Báo cáo đồ án Kiến trúc phần mềm

*<QuanLyTruongMamNon>*

# Thông tin nhóm thực hiện

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | MSSV | Họ tên | Email | Điện thoại |
| 1 | 0812543 | Trần Văn Tri | [trantri2006@gmail.com](mailto:trantri2006@gmail.com) | 0974009485 |
| 2 | 0812609 | Nguyễn Văn Việt | [vanviet.uos@gmail.com](mailto:vanviet.uos@gmail.com) | 01689938202 |

# Thông tin chung về đề tài

Chương trình QuanLyTruongMamNon hỗ trợ **quản lí trường mầm non.** Chương trình bao gồm các chức năng chính sau đây:

***-Quản lí trẻ:***

* Khi trẻ được đăng kí gửi vào trường thì **tạo thông tin hồ sơ của trẻ**( bao gồm tên trẻ, ngày sinh, giới tính, tính trạng sức khỏe, cân nậng, chiều cao hiên tại, sở thích, thói quen, bệnh bẩm sinh, khuyết tật …) và **thông tin phụ huynh** (bao gồm họ tên cha mẹ hoặc người đỡ đầu, địa chỉ liên lạc, điện thoại, nghề nghiệp…)
* Người quản lý có thể **Cập nhật, chỉnh sửa, xóa thông tin** của trẻ **trong lớp hiện tại**(Không cho phép xóa hoàn toàn thông tin của trẻ trong hệ thống) khi có sự thay đổi.
* *Khi xóa hay cập nhật thông tin của trẻ, người dùng có thể tra cứu thông tin của trẻ. Cho phép xem thông tin chi tiết của trẻ.*
* Trẻ được tổ chức theo ***nhóm trẻ hoặc lớp mẫu giáo***

1. ***Đối với nhóm trẻ:*** trẻ từ 3 tháng tuổi đến 36 tháng tuổi được tổ chức thành các nhóm trẻ. Số trẻ tối đa trong 1 nhóm trẻ được qui định như sau:
   * Nhóm trẻ(nhóm 1) từ 3 đến 12 tháng tuổi :15 trẻ/lớp.
   * Nhóm trẻ(nhóm 2) từ13 đến 24 tháng tuổi: 20 trẻ/lớp.
   * Nhóm trẻ(nhóm 3) từ 25 đến 36 tháng tuổi: 25 trẻ/lớp.
2. ***Đối với lớp mẫu giáo:*** trẻ từ 3 đến 6 tuổi được tổ chức thành các lớp mẫu giáo. Số trẻ tối đa trong 1 lớp mẫu giáo được qui định như sau:

* Lớp mẫu giáo(lớp mầm) 3 đến 4 tuổi: 25 trẻ/lớp.
* Lớp mẫu giáo(lớp chồi) 4 đến 5 tuổi: 30 trẻ/lớp.
* Lớp mẫu giáo(lớp lá) 5 đến 6 tuổi: 35 trẻ/lớp.
  + Hệ thống cho phép người dùng thêm trẻ vào đúng lớp qui định. Nếu sau một thời gian qui định, những trẻ có độ tuổi vượt quá qui định của nhóm lớp sẽ xuất thông báo cho người dùng.
  + Hệ thống sẽ kiểm tra sỉ số hiện tại và độ tuổi của trẻ có phù hợp với lớp đăng kí hay không? Nếu không sẽ không cho phép đăng kí trẻ.
* Sau mỗi tháng chương trình cho phép cập nhập thông tin **kiểm tra định kì** của trẻ về thể chất(chiều cao, cân nặng, …).

***-Quản lí lớp:***

* Trường bao gồm 2 khối lớp như trên: nhà trẻ(nhóm a) và mẫu giáo(nhóm b).
* Người dùng có thể ***Thêm, Cập nhập hay Xóa một khối lớp***.
* Khôi nhà trẻ gồm 3 nhóm lớp(như trên) với mỗi nhóm gồm nhiều lớp(ví dụ nhóm 1a, 1b, 1c, 2a, 3a,…..) và sỉ số của mỗi lớp theo như qui định.
* Khối mẫu giáo gồm 3 nhóm lớp(như trên) với mỗi nhóm gồm nhiều lớp(ví dụ mầm 1, mầm 2, chồi 1, lá 1 , lá 2….) và sỉ số của mỗi lớp theo như qui định.
* Mỗi lớp có 2 giáo viên và 1 bảo mẫu phụ trách. Hệ thống cho phép ***thêm, cập nhập, xóa thông tin lớp học.***
* *Khi cập nhật ,xóa thông tin của lớp, người dùng có thể tra cứu thông tin của lớp và hiển thị danh sách trẻ trong lớp đó. Khi tra cứu thông tin lớp có thể xem thông tin chi tiết của trẻ.*
* Người dùng có thể thực hiện lập báo cáo hàng tháng về tình hình học tập, sức khỏe, điểm đánh giá thi đua của trẻ cũng như giáo viên phụ trách lớp đó. Cho phép in báo cáo đó.

***-Quản lý GV-CB-CNV:***

* Thông tin của một giáo viên gồm có: họ tên, ngày sinh, giới tính, chức vụ, địa chỉ, số điện thoại, bằng cấp, mức lương, phụ cấp…
* Chương trình cho phép ***thêm, cập nhập, xóa thông tin giáo viên.***

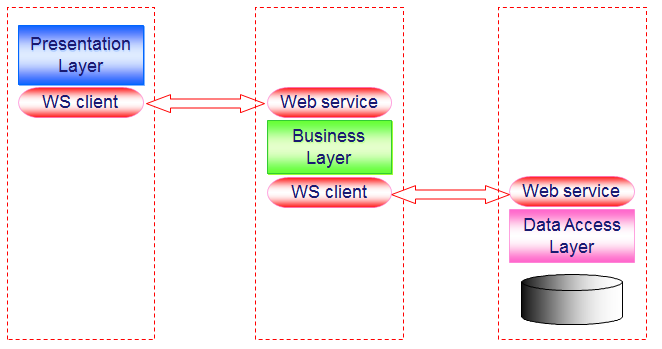
***-Báo cáo hàng tháng:***

* Thực hiện lập báo cáo hàng tháng về tình hình hoạt động và so sánh điểm thi đua của các lớp, cho phép in báo cáo.
* Thực hiện lập báo cáo về thu chi, quỹ hoạt động hàng tháng, cho phép in báo cáo.

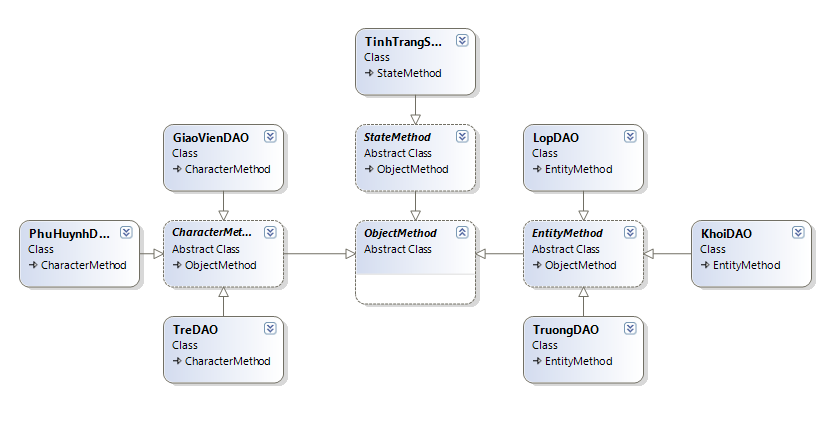
# Nội dung triển khai

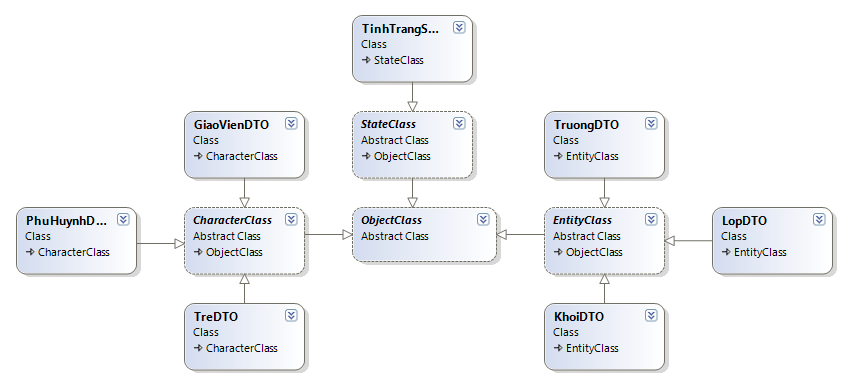
## Kiến trúc đề tài

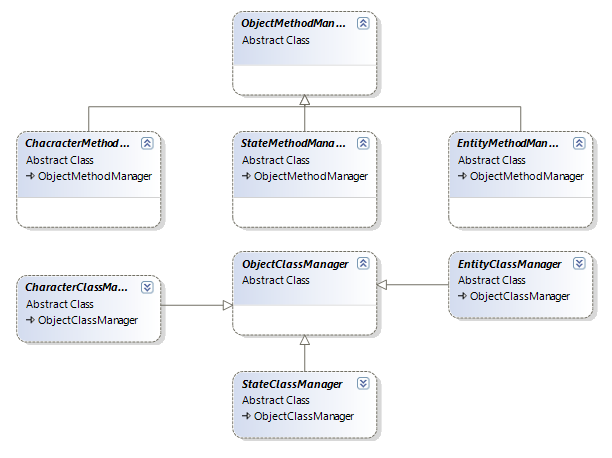
* Mô hình 3 tier kết hợp 3layer:



* GUI (Presentation) Layer: Nhập liệu và trình bày dữ liệu, có thể bao gồm các bước kiểm tra dữ liệu trước khi gọi Business Logic Layer.
* Business Logic Layer: Kiểm tra các yêu cầu nghiệp vụ trước khi cập nhật dữ liệu, quản lý các transaction, quản lý các concurrent access.
* Data Access Layer: Kết nối CSDL, tìm kiếm, thêm, xóa, sửa,…trên CSDL.
* Sử dụng Factory Pattern:
  + Tạo abstract class Manager để quản lý việc tạo lập và gọi các phương thức của từng lớp: Tre, Phu Huynh, Giao Vien, Truong, Lop, Khoi, TinhTrangSucKhoe







* Distribute Object:
  + Sử dụng RemoteObjectEngine trong đó có class RemoteObjectManager: quản lý các đối tượng ở Client, gọi các hàm của dịch vụ Webservice của Server.

## Các kĩ thuật sử dụng

* Dùng **delegate** để thay đổi giao diện chương trình. Cho phép người dùng thiết lập giao diện chương trình.
* Dùng **LinQ** để truy xuất dữ liệu nên các dữ liệu được chuyển thành đối tượng, truy xuất dễ dàng hơn.

# Kết luận

* Hoàn thành 1 ứng dụng với các chức năng trên
* Áp dụng một số mẫu thiết kế và các kiến trúc, kĩ thuật đã học
* Nhờ các mẫu kiến trúc mà việc xử lý, kết nối dữ liệu trở nên linh động hơn. Chương trình dễ nâng cấp, bảo trì hơn.
* Có thể tái sử dụng các hàm cung cấp của WebService cho các ứng dụng khác có cùng chức năng. VD: phát triển ứng dụng này trên WEB cho người dùng có thể dùng bất kì nơi đâu. Phụ huynh có thể dễ dàng theo dõi con em mình.