# I. Architecturally significant design packages

## 1. Technical constraints

Dưới đây là các ràng buộc phải được xem xét khi thiết kế kiến trúc cho hệ thống:

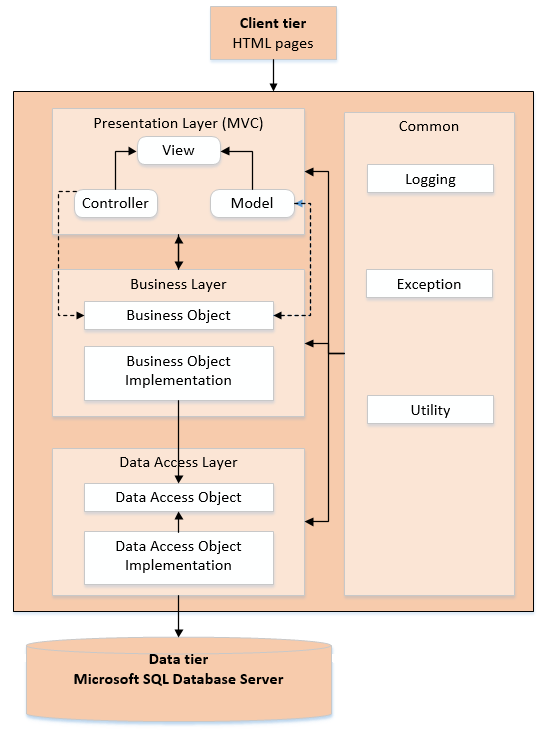
* System Framework: sử dụng MVC 4.5, repository pattern.
* Client Framework: sử dụng template admin AdminLTE-2.3.11:
* HTML, JavaScript, Ajax.
* Web responsive.
* Web support: IE >= 11, Chrome >= 40, Firefox 4.5.
* Database: Microsoft SQL Server 2012.
* Web Application server: IIS web server.

## 2. Logical view

- Phần này mô tả các công nghệ và framework được sử dụng trong tất cả các lớp của ứng dụng này.

- Kiến trúc của hệ thống được thiết kế theo kiến trúc n-tier và MVC.

- Sơ đồ dưới đây cho thấy các lớp logic chính của kiến trúc và các lớp logic này tương tác với nhau như thế nào.



### a. Presentation Layer

* Lớp này dùng để hiển thị màn hình cho end-user.
* Quản lý yêu cầu / phản hồi từ / tới của client.
* Gồm các controller dùng để điều hướng và gọi tới các method trong lớp business logic.
* Xử lý ngoại lệ từ các lớp khác.

### b. Business Layer

Lớp này quản lý các business rule và business logic. Nó bao gồm một bộ Business Objects để hỗ trợ thực hiện business logic của hệ thống.

* Xử lý business logic và business validation.
* Quản lý transactions.
* Cho phép interfaces tương tác với các lớp khác.

### c. Data Access Layer

Lớp này quản lý truy cập các đối tượng.

Chia sẻ các đối tượng truy cập dữ liệu (DAO) giữa các ứng dụng dễ dàng hơn.

Lớp này quản lý việc đọc, viết, cập nhật và xóa dữ liệu được lưu trữ.

### d. Common classes

***Logging Package***:

Package này dùng để xuất ra lịch sử thao tác của người dùng khi sử dụng ứng dụng.

Sử dụng Package log4net từ nuget.

Có 4 loại log: Error, Info, Debug, Warning.

* DEBUG: ở cấp độ này, các thông tin debug sẽ được log.
* INFO: ở cấp độ này, các thông tin về luồng làm việc của chương trình sẽ được log.
* WARN: cấp độ cho phép chúng ta log các thông tin cảnh báo của chương trình.
* ERROR: các lỗi khi chạy chương trình sẽ được log.

***Exception Package***:

Package này sẽ bao gồm tất cả các ngoại lệ chung thường được sử dụng.

Có 2 loại exception:

* *System Exception*: Một exception sẽ được ném ra trong trường hợp hệ thống xảy ra lỗi như kết nối lỗi với cơ sở dữ liệu, lỗi đọc tập tin hệ thống ...
* *Functional Exception*: Một exception liên quan đến business logic, ví dụ như xác thực dữ liệu đầu vào, chứng thực ...

Tối thiểu các chi tiết sau đây luôn phải được bắt và hiển thị trong các thông báo lỗi:

- Điều gì đã xảy ra khi lỗi đã được xác định. Điều này nên bao gồm những: Class, Chức năng, Method đã được gọi và từ đâu.

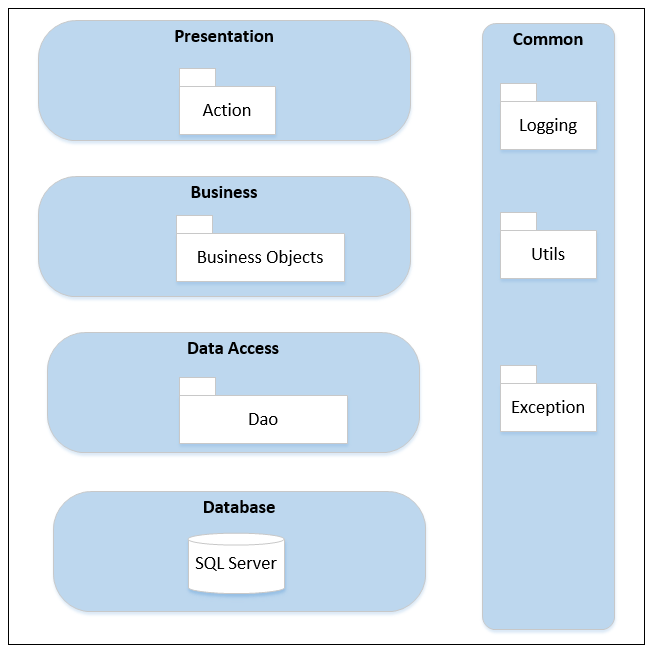
- Các giá trị của bất kỳ biến tại thời điểm lỗi đã được xác định.

***Utility Package***:

- Bao gồm các class dùng chung cho các function.

## 3. Architecturally significant design packages

### a. Package structure

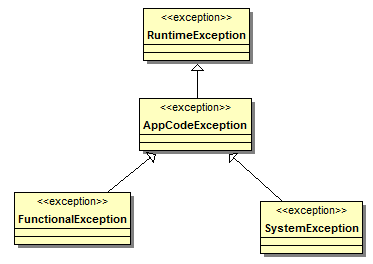


### b. Error and exception handling

Ứng dụng sẽ sử dụng cơ chế bắt tất cả các exception.

Dưới đây là sơ đồ package exception:

***Class Diagram***



*FunctionalException:* là các exception logic, chúng được ném ra trong trường hợp các parameters chuyển giữa các phương thức là không chính xác, sai loại, thiếu dữ liệu (dữ liệu hoặcbusiness validation)…

*SystemException*: là RuntimeException, bất kỳ lỗi không mong đợi như trên bộ nhớ, sự cố về cơ sở dữ liệu…

### c. Log, trace, and debug

Ứng dụng sẽ sử dụng log4net – package logging chuẩn. Chúng chia thành 4 loại logging:

* **Info**: ở cấp độ này, các thông tin về luồng làm việc của chương trình sẽ được log.
* **Warn**: cấp độ cho phép chúng ta log các thông tin cảnh báo của chương trình.
* **Debug**: ở cấp độ này, các thông tin debug sẽ được log.
* **Error**: các lỗi khi chạy chương trình sẽ được log.

### d. Common flow

## 4. Software and deployment architecture

|  |  |
| --- | --- |
| **Application Server** | |
| Operating Systems | Windows |
| Application Logging | Log4net |
| **HTTP Server** | |
| Web Server | IIS web server |
| Operating Systems | Windows |
| **Database Server** | |
| Operating Systems | Windows |
| DB Management System | Microsoft SQL Server |