TRƯỜNG ĐẠI HỌC KINH TẾ

**KHOA THỐNG KÊ – TIN HỌC**

–––––––––––––––––––––––––––––––



**BÁO CÁO THỰC TẬP NGHỀ NGHIỆP**

**NGÀNH HỆ THỐNG THÔNG TIN QUẢN LÝ**

**CHUYÊN NGÀNH QUẢN TRỊ HỆ THỐNG THÔNG TIN**

LẬP TRÌNH DANH MỤC,CHỨNG TỪ CHO PHẦN MỀM ERP

Sinh viên thực hiện : Nguyễn Minh Hiếu

Lớp : 45K21.1

Đơn vị thực tập : Công ty Phần mềm FAST

Cán bộ hướng dẫn : Phan Thanh Sáng

Giảng viên hướng dẫn : ThS. Cao Thị Nhâm

**Đà Nẵng, 8/202**

**NHẬN XÉT CỦA ĐƠN VỊ THỰC TẬP**

Họ và tên sinh viên:

Lớp: Khoa: Trường:

Thực tập từ ngày: …./……/ 2022 đến ngày: ........./ .……./ 2022

Tại:

Địa chỉ:

Sau quá trình thực tập tại đơn vị của sinh viên, chúng tôi có một số nhận xét, đánh giá như sau:

**1. Về thái độ, ý thức, đạo đức, kỷ luật**

**2. Kiến thức chuyên môn**

**3. Khả năng hòa nhập và thích nghi với công việc**

**4. Trách nhiệm, sáng tạo trong công việc**

**5. Các nhận xét khác**

**Đánh giá chung:**

**Điểm:**

……….., ngày .......tháng ......năm 2022

**Xác nhận của đơn vị thực tập**

# LỜI CẢM ƠN

# LỜI CAM ĐOAN

Quy định:

Soạn thảo trên trang A4 (trang dọc, lề trái: 3.5cm; trên, phải, dưới: 2.5 cm), Font Times New Roman, canh đều 2 bên, size 13, cách dòng 1.5, cách đoạn trên 6pt, cách đoạn dưới 3pt, hàng đâu tiên lùi vào 1.27 cm; hình và bảng soạn theo caption, chèn trích dẫn chéo (Cross-reference) cho bảng và hình; các danh mục hình, bảng, mục lục làm tự động; đánh số trang như file mẫu (bìa không có số trang, danh mục + mục lục số trang theo i, ii, iii…, nội dung chính theo 1,2,3…)

# MỤC LỤC

[NHẬN XÉT CỦA DOANH NGHIỆP THỰC TẬP ii](#_Toc98339732)

[LỜI CẢM ƠN iii](#_Toc98339733)

[LỜI CAM ĐOAN iv](#_Toc98339734)

[MỤC LỤC vi](#_Toc98339735)

[DANH MỤC HÌNH ẢNH viii](#_Toc98339736)

[DANH MỤC BẢNG BIỂU ix](#_Toc98339737)

[DANH MỤC CÁC TỪ VIẾT TẮT x](#_Toc98339738)

[LỜI MỞ ĐẦU 1](#_Toc98339739)

[CHƯƠNG 1. TỔNG QUAN … 2](#_Toc98339740)

[1.1. Mục 1.1 2](#_Toc98339741)

[1.1.1. Mục 1.1.1 2](#_Toc98339742)

[1.1.2. Mục 1.1.2 2](#_Toc98339743)

[1.2. Mục 1.2 2](#_Toc98339744)

[CHƯƠNG 2. LÝ THUYẾT 3](#_Toc98339745)

[2.1. Mục 2.1 3](#_Toc98339746)

[2.1.1. Mục 2.1.1 3](#_Toc98339747)

[2.1.2. Mục 2.1.2 3](#_Toc98339748)

[2.2. Mục 2.2 3](#_Toc98339749)

[CHƯƠNG 3. TRIỂN KHAI … 4](#_Toc98339750)

[3.1. Mục 3.1 4](#_Toc98339751)

[3.1.1. Mục 3.1.1 4](#_Toc98339752)

[3.1.2. Mục 3.1.2 4](#_Toc98339753)

[3.2. Mục 3.2 4](#_Toc98339754)

[CHƯƠNG 4. KẾT QUẢ 5](#_Toc98339755)

[4.1. Mục 4.1… 5](#_Toc98339756)

[4.2. Mục 4.2… 5](#_Toc98339757)

[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 6](#_Toc98339758)

[TÀI LIỆU THAM KHẢO 7](#_Toc98339759)

[PHỤ LỤC 8](#_Toc98339760)

*(Mục lục này chỉ là ví dụ)*

# DANH MỤC HÌNH ẢNH

[Hình 1.1 Ngôn ngữ lập trình Python 2](#_Toc98336120)

[Hình 4.1 Kiến trúc của mô hình 5](#_Toc98336121)

# DANH MỤC BẢNG BIỂU

[Bảng 4.1 Kiến trúc 5](#_Toc74235471)

# DANH MỤC CÁC TỪ VIẾT TẮT

**AI** : Artificial Intelligence

**…**

# LỜI MỞ ĐẦU

1. **Mục tiêu nghiên cứu của đề tài**

* Đề tài này nghiên cứu …

1. **Nhiệm vụ của đề tài**

* Nghiên cứu ...
* ...

1. **Phương pháp nghiên cứu**

* ...

1. **Đối tượng và phạm vi nghiên cứu**

* ...

1. **Kết cấu của đề tài**

Đề tài được tổ chức gồm phần mở đầu, x chương nội dung và phần kết luận...

* Mở đầu
* **Chương 1**:
* **Chương 2**:
* **Chương 3**:
* **Chương 4**:
* Kết luận và hướng phát triển

# CƠ SỞ LÝ THUYẾT

## XML là gì? Tìm hiểu về XML.

### XML là gì?

**XML** là từ viết tắt của từ **Extensible Markup Language**là ngôn ngữ đánh dấu mở rộng. XML có chức năng truyền dữ liệu và mô tả nhiều loại dữ liệu khác nhau. Tác dụng chính của XML là đơn giản hóa việc chia sẻ dữ liệu giữa các nền tảng và các hệ thống được kết nối thông qua mạng Internet.

XML dùng để cấu trúc, lưu trữ và trong trao đổi dữ liệu giữa các ứng dụng và lưu trữ dữ liệu. Ví dụ khi ta xây dựng một ứng dụng bằng Php và một ứng dụng bằng Java thì hai ngôn ngữ này không thể hiểu nhau, vì vậy ta sẽ sử dụng XML để trao đổi dữ liệu. Chính vì vậy, XML có tác dụng rất lớn trong việc chia sẻ, trao đổi dữ liệu giữa các hệ thống.

### Ưu và nhược điểm của XML.

**Ưu điểm**:

* Ưu điểm lớn nhất của XML là sự độc lập. XML được sử dụng để mô tả dữ liệu dưới dạng text, nên hầu hết các phần mềm hay các chương trình bình thường đều có thể đọc được chúng.
* XML có thể đọc và phân tích nguồn dữ liệu khá dễ dàng nên nó được sử dụng với mục đích chính là trao đổi dữ liệu giữa các chương trình, các hệ thống khác nhau. XML được sử dụng cho Remote Procedure Calls nhằm phục vụ cho các dịch vụ của website.

**Nhược điểm**: Tỷ lệ sai sót khi sử dụng XML để truyền dữ liệu khoảng từ 5-7%. Con số này tuy không quá cao, nhưng trên thực tế, người ta vẫn cần cân nhắc trước khi sử dụng nó để trao đổi thông tin.

### So sánh XML và HTML.

|  |  |
| --- | --- |
| **XML** | **HTML** |
| XML cho phép người dùng tạo biểu tượng đánh dấu riêng để mô tả nội dung, tạo một biểu tượng không giới hạn và tự định nghĩa | Được định nghĩa trước và người dùng phải tuân thủ |
| Được thiết kế để chuyển tải và lưu trữ dữ liệu | Được thiết kế để hiển thị dữ liệu |

### Cú pháp của XML

#### Khai báo XML (XML Declaration)

Tài liệu XML có thể tùy ý có một phần khai báo XML. Nó được viết như sau:

|  |
| --- |
| <?xml version=“1” encoding=“UTF-8”?> |

Version là phiên bản XML và encoding xác định mã hóa ký tự được sử dụng trong tài liệu.

#### Các qui tắc cú pháp để khai báo XML

* Khai báo XML (XML declaration) là phân biệt kiểu chữ và phải bắt đầu với “<?xml>” ở đây “xml” viết ở dạng chữ thường.
* Nếu tài liệu chứa khai báo XML, thì nó phải là lệnh đầu tiên của tài liệu XML.
* Khai báo XML phải là lệnh đầu tiên của tài liệu XML.
* Một HTTP Protocol có thể ghi đè giá trị của encoding mà bạn đặt trong khai báo XML.

#### Tags và Elements

Một XML file được cấu thành bởi một số phần tử XML (Element), còn được gọi là XML-node hoặc XML-tags. Tên các phần tử XML được bao trong các dấu < > như sau:

|  |
| --- |
| <element> |

#### Qui tắc cú pháp (Syntax Rules) cho các thẻ và phần tử

**Cú pháp phần từ**: Mỗi phần tử XML cần được bao trong hoặc các phần tử bắt đầu hoặc kết thúc như sau:

|  |
| --- |
| <element>…</element> |

**Luồng các phần tử**: Một phần tử XML có thể chứa nhiều phần tử XML khác như là con của nó, nhưng các phần tử con này phải không đè lên nhau, ví dụ: Một thẻ đóng của một phần tử phải có cùng tên như thẻ mở kết nối với nó.

|  |
| --- |
| <?xml version="1.0"?>  <grid>  <fields>…</fields>  </grid> |

**Phần tử gốc(Root Element)**: Lồng các phần tử: Một phần tử Một tài liệu XML có thể chỉ có một phần tử gốc. Ví dụ sau minh họa một tài liệu XML sai cú pháp, bởi vì cả hai phần tử x và y xuất hiện ở cấp cao nhất mà không phải là một phần tử gốc.

**Phân biệt kiểu chữ**: Tên của các phần tử XML là phân biệt kiểu chữ. Nghĩa là tên của thẻ mở và thẻ đóng phải cùng kiểu.

#### Thuộc tính (Attributes)

Một thuộc tính xác định thuộc tính cho phần tử, sử dụng một cặp tên/giá trị. Một phần tử XML có thể có một hoặc nhiều thuộc tính. Ví dụ:

|  |
| --- |
| <field name="ma\_hd" isPrimaryKey="true" width="100" > |

#### Qui tắc cú pháp cho thuộc tính trong XML

* Tên thuộc tính trong XML là phân biệt kiểu chữ (không giống như HTML). Tức là, width và WIDTH là hai thuộc tính khác nhau trong XML.
* Cùng một thuộc tính không thể có hai giá trị trong một cú pháp.
* Tên thuộc tính được định nghĩa không có sự trích dẫn, trong khi giá trị thuộc tính phải luôn luôn trong các dấu trích dẫn

#### Tham chiếu trong XML

**Tham chiếu (References)** thường cho phép bạn thêm hoặc bao phần text hoặc phần đánh dấu bổ sung trong một tài liệu XML. Các tham chiếu luôn luôn bắt đầu với biểu tượng “&” , đây là ký tự dành riêng và kết thúc với ký tự “;”.

|  |
| --- |
| <field name="ngay\_hd" allowFilter="&GridListAllowFilter;"> |

XML có hai kiểu tham chiếu:

**Tham chiếu thực thể (Entity Reference)**: Một tham chiếu thực thể chứa một tên giữa dấu tách mở và dấu tách đóng. Ví dụ: &amp; có amp là tên. Tên tham chiếu tới một chuỗi văn bản hoặc đánh dấu đã được định nghĩa trước.

**Tham chiếu ký tự (Character Reference**): Chứa các tham chiếu, ví dụ &#65;, chứa một dấu băm (#) được theo sau bởi một số. Số này luôn luôn tham chiếu tới mã hóa Unicode của ký tự. Trong ví dụ này, 65 tham chiếu tới chữ cái “A”.

#### Text trong XML

* Tên của phần tử XML và thuộc tính XML là phân biệt kiểu chữ, nghĩa là tên của phần tử mở và phần tử đóng phải ở được viết cùng kiểu.
* Để tránh các vấn đề về mã hóa ký tự, tất cả XML file nên được lưu ở dạng Unicode UTF-8 hoặc UTF-16.
* Các ký tự whitespace như khoảng trắng, tab và ngắt dòng giữa các phần tử XML và giữa các thuộc tính XML sẽ bị bỏ qua.
* Một số ký tự được dành riêng trong cú pháp XML. Vì thế, chúng không thể được sử dụng một cách trực tiếp. Để sử dụng chúng, một số thực thể thay thế được sử dụng, các thực thể này được liệt kê trong bảng dưới:

|  |  |  |
| --- | --- | --- |
| ***Ký tự không được phép sử dụng*** | ***Thực thể thay thế*** | ***Miêu tả*** |
| < | &lt; | Nhỏ hơn |
| > | &gt; | Lớn hơn |
| & | &amp; | Và |
| ‘ | &apos; | Ngoặc đơn |
| “ | &quot; | Ngoặc kép |

## SQL là gì? SQL có thể làm gì?

### SQL là gì?

SQL, viết tắt của Structured Query Language, là ngôn ngữ truy vấn có cấu trúc, cho phép bạn truy cập và thao tác với các cơ sở dữ liệu để tạo, xóa, sửa đổi, trích xuất dữ liệu.

SQL cũng là ngôn ngữ tiêu chuẩn cho các hệ cơ sở dữ liệu quan hệ. Tất cả các hệ thống quản trị cơ sở dữ liệu (RDBMS) như MySQL, MS Access, Oracle, Sybase, Informix, Postgres hay SQL Server đều lấy SQL làm ngôn ngữ cơ sở dữ liệu tiêu chuẩn.

### SQL có thể làm gì?

* Tạo cơ sở dữ liệu mới
* Tạo bảng mới trong cơ sở dữ liệu
* Tạo view (khung nhìn) mới
* Thực hiện truy vấn trên cơ sở dữ liệu
* Giup mô tả dữ liệu
* Tạo, chèn, xóa, sửa đổi bản ghi trong cơ sở dữ liệu
* Trích xuất dữ liệu từ cơ sở dữ liệu
* Thiết lập quyền trên bảng, thủ tục và view
* Nhúng các ngôn ngữ khác sử dụng mô-đun SQL, thư viện, trình biên dịch sẵn.

### Các lệnh SQL

##### DDL - Data Definition Language (Ngôn ngữ định nghĩa dữ liệu)

CREATE: Tạo bảng mới, view của bảng và các đối tượng khác trong cơ sở dữ liệu.

ALTER: Chỉnh sửa các đối tượng dữ liệu đã có, như bảng.

DROP: Xóa toàn bộ bảng, view của bảng hoặc các đối tượng khác trong cơ sở dữ liệu.

##### DML - Data Manipulation Language (Ngôn ngữ để thao tác với dữ liệu)

SELECT: Trích xuất bản ghi cụ thể từ một hoặc nhiều bảng

INSERT: Chèn dữ liệu mới vào cơ sở dữ liệu.

UPDATE: Sửa đổi, cập nhật dữ liệu trong cơ sở dữ liệu.

DELETE: Xóa dữ liệu từ cơ sở dữ liệu.

##### DCL - Data Control Language (Ngôn ngữ kiểm soát dữ liệu)

GRANT: Cấp đặc quyền cho user

REVOKE: Lấy lại quyền đã cấp cho user

## Javascript là gì? Ưu điểm và nhược điểm.

### Javascript là gì?

JavaScript là 1 trong 3 ngôn ngữ chính của lập trình web, và nó được dùng phổ biến trong suốt 20 năm qua. Từ thuở sơ khai, nó còn có tên là Mocha (năm 1995), sau đó được đổi thành Mona, Livescript, và cuối cùng là JavaScript như hiện nay.

### Ưu điểm và nhược điểm:

#### Ưu điểm:

* JavaScript là ngôn ngữ lập trình dễ học.
* Lỗi của JavaScript dễ phát hiện hơn và vì vậy dễ sửa hơn.
* JavaScript hoạt động trên nhiều trình duyệt, nền tảng.
* JavaScript giúp website tương tác tốt hơn với khách truy cập.
* JavaScript nhanh hơn và nhẹ hơn các ngôn ngữ lập trình khác.

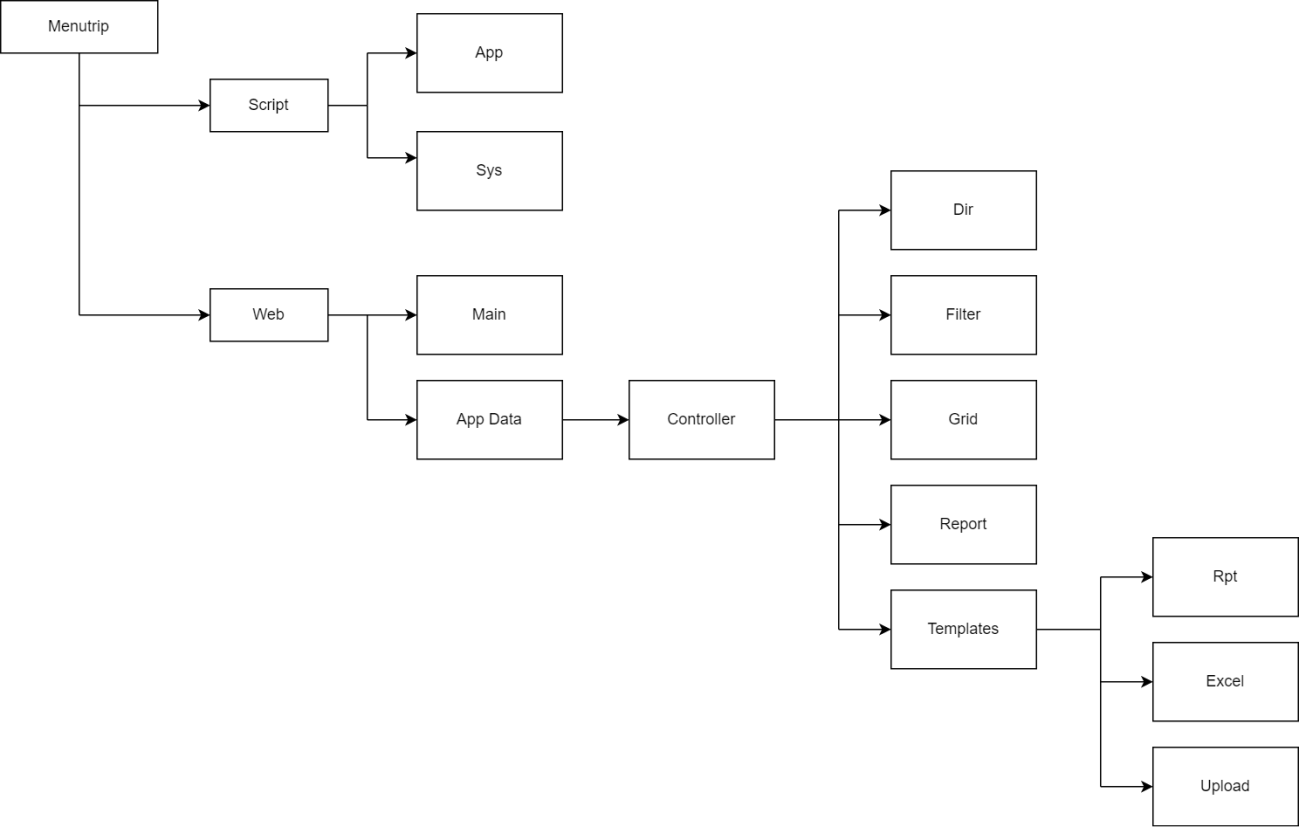
#### Nhược điểm:

Bởi JavaScript là ngôn ngữ lập trình phổ biến, được sử dụng rộng rãi trên nền tảng website, vì vậy nó cũng là đối tượng tiềm năng của một lượng lớn hacker, scammer – những kẻ chuyên tìm kiếm lỗ hổng và các lỗi bảo mật và thâm nhập, đánh cắp dữ liệu nội bộ.

Một số khuyết điểm của JavaScript có thể kể đến như:

* Dễ bị khai thác.
* Có thể được dùng để thực thi mã độc trên máy tính của người dùng.
* Có thể bị triển khai khác nhau tùy từng thiết bị dẫn đến việc không đồng nhất.

## Sơ đồ cấu trúc:

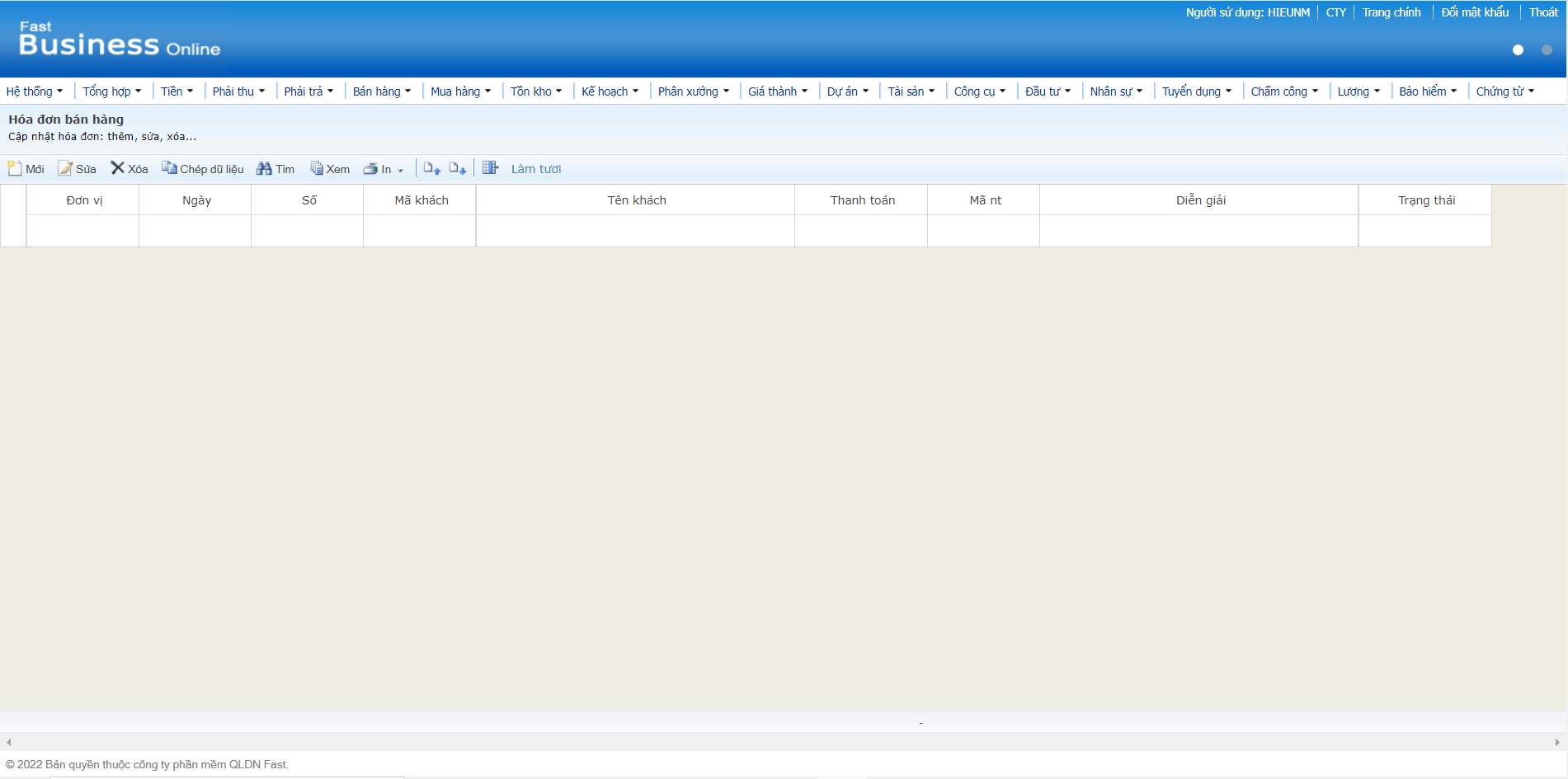


Trong đó:

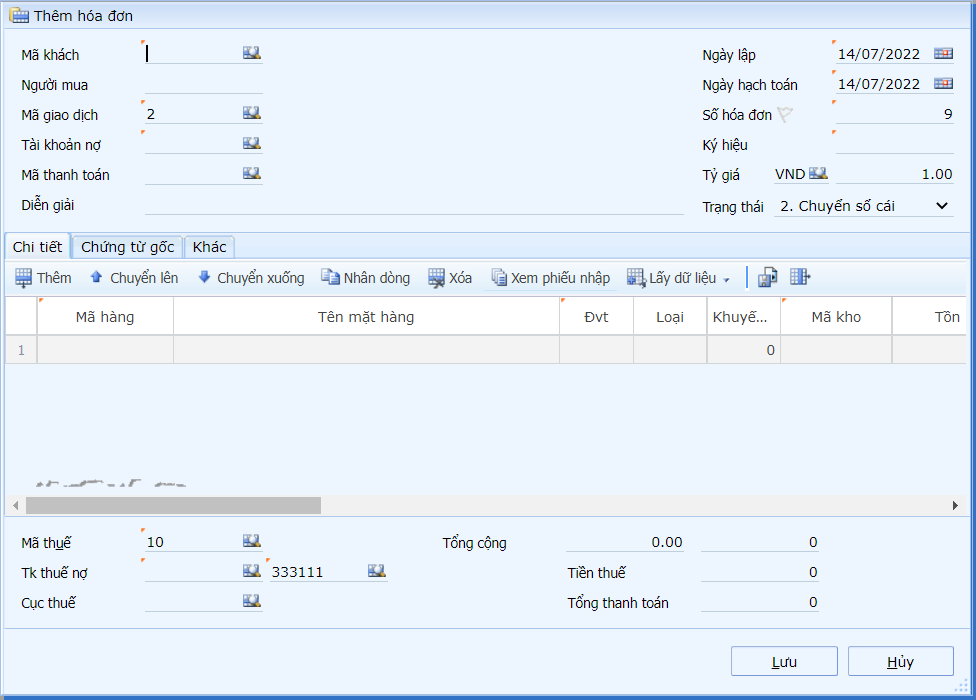
* App: Danh sách file database App
* Sys: Danh sách file database Sys
* Main: File chạy chương trình
* Dir: File controller thư mục dir
* Filter: File controller thư mục Filter
* Grid: File controller thư mục Grid
* Report: File controller thư mục Report
* Rpt: File báo cáo Crytal Report
* Excel: File mẫu import excel, hoặc file báo cáo excel
* Upload: File controller thư mục Upload

## Hình ảnh của các màn hình:

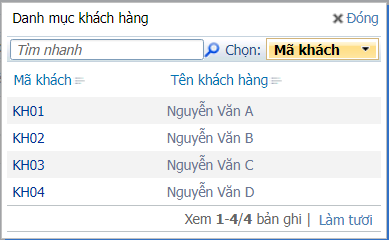
-Màn hình browse của danh mục



-Màn hình thêm/sửa/xóa của danh mục:

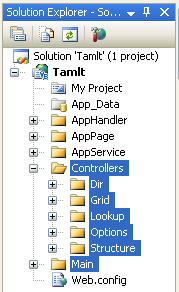


-Màn hình lookup lúc nhập liệu:



## Tạo mới một dự án:

* Cấu trúc trang web sẽ có cấu trúc như sau:



* Thư mục Main: chứa các trang web .aspx
* Thư mục Controller: chứa các định nghĩa bằng file xml để mô tả về một đối tượng nào đó, ở đây phân chia làm các đối tượng: Dir, Grid, Lookup, Options và Structure, đối với chứng từ sẽ có sau này thì xuất hiện thêm thư mục Filter và Include.
* Thư mục Dir: chứa các file xml để mô tả cho màn hình thêm, sửa trong danh mục. Ví dụ màn hình thêm/sửa của danh mục vụ việc.
* Thư mục Grid: chức các file xml để mô tả cho màn hình browser trong danh mục. Ví dụ màn hình browse của danh mục vụ việc.
* Thư mục Lookup: chứa các file xml để mô tả cho màn hình tìm kiếm giá trị cần, được dùng để nhập liệu nhanh cho 1 trường nào đó.
* Thư mục Option: chứa các file xml để mô tả cho từ điển các giá trị dùng chung, các giá trị ở đây được sử dụng xuyên suốt trong toàn bộ dự án.

## Các thẻ và thuộc tính của thẻ

#### Thẻ Grid

|  |  |  |
| --- | --- | --- |
| **Thẻ <grid> trong Controller\Grid\.xml** | | |
| Thẻ gốc chứa toàn bộ thông tin của 1 grid | | |
| Thuộc tính | Diễn giải | Giá trị mặc định |
| table | Khai báo bảng chứa dữ liệu của danh mục, hoặc view (trường hợp có dùng view).  Nếu là chứng từ thì thuộc tính này chứa tên bảng master để lấy cấu trúc. | String  Thuộc tính không được để trống |
| code | Khai báo trường các khóa của danh mục, các khóa cách nhau bởi dấu phẩy. | String  Thuộc tính không được để trống |
| order | Khai báo các trường sắp xếp thứ tự của danh mục, các khóa cách nhau bởi dấu phẩy. | String  Thuộc tính không được để trống |
| xmlns | Trường xác định xml namspase, dùng để tạo các assist tooltip hỗ trợ để lập trình. | String |
| Thẻ con | Diễn giải | Ghi chú |
| [title](#grid_grid_title) | Thẻ dùng để khai báo các tiêu đề của grid. |  |
| [subTitle](#grid_grid_subtitle) | Thẻ dùng để khai báo các tiêu đề phụ được hiển thị phía trên thanh toolbar. |  |
| [fields](#grid_grid_fields) | Thẻ dùng để chứa khối khai báo các trường dữ liệu dùng cho grid. |  |
| [views](#grid_grid_views) | Thẻ dùng để chứa khối khai báo các trường nào trong thẻ fields sẽ được hiển thị trong grid. |  |
| [partition](#grid_grid_partition) | Thẻ dùng để khai bảng master, detail và các thông tin chính khác của chứng từ. | Dùng cho chứng từ |
| [queries](#grid_grid_queries) | Thẻ dùng để xử lý cho các sự kiện truy vấn dữ liệu. | Dùng cho chứng từ |
| [toolbar](#grid_grid_toobar) | Thẻ chứa các khai báo cho thanh công cụ phía trên grid. | Dùng cho chứng từ |

|  |  |  |
| --- | --- | --- |
| **Thẻ <title> trong trong Controller\Grid\.xml > grid** | | |
| Thẻ dùng để khai báo các tiêu đề của grid | | |
| Thuộc tính | Diễn giải | Giá trị mặc định |
| v | Tên tiêu đề tiếng Việt | String = “” |
| e | Tên tiêu đề tiếng Anh | String = “” |

|  |  |  |
| --- | --- | --- |
| **Thẻ <subtitle> trong trong Controller\Grid\.xml > grid** | | |
| Thẻ dùng để khai báo các tiêu đề phụ được hiển thị phía trên thanh toolbar | | |
| Thuộc tính | Diễn giải | Giá trị mặc định |
| v | Tên tiêu đề tiếng Việt | String = “” |
| e | Tên tiêu đề tiếng Anh | String = “” |
| **Thẻ <fields> trong trong Controller\Grid\.xml > grid** | | |
| Thẻ dùng để chứa khối khai báo các trường dữ liệu dùng cho grid. | | |
| Thẻ con | Diễn giải | Ghi chú |
| field | Thẻ dùng định nghĩa các trường của bảng dữ liệu được lấy từ database | Có ít nhất là 1 thẻ field. Các thẻ field này phân biệt bằng thuộc tính “name”. |
| Thuộc tính của thẻ con | Diễn giải | Giá trị mặc định |
| aliasName | Dùng để xác định trường này được lấy chính xác từ bảng nào trong câu truy vấn.  aliasName sẽ là “a” hoặc “b” tùy theo trường muốn lấy ở bảng nào. |  |
| align | Canh lề dữ liệu của cột tương ứng. Có 3 giá trị: left, right, center. | “left” |
| allowFilter | Xác định cột tương ứng có thêm tính năng lọc dữ liệu hay không? | “true” |
| allowNulls | Trường không được để trống. | “true” |
| allowSorting | Cho phép sắp xếp tăng dần, giảm dần tại cột tương ứng. | “true” |
| dataFormatString | Dùng để khai báo kiểu format của dữ liệu trên control, nếu kiểu format đã được định nghĩa trong Options.xml thì dùng ký tự @ten\_trong\_option để lấy giá trị format, nếu chưa có thì dùng định nghĩa format của ngôn ngữa VB. |  |
| hidden | Trường này có bị ẩn trong grid hay không? | “false” |
| inactivate | Xác định control được tạo ra có tabstop khi focus bằng phím tab hay không. |  |
| isPrimaryKey | Xác định đây có phải là trường khóa hay không. | Boolean = “false” |
| name | Xác định tên trường của bảng dữ liệu lấy từ database. | Trường không được trống |
| width | Độ rộng của trường hiện tại. |  |

|  |  |  |
| --- | --- | --- |
| **Thẻ <views> trong trong Controller\Grid\.xml > grid** | | |
| Thẻ dùng để chứa khối khai báo các trường nào trong thẻ fields sẽ được hiển thị trong grid | | |
| Thẻ con | Diễn giải | Ghi chú |
| view | Thẻ dùng để chứa khối khai báo cho 1 grid. | Các thẻ view này phân biệt bằng thuộc tính “id”. |
| Thuộc tính của thẻ con | Diễn giải | Giá trị mặc định |
| id | Dùng để khai báo định danh cho grid | String. Không được để trống |
| Thẻ con của thẻ view | Diễn giải | Ghi Chú |
| field | Thẻ dùng để khai báo trường field nào sẽ được hiển thị | Các thẻ field này phân biệt bằng thuộc tính “name”. |

|  |  |  |
| --- | --- | --- |
| **Thẻ <partition> trong trong Controller\Grid\.xml > grid** | | |
| Thẻ dùng để khai bảng master, detail và các thông tin chính khác của chứng từ | | |
| Thuộc tính | Diễn giải | Giá trị mặc định |
| default | Giá trị mặc định để ghép với khai báo ở thuộc tính inquiry và prime. Thường là = “000000” | String. Không được trống |
| expression | Chứa 1 biểu thức xử lý cho dữ liệu của trường được khai báo ở thuộc tính field. Giá trị của biểu thức ở đây sẽ trả về 1 chuỗi để ghép với thuộc tính prime để cho ra bảng dữ liệu, ghép với thuộc tính inquiry để cho ra bảng phục vụ tìm kiếm, phân trang | String |
| field | Khai báo trường dùng để phân chia bảng dữ liệu | String. Không được trống |
| increase | Chứa biểu thức xác định tên bảng tăng theo qui luật nào của trường được khai báo ở thuộc tính field. |  |
| inquiry | Khai báo phần đầu của bảng chứa thông tin phục search và phân trang. Vd có các bảng: i81$000000, i81$201001, i81$201002… thì prime lúc này = “i81$” | String. Không được trống |
| table | Khai báo bảng chứa các khóa để kiểm tra trùng chứng từ. | String. Không được trống |

|  |  |  |
| --- | --- | --- |
| **Thẻ <queries> trong trong Controller\Grid\.xml > grid** | | |
| Thẻ dùng để xử lý cho các sự kiện truy vấn dữ liệu. | | |
| Thẻ con | Diễn giải | Ghi chú |
| query | Thẻ dùng để xác định query cho từng sự kiện nào |  |
| Thuộc tính của thẻ con | Diễn giải | Giá trị mặc định |
| event | Thuộc tính dùng để xác định các khối text SQL bên trong thẻ query này xử lý cho sự kiện nào.Gồm các sự kiện sau:  -Loading: Sự kiện load dữ liệu cho grid  -Finding: Sự kiện load dữ liệu cho tìm kiếm | String. Không được để trống  Dùng cho chứng từ |

|  |  |  |
| --- | --- | --- |
| **Thẻ <toolbar> trong trong Controller\Grid\.xml > grid** | | |
| Thẻ chứa các khai báo cho thanh công cụ phía trên grid. Thông thường thẻ này được dùng chung cho nhiều chứng từ nên được khai báo ở 1 file XML riêng trong thư mục Controller>Include>XML. | | |
| Thẻ con | Diễn giải | Ghi chú |
| button | Thẻ này đại diện cho một nút chức năng trên thanh công cụ.  Thẻ này có một thuộc tính command để xác định đó là chức năng nào, các giá trị là:  -“New” : chức năng thêm.  -“Edit” : chức năng sửa.  -“Delete” : chức năng xóa.  -“Search” : chức năng tìm kiếm.  -“View” : chức năng xem.  -“Print” : chức năng in. |  |

#### Thẻ dir

|  |  |  |
| --- | --- | --- |
| Thẻ <dir> trong Controller\Dir\.xml | | |
| Thẻ gốc chứa toàn bộ thông tin của 1 màn hình xem hoặc nhập liệu thông tin. | | |
| Thuộc tính | Diễn giải | Giá trị mặc định |
| table | Xác định bảng để truy xuất khi thêm mới hoặc sửa. | String  Thuộc tính không được để trống |
| code | Xác định trường khóa của bảng. Có tác dụng xác định trường khóa khi thực hiện chọn và xem thông tin hay khi update thông tin.. | String  Thuộc tính không được để trống |
| order | Khai báo các trường sắp xếp thứ tự của danh mục, các trường cách nhau bởi dấu phẩy. Thuộc tính này chưa có tác dụng gì, nhưng phải khai báo 1 trường nào đó vì để đảm bảo không lỗi. | String  Thuộc tính không được để trống |
| xmlns | Trường xác định xml namspase, dùng để tạo các assist tooltip hỗ trợ để lập trình. | String |
| Thẻ con | Diễn giải | Ghi chú |
| [title](#dir_dir_title) | Thẻ dùng để khai báo các tiêu đề của màn hình xem/thêm/sửa. |  |
| [partition](#dir_dir_partition) | Thẻ dùng để khai bảng master, detail và các thông tin chính khác của chứng từ. |  |
| [fields](#dir_dir_fields) | Thẻ dùng để chứa khối khai báo các trường sẽ được sử dụng. |  |
| [views](#dir_dir_views) | Thẻ dùng để chứa khối khai báo các trường nào trong thẻ fields sẽ được hiển thị trong màn hình. |  |
| [commands](#dir_dir_commands) | Thẻ dùng để chứa các khối lệnh xử lý hỗn hợp và phức tạp. |  |
| [script](#dir_dir_script) | Thẻ dùng để định nghĩa các khối lệnh javacript. |  |
| [response](#dir_dir_response) | Thẻ dùng để định nghĩa các khối lệnh xử lý kết quả trả về cho client. |  |
| [css](#dir_dir_css) | Thẻ dùng để định nghĩa thêm các CSS (Cascading Style Sheets) |  |

|  |  |  |
| --- | --- | --- |
| **Thẻ <fields> trong Controller\Dir\.xml > dir > fields** | | |
| Thẻ dùng để chứa khối khai báo các trường dữ liệu dùng cho grid. | | |
| Thẻ con | Diễn giải | Ghi chú |
| clientScript | Định nghĩa các khối javascript xử lý sự kiện cho control hiện tai. |  |
| footer | Dùng để khai báo các text diễn giải của control. Giá trị của thẻ này thường được dùng chú thích ngay phía sau control, ví dụ như chú thích “0 – Không còn sử dụng, 1 – Còn sử dụng” sau control trạng thái. Thẻ footer có 2 thuộc tính là “v” và “e”. |  |
| header | Dùng để định nghĩa tên trường tiếng Việt, tiếng Anh của trường này, header có 2 thuộc tính tương ứng là “v” và “e”. |  |
| items | Xác định control tạo ra ở dạng nào |  |

|  |  |  |
| --- | --- | --- |
| **Thẻ <items> trong trong Controller\Dir\.xml > dir > fields > field** | | |
| Xác định control tạo ra ở dạng nào | | |
| Thuộc tính | Diễn giải | Giá trị mặc định |
| check | Dùng cho control lookup, thường chứa 1 điều kiện lọc những record nào được valid giá trị khi ngang qua control này. |  |
| controller | Tùy trường hợp mà sử dụng, phụ thuộc vào khai báo của thuộc tính style của thẻ này. Thường dùng để khai báo liên kết đến 1 controller (tức là đến 1 file XML) nào đó trong thư mục lookup, dir, grid … |  |
| information | Dùng để khai báo liên kết giữa trường hiện tại với 1 trường khóa của bảng khác và kèm theo tên của khóa đó. |  |
| key | Tùy trường hợp mà sử dụng, phụ thuộc vào khai báo của thuộc tính style của thẻ này. Thường dùng để khai báo điều kiện lấy dữ liệu cho control hiện tại. Trường này để trống tức là lấy tất cả (điều kiện 1=1) |  |
| new | Dùng để bật chức năng thêm mới/sửa ngay trên màn hình lookup. Giá trị thường là: new = “Default”. |  |
| reference | Tùy trường hợp mà sử dụng, phụ thuộc vào khai báo của thuộc tính style. Trường này xác định field nào (tức control sẽ được tạo ra) sẽ nhận giá trị diễn giải của việc Autocomplete, khi dùng thuộc tính này thì field được liên kết đến phải tồn tại. |  |
| style | Xác định control thuộc dạng nào. |  |

|  |  |  |
| --- | --- | --- |
| **Thẻ <views> trong trong Controller\Dir\.xml > dir** | | |
| Dùng để chứa các định nghĩa về cách hiển thị của các trường field trên màn hình. | | |
| Thẻ con | Diễn giải | Ghi chú |
| view | Thẻ dùng để chứa khối khai báo cho 1 grid. | Các thẻ view này phân biệt bằng thuộc tính “id”. |
| Thuộc tính của thẻ con | Diễn giải | Giá trị mặc định |
| id | Định danh của khối khai báo trong thẻ view này |  |
| height | Chiều cao của màn hình thêm, sửa. |  |
| anchor | Xác định cột nào được phép co giãn trong lúc thay đổi kích thước form nhập liệu. |  |
| Thẻ con của thẻ view | Diễn giải | Ghi chú |
| [item](#dir_dir_views_view_item) | Dùng để khai báo sự hiển thị của từng control như thế nào trong màn hình thêm, sửa. |  |
| [categories](#dir_dir_views_view_categories) | Dùng để khải báo các tab được sử dụng trong màn hình. |  |

|  |  |  |
| --- | --- | --- |
| **Thẻ <Commands> trong trong Controller\Dir\.xml > dir** | | |
| Dùng để chứa các định nghĩa về xử lý như: Loading, Declare, Scrattering, Inserting, Inserted, Updating, Updated, Deleting, Deleted, Checking, Closing và Result. | | |
| Thẻ con | Diễn giải | Ghi chú |
| command | Các command (sự kiện) cụ thể, mỗi command đại diện cho 1 sự kiện bằng cách khai báo thuộc tính event của thẻ command. | Các thẻ view này phân biệt bằng thuộc tính “id”. |

## Danh sách các sự kiện, hàm trong hệ thống

### Các sự kiện:

* Processing: Sử lý store view báo cáo.
* Declare: Khai báo cáo biến sql.
* Loading: Sự kiện Load Form.
* Scattering: Refesh Form.
* InitExternalFields: Khởi tạo các trường giả.
* Closing: Đóng Form.
* Inserting: Trước khi lưu .
* Inserted: Đã lưu.
* Updating: Trước khi sửa.
* Updated: Sau khi sửa .
* Deleting: Trước khi xóa.
* Deleted: Sau khi xóa.

### Các thuộc tính của Form:

* f.getItem(name): Lấy giá trị của đối tượng kiểu số , ngày.
* f.getItem(name).value: Lấy giá trị hoặc gán giá trị cho đối tượng(thường dùng cho đối tượng kiểu text).
* f.getItemValue(name): Lấy giá trị của đối tượng kiểu số , ngày
* f.setItemValue(name,giatri): Gán giá trị cho đối tượng kiểu số , ngày hoặc text.
* var f = o. parentForm: Lấy đổi tượng cấp trên.
* f.live(o) hoặc f.getItem(name).focus():Focus tới 1 đối tượng trên form.
* f.setItemControlBehavior(name, result[0].Value, ''): Gán giá trị trên form theo giá trị sql trả về.
* f.\_action: Lấy sự kiện trên form (New, Edit, View).
* f.setReferenceKeyFilter(name): Gọi lại lookup của 1 trường.
* f.\_tabContainer.\_activeTabIndex: Lấy index của tab
* f.setReadOnlyFields(danh sách trường): Thiết lập readOnly cho 1 trường , loại bỏ nút tích lookup.
* f.\_fields[i].Name: Lấy tên đối tượng khai báo trên form.
* f = g.get\_element().parentForm: Lấy thông tin trên form từ grid.
* f.\_tabContainer.set\_activeTabIndex(tabIndex): Set lai tab active (tính từ 0).
* f.\_tabContainer.\_activeTabIndex: Lấy giá trị của tab hiện thời.
* f.request('Type', 'Type', ['loai'], o): Gửi request trên form.

### Các thuộc tính của Grid:

* g = f.getItem(name).\_controlBehavior: Lấy đối tượng grid trên Form.
* g.\_getColumnOrder(name): Để lấy vị trị của cột theo theo tên.
* g. \_getItem(i, j): Lấy đối tượng cell trên lưới.
* g.\_getItemValue(i, j): Lấy giá trị theo dòng, cột trên lưới.
* o.grid.request(o, 'Hang\_muc', 'Hang\_muc', ['ma\_vv', 's1'], null, true): Gửi request của 1 đối tượng trên lưới
* g.\_getRow(i): Trả về mảng các giá trị của dòng i.
* g.\_subTitle: Tiêu đề phụ của grid.
* g.sum(name): Tính tổng giá trị của cột

## Các công cụ sử dụng:

* ToolF1V2: Hỗ trợ khi gen update cho triển khai.
* FSI Studio: Hỗ trợ tạo command , chỉnh sửa , truy vấn SQL.
* Visual Studio 2019: Công cụ lập trình trên FBO.
* Crytal Report 2008: Tạo mới , chỉnh sửa báo cáo trên FBO.
* MSSQL 2019: Hỗ trợ trong việc bắt Profiler, và debug code liên quan đến thêm, sửa, xóa dữ liệu.

# TRIỂN KHAI

## Lập trình danh mục

## Lập trình chứng từ

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Đề tài đã thực hiện được …

# TÀI LIỆU THAM KHẢO

* + - 1. Tên tác giả, Tên tác giả… *Tên công trình*. Nơi xuất bản, năm, trang
      2. Tên tác giả, Tên tác giả… *Tên công trình*. Nơi xuất bản, năm, trang

# PHỤ LỤC