**3.1 Thiết kế kiến trúc**

**Mô hình MVC**

MVC viết tắt của Model – View – Controller, đây là một mô hình trong lập trình. Nó cho phép tách biệt các đoạn mã xử lý và giao diện người dụng thành các thành phần riêng biệt.



Mô hình MVC cơ bản

Model: thành phần này được giao nhiệm vụ cung cấp dữ liệu cho cơ sở dữ liệu và lưu trữ dữ liệu.

View: thành phần này hiển thị các thông tin cho người dùng và nhận các dữ liệu từ người dùng gửi cho các controller.

Controller: thành phần này có nhiệm vụ cập nhật nội dung ở các View, nhận dữ liệu từ các View, truy xuất các thông tin cần thiết từ Model, xử lý các thao tác, cập nhật dữ liệu ở Model,…

Ưu điểm: Giúp phát triển ứng dụng nhanh, dễ nâng cấp, bảo trì…  
Nhược điểm: Đối với những ứng dụng nhỏ gây cồng kềnh, tốn thời gian trong quá trình trung chuyên dữ liệu giữa các thành phần.

**Mô hình 3 lớp**

 Mô hình 3 lớp cơ bản

Mô hình 3 lớp được cấu thành từ: Presentation Layers, Business Layers, và Data Layers. Các lớp này sẽ giao tiếp với nhau thông qua các dịch vụ (services) mà mỗi lớp cung cấp để tạo nên ứng dụng, lớp này cũng không cần biết bên trong lớp kia làm gì mà chỉ cần biết lớp kia cung cấp dịch vụ gì cho mình và sử dụng nó.

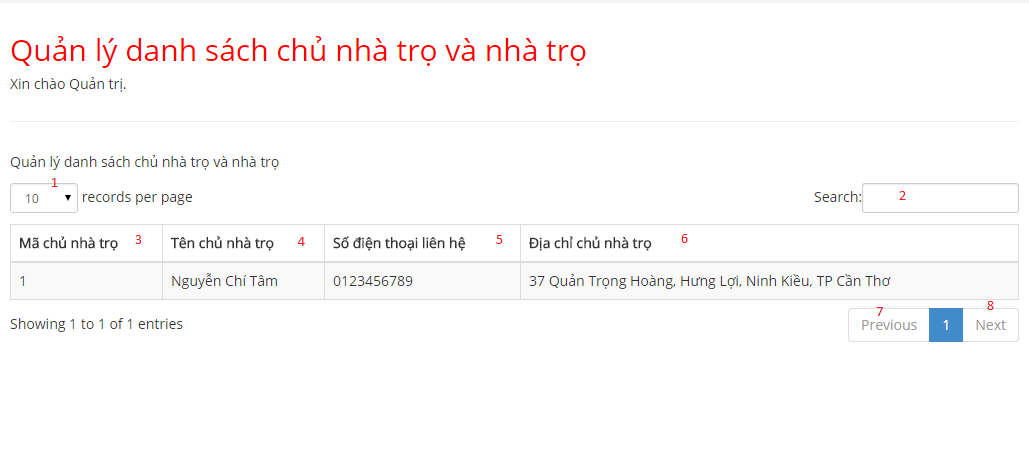
- Presentation Layer (GUI): Là thành phần giao diện, là các form của chương trình tương tác với người dùng hệ thống. Lớp này làm nhiệm vụ giao tiếp với người dùng cuối để thu thập dữ liệu và hiển thị kết quả, dữ liệu thông qua các đối tượng trong giao diện người sử dụng.   
- BusinessLogicLayer (BLL): được phân công là lớp xử lý các nghiệp vụ của chương trình như tính toán, xử lý các yêu cầu và kiểm tra tính hợp lệ và toàn vẹn về mặt dữ liệu trước khi được đưa lên hiển thị trên màn hình hoặc xử lý các dữ liệu trước khi chuyển xuống Data Access Layer để lưu dữ liệu vào database.  
- Data Access Layer (DAL): Chức năng của lớp này là giao tiếp với các hệ quản trị cơ sở dữ liệu. Lớp này thực hiện các nghiệp vụ liên quan đến lưu trữ và truy xuất dữ liệu của ứng dụng như đọc (select), lưu, cập nhật (insert, update, delete) cơ sở dữ liệu.

3.2

**5.1 Chức năng quản lý danh sách nhà trọ**

- Mục đích: giúp quản trị hệ thống có thể quản lý được danh sách chủ nhà trọ và nhà trọ của chủ nhà trọ

- Giao điện:



- Các thành phần trong giao diện:

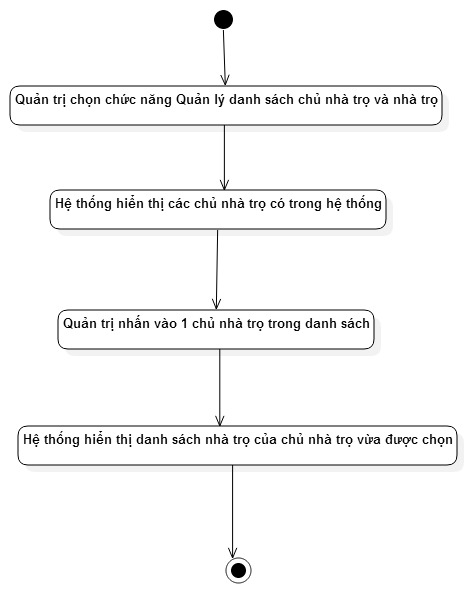
|  |  |  |  |
| --- | --- | --- | --- |
| STT | Loại điều khiển | Giá trị mặc định | Ghi chú |
| 1 | Combobox | 10 | Hiển thị số dòng hiển thị ở bảng hiển thị danh sách chủ nhà trọ |
| 2 | Textbox |  | Nhập thông tim tìm kiếm chủ nhà trọ |
| 3 | Character |  | Hiển thị mã chủ nhà trọ |
| 4 | Charcter |  | Hiển thị tên chủ nhà trọ |
| 5 | Character |  | Hiển thị số điện thoại liên hệ của chủ nhà trọ |
| 6 | Character |  | Hiển thị địa chỉ chủ nhà trọ |
| 7 | Button |  | Về trang phân trang trước |
| 8 | Button |  | Về trang phân trang sau |

* Giao diện quản lý danh sách nhà trọ của chủ nhà trọ



- Các thành phần trong giao diện:

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Loại điều khiển | Giá trị mặc định | Ghi chú |
| 1 | Combobox | 10 | Hiển thị số dòng hiển thị ở bảng hiển thị danh sách nhà trọ |
| 2 | Textbox |  | Nhập thông tim tìm kiếm nhà trọ |
| 3 | Character |  | Hiển thị mã chủ nhà trọ |
| 4 | Charcter |  | Hiển thị tên nhà trọ |
| 5 | Character |  | Hiển thị tên chủ nhà trọ |
| 6 | Character |  | Hiển thị số điện thoại liên hệ của chủ nhà trọ |
| 7 | Character |  | Hiển thị địa chỉ nhà trọ |
| 8 | Character |  | Hiển thị số phóng của nhà trọ |
| 9 | Character |  | Hiển thị giá thuê phóng của nhà trọ |
| 10 | Character |  | Hiển thị các mô tả tổng quan về nhà trọ. |
| 11 | Button |  | Về trang phân trang trước |
| 12 | Button |  | Về trang phân trang sau |

Xử lý