khachHang = khachHang;
35     }
36     public void setBanDuocDat(BanDuocDat banDuocDat) {
37         this.banDuocDat = banDuocDat;
38     }

```

```
39 }
```

## 2.3.Lớp BanDuocDat98

```
1 package model;
2 import model.Ban98;
3 import java.util.List;
4 public class BanDuocDat98 {
5     private int ID;
6     private List<Ban98> ban;
7     public BanDuocDat98() {
8     }
9     public BanDuocDat98(int ID, List<Ban> ban) {
10         this.ID = ID;
11         this.ban = ban;
12     }
13     public int getID() {
14         return ID;
15     }
16     public void setID(int ID) {
17         this.ID = ID;
18     }
19     public List<Ban> getBan() {
20         return ban;
21     }
22     public void setBan(List<Ban> ban) {
23         this.ban = ban;
24     }
25 }
```

## 2.4.Lớp Khachhang

```
1 package model;
2 public class Khachhang98 {
3     private int ID;
4     private String ten;
5     private String email;
6     private String soDienThoai;
7     private String diaChi;
8     public Khachhang98(int ID, String ten, String email, String
soDienThoai, String diaChi) {
9         this.ID = ID;
10        this.ten = ten;
11        this.email = email;
12        this.soDienThoai = soDienThoai;
13        this.diaChi = diaChi;
14    }
15    public int getID() {
16        return ID;
17    }
18    public void setID(int ID) {
```

```

19         this.ID = ID;
20     }
21     public String getTen() {
22         return ten;
23     }
24     public void setTen(String ten) {
25         this.ten = ten;
26     }
27     public String getEmail() {
28         return email;
29     }
30     public void setEmail(String email) {
31         this.email = email;
32     }
33     public String getSoDienThoai() {
34         return soDienThoai;
35     }
36     public void setSoDienThoai(String soDienThoai) {
37         this.soDienThoai = soDienThoai;
38     }
39     public String getDiaChi() {
40         return diaChi;
41     }
42     public void setDiaChi(String diaChi) {
43         this.diaChi = diaChi;
44     }
45 }

```

### 3. Lớp DatBanDAO98

```

1. package control;
2. import model.Ban;
3. import java.sql.Connection;
4. import java.sql.DriverManager;
5. import java.sql.Statement;
6. import java.sql.PreparedStatement;
7. import java.sql.ResultSet;
8. import java.sql.SQLException;
9. import java.text.SimpleDateFormat;
10. import java.util.ArrayList;
11. import java.util.List;
12. import model.BanDuocDat;
13. import model.Khachhang;
14. import model.LanDatBan;
15.
16. public class DatBanDAO {
17.     public static Connection dbCon;
18.
19.     public DatBanDAO(){
20.         if(dbCon == null){
21.             String dbUrl =
22. "jdbc:mysql://localhost:3306/quanlinhahang?autoReconnect=true&useSSL=false";
23.             String dbClass = "com.mysql.jdbc.Driver";

```

```

24.         try {
25.             Class.forName(dbClass);
26.             dbCon = DriverManager.getConnection (dbUrl,
"root", "123456");
27.         } catch (ClassNotFoundException | SQLException e) {
28.         }
29.     }
30. }
31.
32.     public List<Ban> GetlistBanTrong(String ngayGio, String
soLuongKhach){
33.         String sql = "SELECT * FROM tblban WHERE ID NOT IN
(SELECT BanID FROM tblBanDuocDat WHERE LanDatBanID IN (SELECT ID
FROM tblLanDatBan WHERE NgayGio = ?))";
34.         List<Ban> listBan = new ArrayList<>();
35.         try{
36.             PreparedStatement ps = dbCon.prepareStatement(sql);
37.             ps.setString(1, ngayGio);
38.             ResultSet res = ps.executeQuery();
39.             while(res.next()){
40.                 SimpleDateFormat df = new SimpleDateFormat("yyyy-
MM-dd hh:mm:ss");
41.                 listBan.add(new
Ban(Integer.parseInt(res.getString("ID")),res.getString("Ten"),Inte
ger.parseInt(res.getString("SoLuongKhachToiDa")),res.getString("Mot
a"))));
42.             }
43.         } catch (NumberFormatException | SQLException e) {
44.         }
45.         return listBan;
46.     }
47.
48.     public List<Khachhang> GetListKhachhang(String ten){
49.         String sql = "SELECT * FROM tblkhachhang WHERE Ten = ?";
50.         List <Khachhang> listKH= new ArrayList<>();
51.         try{
52.             PreparedStatement ps = dbCon.prepareStatement(sql);
53.             ps.setString(1, ten);
54.             ResultSet rs = ps.executeQuery();
55.             while(rs.next()){
56.                 listKH.add(new
khachhang(Integer.parseInt(rs.getString("ID")),
rs.getString("Ten"), rs.getString("Email"), rs.getString("Diachi"),
rs.getString("SoDienThoai")));
57.             }
58.         }
59.         catch (NumberFormatException|SQLException e){
60.         }
61.         return listKH;
62.     }
63.
64.     public void addLanDatBan(LanDatBan lanDatBan){
65.         String sqlAddLanDatBan = "INSERT INTO
tblLanDatBan(KhachhangID, NgayGio) VALUES(?,?)";
66.         String sqlAddBanDuocDat = "INSERT INTO
tblBanDuocDat(LanDatBanID, BanID) VALUES(?,?)";
67.         SimpleDateFormat df = new SimpleDateFormat("yyyy-MM-dd
hh:mm");

```

```

68.         try{
69.             PreparedStatement ps =
dbCon.prepareStatement(sqlAddLanDatBan,
Statement.RETURN_GENERATED_KEYS);
70.             ps.setInt(1, lanDatBan.getKhachHang().getID());
71.             ps.setString(2, df.format(lanDatBan.getNgayGio()));
72.             ps.executeUpdate();
73.             ResultSet generatedKeys = ps.getGeneratedKeys();
74.             ps = dbCon.prepareStatement(sqlAddBanDuocDat);
75.             if(generatedKeys.next()){
76.                 lanDatBan.setID(generatedKeys.getInt(1));
77.                 for(Ban ban: lanDatBan.getBanDuocDat().getBan()){
78.                     ps.setInt(1, lanDatBan.getID());
79.                     ps.setInt(2, ban.getID());
80.                     ps.executeUpdate();
81.                 }
82.             }
83.         }
84.     }catch(SQLException e){
85.     }
86. }
87.
88.     public void addkhach(khachhang khachHang) throws Exception{
89.         String sqlAddkhachHang = "INSERT INTO tblkhachhang (Ten,
Email, SoDienThoai, DiaChi) VALUES(?,?,?,?)";
90.         String timkhachHang = "SELECT * FROM tblkhachhang WHERE
Ten = ? and Email=? and SoDienThoai=? and DiaChi=?";
91.         try{
92.             PreparedStatement ps =
dbCon.prepareStatement(sqlAddkhachHang);
93.             ps.setString(1, khachHang.getTen());
94.             ps.setString(2, khachHang.getEmail());
95.             ps.setString(3, khachHang.getSoDienThoai());
96.             ps.setString(4, khachHang.getDiaChi());
97.             ResultSet rs = ps.executeQuery();
98.             if(rs.next()){
99.                 throw new Exception("Đã tồn tại khách hàng");
100.            }
101.        }catch(SQLException e){
102.            try{
103.                PreparedStatement ps =
dbCon.prepareStatement(sqlAddkhachHang);
104.                ps.setString(1, khachHang.getTen());
105.                ps.setString(2, khachHang.getEmail());
106.                ps.setString(3, khachHang.getSoDienThoai());
107.                ps.setString(4, khachHang.getDiaChi());
108.                ps.executeUpdate();
109.            }catch(SQLException ex){
110.            }
111.        }
112.    }
113. }
114. }

```

#### 4. Lớp GDTimBan98

```

1. package view;
2. import java.awt.BorderLayout;
3. import java.awt.HeadlessException;

```

```

4. import java.awt.event.ActionEvent;
5. import java.awt.event.ActionListener;
6. import java.util.*;
7. import java.util.Date;
8. import javax.swing.JButton;
9. import javax.swing.JFrame;
10. import javax.swing.JPanel;
11. import javax.swing.JTable;
12. import javax.swing.JTextField;
13. import javax.swing.BoxLayout;
14. import control.DatBanDAO;
15. import java.text.ParseException;
16. import java.text.SimpleDateFormat;
17. import java.util.logging.Level;
18. import java.util.logging.Logger;
19. import javax.swing.JComboBox;
20. import javax.swing.JScrollPane;
21. import javax.swing.table.DefaultTableModel;
22. import model.Ban98;
23. import model.BanDuocDat98;
24. public class GDTimBan98 extends JFrame implements ActionListener{
25.     private JTextField txtNgay;
26.     private JComboBox cbxGio;
27.     private JTextField txtSoLuongKhach;
28.     private JButton btnTimKiem;
29.     private JTable tblBan;
30.     private JButton btnTiepTheo;
31.     private JButton btnQuayLai;
32.     private List<Ban98> listBan;
33.     private Date ngayGio;
34.     DefaultTableModel tbm;
35.     public GDTimBan98() throws HeadlessException {
36.         super("Tìm bàn trống");
37.         this.txtNgay = new JTextField("Ngày");
38.         String[] items = {"8:00", "10:00", "13:00", "15:00", "17:00",
"19:00"};
39.         this.cbxGio = new JComboBox(items);
40.         this.txtSoLuongKhach = new JTextField("Số lượng khách");
41.         this.btnTimKiem = new JButton("Tìm kiếm");
42.         this.btnTiepTheo = new JButton("Tiếp theo");
43.         this.btnQuayLai = new JButton("Quay lại");
44.         this.setSize(600,600);
45.         this.setLocation(200,10);
46.         this.setLayout(new BorderLayout());
47.         this.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
48.
49.         String[] columns = {"ID", "Tên", "Số lượng khách tối đa", "Mô
tả"};
50.         this.tbm = new DefaultTableModel(columns, 0);
51.         this.tblBan = new JTable();
52.         tblBan.setModel(tbm);
53.         JScrollPane header = new JScrollPane(tblBan);
54.

```

```

55.         JPanel nhap = new JPanel();
56.         nhap.setLayout(new BoxLayout(nhap, BoxLayout.X_AXIS));
57.         nhap.add(this.txtNgay);
58.         nhap.add(this.cbxCio);
59.         nhap.add(this.txtSoLuongKhach);
60.         nhap.add(this.btnTimKiem);
61.
62.         this.add(nhap, BorderLayout.NORTH);
63.         this.add(header, BorderLayout.CENTER);
64.
65.         JPanel nut = new JPanel();
66.         nut.setLayout(new BoxLayout(nut, BoxLayout.X_AXIS));
67.         nut.add(this.btnQuayLai);
68.         nut.add(this.btnTiepTheo);
69.
70.         this.add(nut, BorderLayout.SOUTH);
71.         btnTimKiem.addActionListener(this);
72.         btnTiepTheo.addActionListener(this);
73.         btnQuayLai.addActionListener(this);
74.     }
75.
76.     @Override
77.     public void actionPerformed(ActionEvent e) {
78.         JButton btn = (JButton)e.getSource();
79.         if(btn.equals(this.btnTimKiem)) {
80.             btnTimKiem_actionPerfromed();
81.         } else if(btn.equals(this.btnTiepTheo)) {
82.             btnTiepTheo_actionPerformed();
83.         } else if(btn.equals(this.btnQuayLai)) {
84.             btnQuayLai_actionPerformed();
85.         }
86.     }
87.     public void btnTimKiem_actionPerfromed() {
88.         DatBanDAO datBan = new DatBanDAO();
89.         tbm.setNumRows(0);
90.         SimpleDateFormat df = new SimpleDateFormat("yyyy-MM-dd
hh:mm");
91.         try {
92.             ngayGio = df.parse(txtNgay.getText()+"
"+cbxCio.getSelectedItem().toString());
93.         } catch (ParseException ex) {
94.             Logger.getLogger(GDTimBan.class.getName()).log(Level.SEVERE, null, ex);
95.         }
96.         listBan = datBan.GetlistBanTrong(txtNgay.getText()+"
"+cbxCio.getSelectedItem().toString(), txtSoLuongKhach.getText());
97.         for(Ban x: listBan) {
98.             tbm.addRow(x.toObject());
99.         }
100.
101.         tblBan.setModel(tbm);
102.         tblBan.setVisible(true);
103.     }

```



```

104.         public void btnTiepTheo_actionPerformed() {
105.             List<Ban98> banDuocDat = new ArrayList<>();
106.             for(int x :tblBan.getSelectedRows()) {
107.                 banDuocDat.add(listBan.get(x));
108.                 System.out.println(listBan.get(x).getID());
109.             }
110.             BanDuocDat98 ban = new BanDuocDat();
111.             ban.setBan(banDuocDat);
112.             (new GDTimKhachhang(ban, ngayGio)).setVisible(true);
113.             this.dispose();
114.         }
115.
116.         public void btnQuayLai_actionPerformed() {
117.             (new GDChinhNhanVien()).setVisible(true);
118.             this.dispose();
119.         }
120.     }

```

## 5. Lớp GDTimKH98

```

1. package view;
2. import control.DatBanDAO98;
3. import java.awt.BorderLayout;
4. import java.awt.event.ActionEvent;
5. import java.awt.event.ActionListener;
6. import java.awt.event.MouseEvent;
7. import java.awt.event.MouseListener;
8. import java.util.*;
9. import javax.swing.BoxLayout;
10.     import javax.swing.JButton;
11.     import javax.swing.JFrame;
12.     import javax.swing.JPanel;
13.     import javax.swing.JScrollPane;
14.     import javax.swing.JTable;
15.     import javax.swing.JTextField;
16.     import javax.swing.table.DefaultTableModel;
17.     import model.BanDuocDat98;
18.     import model.Khachhang98;
19.     import view.GDXacNhan98;
20.     public class GDTimKhachhang98 extends JFrame implements
        ActionListener, MouseListener{
21.         private JTextField txtTen;
22.         private JTable tblKhachhang;
23.         private JButton btnTimKiem;
24.         private JButton btnThem;
25.         private BanDuocDat98 ban;
26.         private Date ngayGio;
27.         private JButton btnQuayLai;
28.
29.         public GDTimKhachhang(BanDuocDat98 ban, Date ngayGio){
30.             this.txtTen = new JTextField("Tên");
31.             this.btnTimKiem = new JButton("Tìm kiếm");
32.             this.btnThem = new JButton("Thêm khách hàng mới");

```

```

33.         this.btnQuayLai = new JButton("Quay lại");
34.         this.ban = ban;
35.         this.ngayGio = ngayGio;
36.
37.         this.setSize(600, 600);
38.         this.setLocation(200, 10);
39.         this.setLayout(new BorderLayout());
40.         this.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
41.
42.         JPanel pane = new JPanel();
43.         pane.setLayout(new BoxLayout(pane, BoxLayout.X_AXIS));
44.         pane.add(txtTen);
45.         pane.add(btnTimKiem);
46.         String[] cols = {"ID", "Tên", "Email", "Số điện thoại", "Địa
chi"};
47.         DefaultTableModel tbm = new DefaultTableModel(cols, 0);
48.         tblKhachhang = new JTable(tbm);
49.         JScrollPane bang = new JScrollPane(tblKhachhang);
50.         this.add(pane, BorderLayout.NORTH);
51.         this.add(bang, BorderLayout.CENTER);
52.
53.         JPanel nut = new JPanel();
54.         nut.setLayout(new BoxLayout(nut, BoxLayout.X_AXIS));
55.         nut.add(this.btnQuayLai);
56.         nut.add(this.btnThem);
57.
58.         this.add(nut, BorderLayout.SOUTH);
59.         btnThem.addActionListener(this);
60.         btnTimKiem.addActionListener(this);
61.         btnQuayLai.addActionListener(this);
62.         tblKhachhang.addMouseListener(this);
63.     }
64.
65.     @Override
66.     public void actionPerformed(ActionEvent e) {
67.         JButton btn = (JButton)e.getSource();
68.         if(btn.equals(btnThem)) {
69.             btnThem_actionPerformed();
70.         }
71.         if(btn.equals(btnTimKiem)) {
72.             btnTimKiem_actionPerformed();
73.         }
74.     }
75.     public void btnThem_actionPerformed() {
76.
77.     }
78.     public void btnTimKiem_actionPerformed() {
79.         DatBanDAO dao = new DatBanDAO();
80.         List<Khachhang98> listKH =
dao.GetListKhachhang(txtTen.getText());
81.         DefaultTableModel tbm =
(DefaultTableModel)tblKhachhang.getModel();
82.         tbm.setRowCount(0);

```

```

83.         for (Khachhang98 x: listKH) {
84.             tblm.addRow(new Object[]{x.getID(), x.getTen(),
x.getEmail(), x.getSoDienThoai(), x.getDiaChi()});
85.         }
86.     }
87.
88.     public void selectKhachhang_performed() {
89.         int x = tblKhachhang.getSelectedRow();
90.         Khachhang98 kh = new
Khachhang98(Integer.parseInt(tblKhachhang.getValueAt(x, 0).toString()),
tblKhachhang.getValueAt(x, 1).toString(), tblKhachhang.getValueAt(x,
2).toString(), tblKhachhang.getValueAt(x, 3).toString(),
tblKhachhang.getValueAt(x, 4).toString());
91.         (new GDXacNhan(ban, kh, ngayGio)).setVisible(true);
92.         this.dispose();
93.     }
94.
95.     public void btnQuayLai_actionPerformed() {
96.         (new GDTimBan98()).setVisible(true);
97.         this.dispose();
98.     }
99.
100.    @Override
101.    public void mouseClicked(MouseEvent e) {
102.        if (e.getSource().equals(tblKhachhang)) {
103.            selectKhachhang_performed();
104.        }
105.    }
106.    @Override
107.    public void mousePressed(MouseEvent e) {}
108.    @Override
109.    public void mouseReleased(MouseEvent e) {}
110.    @Override
111.    public void mouseEntered(MouseEvent e) {}
112.    @Override
113.    public void mouseExited(MouseEvent e) {}
114.    }

```

### 5.1.Lớp GDThemKH98

```

1. package view;
2. import control.DatBanDAO;
3. import java.awt.BorderLayout;
4. import java.awt.event.ActionEvent;
5. import java.awt.event.ActionListener;
6. import javax.swing.BoxLayout;
7. import javax.swing.JButton;
8. import javax.swing.JFrame;
9. import javax.swing.JLabel;
10. import javax.swing.JOptionPane;
11. import javax.swing.JPanel;
12. import javax.swing.JTextField;
13. import model.Khachhang98;
14. public class GDThemKH98 extends JFrame implements ActionListener{
15.     private JTextField txtTen;

```

```

16.     private JTextField txtEmail;
17.     private JTextField txtSoDienThoai;
18.     private JTextField txtDiaChi;
19.     private JButton btnThem;
20.
21.     public GDThemKH(){
22.         txtTen = new JTextField();
23.         txtEmail = new JTextField();
24.         txtSoDienThoai = new JTextField();
25.         txtDiaChi = new JTextField();
26.         btnThem = new JButton("Thêm");
27.         this.setSize(600,600);
28.         this.setLocation(200,10);
29.         this.setLayout(new BorderLayout());
30.         this.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
31.
32.         JPanel pane = new JPanel();
33.         pane.setLayout(new BoxLayout(pane, BoxLayout.X_AXIS));
34.         JLabel lbTen = new JLabel("Tên");
35.         pane.add(lbTen);
36.         pane.add(txtTen);
37.         JLabel lbEmail = new JLabel("Email");
38.         pane.add(lbEmail);
39.         pane.add(txtEmail);
40.         JLabel lbSoDienThoai = new JLabel("Số điện thoại");
41.         pane.add(lbSoDienThoai);
42.         pane.add(txtSoDienThoai);
43.         JLabel lbDiaChi = new JLabel("Địa chỉ");
44.         pane.add(lbDiaChi);
45.         pane.add(txtDiaChi);
46.         this.add(pane, BorderLayout.NORTH);
47.         this.add(btnThem, BorderLayout.SOUTH);
48.         btnThem.addActionListener(this);
49.     }
50.
51.     @Override
52.     public void actionPerformed(ActionEvent e) {
53.
54.     }
55.     public void btnThem_actionPerformed(){
56.         Khachhang98 kh = new Khachhang98();
57.         kh.setTen(txtTen.getText());
58.         kh.setEmail(txtEmail.getText());
59.         kh.setSoDienThoai(txtSoDienThoai.getText());
60.         kh.setDiaChi(txtDiaChi.getText());
61.         DatBanDAO dao = new DatBanDAO();
62.         try{
63.             dao.addkhach(kh);
64.             this.dispose();
65.         }catch(Exception e){
66.             JOptionPane.showMessageDialog(this, "Khách hàng đã
tồn tại");
67.         }
68.     }
69. }

```

## 6. Lớp GDXacNhan

```

1. package view;
2. import control.DatBanDAO98;
3. import java.awt.BorderLayout;
4. import java.awt.Color;
5. import java.awt.event.ActionEvent;
6. import java.awt.event.ActionListener;
7. import java.text.SimpleDateFormat;
8. import java.util.Date;
9. import javax.swing.BoxLayout;
10.     import javax.swing.JButton;
11.     import javax.swing.JFrame;
12.     import javax.swing.JOptionPane;
13.     import javax.swing.JPanel;
14.     import javax.swing.JScrollPane;
15.     import javax.swing.JTable;
16.     import javax.swing.JTextField;
17.     import javax.swing.table.DefaultTableModel;
18.     import model.Ban;
19.     import model.BanDuocDat;
20.     import model.Khachhang98;
21.     import model.LanDatBan98;
22.     public class GDXacNhan98 extends JFrame implements ActionListener{
23.         private BanDuocDat98 ban;
24.         private Khachhang98 khachhang;
25.         private Date ngayGio;
26.         private JButton btnXacNhan;
27.         private JButton btnHuy;
28.         private JTable tblDanhSachBan;
29.         private JTable tblKhachHang;
30.         private JTextField txtNgayGio;
31.         private LanDatBan98 lanDatBan;
32.
33.         public GDXacNhan98(BanDuocDat98 ban, Khachhang98 khachhang, Date
ngayGio) {
34.             this.setSize(600,600);
35.             this.setLocation(200,10);
36.             this.setLayout(new BorderLayout());
37.             this.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
38.
39.             this.ban = ban;
40.             this.khachhang = khachhang;
41.             this.ngayGio = ngayGio;
42.             this.btnHuy = new JButton("Hủy");
43.             this.btnXacNhan = new JButton("Xác nhận");
44.             DefaultTableModel bm = new DefaultTableModel(
45.                 new String[]{"ID Bàn", "Tên Bàn", "Số lượng khách tối
đa", "Mô tả"}, 0
46.             );
47.             DefaultTableModel kh = new DefaultTableModel(
48.                 new String[]{"ID", "Tên", "Email", "Số điện thoại", "Địa
chỉ"}, 0
49.             );
50.             tblDanhSachBan = new JTable();

```

```

51.         tblKhachHang = new JTable();
52.         tblDanhSachBan.setModel(bm);
53.         tblKhachHang.setModel(kh);
54.         kh.addRow(new Object[]{khachhang.getID(), khachhang.getTen(),
khachhang.getEmail(), khachhang.getSoDienThoai(), khachhang.getDiaChi()});
55.         for(Ban x: ban.getBan()){
56.             bm.addRow(x.toObject());
57.         }
58.         JScrollPane jbm = new JScrollPane(tblDanhSachBan);
59.         JScrollPane jkh = new JScrollPane(tblKhachHang);
60.         JPanel bang = new JPanel();
61.         bang.setLayout(new BorderLayout(bang, BorderLayout.Y_AXIS));
62.         bang.add(jbm);
63.         bang.add(jkh);
64.
65.         JPanel nut = new JPanel();
66.         nut.setLayout(new BorderLayout(nut, BorderLayout.X_AXIS));
67.         nut.add(btnHuy);
68.         nut.add(btnXacNhan);
69.
70.         this.add(bang, BorderLayout.CENTER);
71.         this.add(nut, BorderLayout.SOUTH);
72.         SimpleDateFormat df = new SimpleDateFormat("yyyy-MM-dd
hh:mm");
73.         txtNgayGio = new JTextField("Giờ đặt bàn:
"+df.format(this.ngayGio));
74.         txtNgayGio.setForeground(Color.red);
75.         txtNgayGio.setEnabled(false);
76.         bang.add(txtNgayGio);
77.         btnHuy.addActionListener(this);
78.         btnXacNhan.addActionListener(this);
79.     }
80.
81.     @Override
82.     public void actionPerformed(ActionEvent e) {
83.         JButton btn = (JButton) e.getSource();
84.         if(btn.equals(btnHuy)){
85.             btnHuy_actionPerformed();
86.         } else if(btn.equals(btnXacNhan)){
87.             btnXacnhan_actionPerformed();
88.         }
89.     }
90.
91.     public void btnHuy_actionPerformed(){
92.
93.     }
94.     public void btnXacnhan_actionPerformed(){
95.         this.lanDatBan = new LanDatBan98(ngayGio, khachhang, ban);
96.         DatBanDAO98 dao = new DatBanDAO98();
97.         dao.addLanDatBan(lanDatBan);
98.         JOptionPane.showMessageDialog(this, "Đặt bàn thành công");
99.         (new GDChinhNhanVien98()).setVisible(true);
100.        this.dispose();

```

```

101.         }
102.     }

```

## 7. Lớp Test

```

1. package test;
2.
3. import view.GDTimBan98;
4.
5. public class test {
6.     public class Test {
7.         public static void main(String[] args) {
8.             GDTimBan98 gdtb = new GDTimBan98();
9.             gdtb.setVisible(true);
10.        }
11.    }
12. }

```

## 8. Kiểm thử module.

Test case	Kết quả
1. Thêm một LanDatBan chưa có trong CSDL	Passed
2. Thêm một BanDuocDat chưa có trong CSDL	Passed
3. Tìm kiếm một bàn đã có trong CSDL	Passed
4. Tìm kiếm một khách hàng đã có tên trong CSDL	Passed
5. Tìm kiếm khách hàng chưa có trong CSDL	Passed
6. Thêm khách hàng chưa có trong CSDL	Passed
7. Thêm khách hàng đã có trong CSDL	Passed