LỜI CẢM ƠN

Trước hết em xin chân thành thầy Ngô Trường Giang là giáo viên hướng dẫn em trong suốt quá tình thực tập và làm đề tài tốt nghiệp. Thầy đã giúp em rất nhiều và đã cung cấp cho em nhiều tài liệu quan trọng phục vụ cho quá trình tìm hiểu về đề tài "Tìm hiểu về phần mềm mã nguồn mở GreenStone".

Thứ hai, Em xin chân thành cảm ơn các thầy cô trong bộ môn công nghệ thông tin đã chỉ bảo bảo em trong quá trình học và rèn luyện trong 4 năm học vừa qua. Đồng thời em cảm ơn các bạn sinh viên lớp CT901 đã gắn bó với em trong quá trình rèn luyện tại trường.

Cuối cùng em xin chân thành cảm ơn ban giám hiệu trường Đại Học Dân Lập Hải Phòng đã tạo điều kiện cho em có kiến thức, thư viện của trường là nơi mà sinh viên trong trường có thể thu thập tài liệu trợ giúp cho bài giảng trên lớp. Đồng thời các thầy cô trong trường giảng dạy cho sinh viên kinh nghiệm cuộc sống. Với kiến thức và kinh nghiệm đó sẽ giúp em cho công việc và cuộc sống sau này.

Em xin chân thành cảm ơn!

Hải Phòng, tháng 7 năm 2009.

Sinh viên

Vũ Thị Thu Trang

Mở đầu

Trong thời đại Internet lượng thông tin bùng nổ, con người đã đặt ra những yêu cầu trong việc tiếp nhận và quản lý thông tin. Đó là phải tìm kiếm nhanh chóng, thuận tiện, đơn giản đối với người cần tìm kiếm thông tin, phải dễ dàng xây dựng và phân phối đối với người quản lý thông tin và phải tiết kiệm không gian lưu trữ.

Em nhận thấy phần mềm mã nguồn mở Greenstone thỏa mãn được những yêu cầu trên đối với thông tin. Chính vì vậy em đã thực hiện đề tài này với mục đích, hiểu rõ về phần mềm mã nguồn mở GreenStone và khai thác được phần mềm này để ứng dụng vào sử dụng tại trường Đại học Dân lập Hải Phòng.

Đồ án được chia làm 5 chương: Chương 1 đưa ra một cái nhìn tổng quan về GreenStone. Chương 2 đề cập đến vấn đề xây dựng bộ sưu tập. Hiệu chỉnh giao diện và hệ thống web của GreenStone được trình bày trong chương 3 và 4. Chương 5 là phần ứng dụng với việc xây dựng một bộ sưu tập cụ thể và hiệu chỉnh giao diện web cho phù hợp để sử dụng tại Đại học Dân lập Hải Phòng, và cuối cùng là kết luận.

Mục lục

CHƯƠNG 1: Tổng quan về GreenStone

1.1. Thư viện và thư viện số

1.1.1. Giới thiệu

Thư viện là kho tàng tri thức đã có những đóng góp lớn cho sự phát triển của nhân loại. Tuy nhiên, trong thời đại của Internet, thời đại của mạng thông tin toàn cầu, lượng thông tin vận hành ngày càng lớn và người ta cần tìm ra cách quản lí thông tin cho hiệu quả với tiêu chí hàng đầu là tiết kiệm không gian lưu trữ và tìm kiếm thông tin nhanh thì thư viện truyền thống với việc lưu trữ chủ yếu là trên giấy liệu có còn là giải pháp tối ưu? Hơn thế, con người muốn tại bất cứ nơi đâu người ta cũng có thể lấy được thông tin trên khắp thế giới, và thư viện số(digital library) đã ra đời.

1.1.2. Thư viện số

Theo định nghĩa của Akscyn và Witten(Trường Đại học Waikato – NewZealand) thư viện số là tập hợp các bộ sưu tập số của các đối tượng kĩ thuật bao gồm văn bản, hình ảnh, video, âm thanh cho phép:

- Truy cập, hiển thị và chọn lọc tài nguyên số (dành cho độc giả).
- Xây dựng, tổ chức và lưu hành (dành cho cán bộ thư viện).

Hiện nay, trên thế giới có rất nhiều hệ thống thư viện số, ngay tại Việt Nam cũng có một số công ty cung cấp các sản phẩm phần mềm loại này (phần mềm Libol mà Trường Đại học Dân lập Hải Phòng là một trong số đó). Tuy nhiên, để tạo sự liên kết giữa các hệ thống thư viện cần phải có một hệ thống thư viện số thỏa mãn những tiêu chuẩn quốc tế.

1.2. Thư viện số GreenStone

1.2.1. Giới thiệu

Đứng trước yêu cầu thực tế, năm 1995, một nhóm giảng viên và sinh viên trường Đại học Waikato – NewZealand đã xây dựng phần mềm thư viện số GreenStone. Thấy được nghĩa và tác dụng, tháng 8 năm 2000, UNESCO và Human Info NGO đã tham gia hỗ trợ và phát triển GreenStone. GreenStone là bộ phần mềm giúp người sử dụng dễ dàng xây dựng và phân phối bộ sưu tập thư

viện số, nó cung cấp phương pháp mới để tổ chức thông tin và xuất bản thông tin trên Internet và qua CD ROM. GreenStone là phần mềm mã nguồn mở mang tính quốc tế được cung cấp trên http://www. greenstone. org với mục đích cung cấp cho các trường Đại học, thư viện và các viện nghiên cứu xây dựng các bộ sưu tập cho riêng mình.

1.2.2. Đặc điểm

Các đặc điểm nổi bật của GreenStone:

- Truy cập qua trình duyệt web, cả ở chế độ cục bộ (local) và từ xa (remote).
- Chạy được trên nhiều hệ điều hành (multiplatform): Windows, Linux, Sun Solaris, Macintosh, …
- Tìm kiếm toàn văn bản và tìm kiếm theo từng trường riêng biệt.
- Tận dụng các metadata sẵn có trong tài liệu, giúp người tạo lập bộ sưu tập không phải làm bằng tay.
- Khả năng linh động, dễ mở rộng hệ thống nhờ các thành phần như plugin, classifier.
- Hỗ trợ xử lý tài liệu với nhiều ngôn ngữ.
- Cung cấp giao diện đa ngôn ngữ.
- Ngoài các bộ sưu tập văn bản, hình ảnh thông thường, GreenStone còn cho phép tạo các bộ sưu tập hình ảnh, âm thanh đa phương tiện.
- Xây dựng bộ sưu tập đơn giản, có hiệu quả.
- Khả năng xuất bản các bộ sưu tập ra CD với đầy đủ tính năng có thể tự cài đặt và chạy độc lập.
- Các bộ sưu tập dễ dàng mang chuyển, phân phối, chia sẻ.

1.3. Một số khái niệm cơ bản

1.3.1. Tài liệu

GreenStone hỗ trợ các loại tài liệu dạng HTML, XML, TXT và các dạng phức tạp như Word hoặc dạng đang được sử dụng phổ biến trên nhiều môi trường như PDF, multi-media, ….

1.3.2. Bộ sưu tập

Một thư viện số do GreenStone tạo ra chứa được nhiều bộ sưu tập. Bộ sưu tập có thể xem là đơn vị của một thư viện số GreenStone. Mỗi bộ sưu tập tập trung vào một vấn đề nào đó. Ví dụ, bộ sưu tập Sách, bộ sưu tập Tạp chí, … Các bộ sưu tập có thể được bổ sung cập nhật, kích thước các bộ sưu tập có thể lên đến hàng Gigabyte dữ liệu.

1.3.3. Tìm kiếm

Các bộ sưu tập cho phép tìm kiếm trên toàn bộ nội dung văn bản hoặc có thể tìm kiếm trên từng đoạn. Cũng có thể tìm kiếm theo các từ khóa, các cụm từ và kết quả sẽ được sắp xếp theo thứ tự yêu cầu của câu truy vấn.

1.3.4. Duyệt tài liệu

GreenStone cho phép định nghĩa trước các cấu trúc để duyệt tài liệu trong mỗi bộ sưu tập dựa trên những metadata tìm thấy trong bộ sưu tập đó. Ví dụ như duyệt theo “đề mục” hoặc những tài liệu nào phân cấp theo mục lục thì ta có thể duyệt theo chính “mục lục” đó, …

1.3.5. MetaData

Là thông tin mô tả cho một tài liệu trong bộ sưu tập, ví dụ tên tài liệu, nhà xuất bản, tác giả, … GreenStone dùng các thẻ XML để mô tả thông tin cho tài liệu. Ví dụ:

<Metadata name=”Title”>Tìm hiểu phần mềm GreenStone</Metadata>

Các thẻ này có thể:

- Được nhúng trong tài liệu của bộ sưu tập (ví dụ như các thẻ HTML trong tài liệu HTML).

- Được lưu thành tệp tin Metadata kèm theo tài liệu.
- Được trích một cách tự động từ một tài liệu nào đó, ví dụ thông tin về tên, kích thước, ngày tạo, ngày hiệu chỉnh, … tập tin tài liệu.

1.3.6. Biên mục

Biên mục là khái niệm của nghiệp vụ thư viện để chỉ hành đọng cung cấp thông tin mô tả cho các tài liệu trong thư viện. Hiện nay người ta thường biên mục tài liệu theo chuẩn quốc tế Dublin Core.

1.3.7. Plugin

1.3.7.1. Giới thiệu

Plugin là một chương trình con được dùng trong quá trình xây dựng bộ sưu tập.

Do nguồn vào có nhiều dạng tài liệu khác nhau (pdf, word, text, …) nên cần plugin để chuyển chúng về một loại thống nhất là XML của GreenStone và trích thông tin từ tài liệu nguồn đưa vào tập tin XML này.

Mỗi bộ sưu tập có một tập tin cấu hình collect. cfg. Tập tin này liệt kê các Plugin được dùng trong quá trình xây dựng bộ sưu tập. Tùy theo tài liệu nguồn có dạng nào thì ta sẽ chọn Plugin tương ứng. Ví dụ tài liệu nguồn là tập tin word thì ta dùng Plugin WordPlug.

Các Plugin được viết bằng ngôn ngữ Perl. Mọi plugin đều được kế thừa từ plugin cơ sở BasPlug. Plugin cơ sở BasPlug thực hiện những thao tác cơ bản như tạo tài liệu mới XML theo định dạng của Greenstone, gán định danh cho tài liệu. Các plugin được đặt trong thư mục “greenstone\perllib\plugins”.

Ta có thể tìm hiểu thông tin của plugin hoặc viết plugin mới.

1.3.7.2. Danh sách các Plugin

Bảng 1.1 – Danh sách các Plugin.

Tên Plugin	Công dụng	Kiểu tập tin xử lý	Các tập tin không xử lý
BasPlug	Là lớp cơ sở cho tất cả các plugin		

ConvertToPlug	Gọi các chương trình để chuyển các tài liệu độc quyền sang HTML hay plain text		
ArcPlug	Xử lýcác tập tin chỉ ra trong tập archives. inf, tập tin archives. inf là cầu nối giữa tiến trình import và tiến trình build. Plugin này bắt buộc phải khai báo trong tập tin cấu hình.		
RecPlug	Duyệt qua thư mục để xử lý các tập tin mà plugin này tìm thấy.		
GAPlug	Xử lý các tập tin của Greenstone được phát sinh từ chương trình import. pl	. xml	
TextPlug	Xử lý tập tin text thuần túy.	. txt,. Text	
HTMLPlug	Xử lý tập tin HTML	. htm,. html,. cgi,. php,. asp,. shm,. Shtml	. gif,. jpg,. jpeg,. png,. css,. rtf
WordPlug	Xử lý tập tin word	. doc	. gif,. jpg,. jpeg,. png,. css,. rtf
PDFPlug	Xử lý tập tin PDF	. pdf	. gif,. jpg,. jpeg,. png,. css,. rtf
PSPlug	Xử lý tài liệu postscript, trích thông tin metadata ngày, tựa đề, số trang, …	. ps	. eps
EMAILPlug	Xử lý thông điệp email, trích thông tin như tác giả, ngày, chủ đề, …	Tập tin kết thúc bằng số hoặc số theo sau là. Email	
BibTexPlug	Xử lý các tập tin bibliography theo chuẩn Bib Tex	. bib	
ReferPlug	Xử lý các tập tin bibliography theo chuẩn Refer	. bib	
SRCPlug	Xử lý các tập tin mã nguồn	Makefile,	. o,. obj,. a,.

		Readme,. c,. cc,. cpp,. h,. hpp,. pl,. pm,. Sh	so,. dll
ImagePlug	Xử lý các tập tin ảnh. Plugin này chỉ dùng trên Unix	. gif,. jpg,. jpeg,. png,. bmp,. xbm,. tif,. Tiff	
SplitPlug	Giống BasPlug và ConvertToPlug. Không dùng trực tiếp plugin này, plugin này phải được thừa kế để xử lý tài liệu.		
FoxPlug	Xử lý các tập tin FoxBase	. dbt,. Dbf	
ZipPlug	Xử lý các tập tin nén	. gzip,. bzip,. tar,. zip,. gz,. bz,. tgz,. Taz	

1.3.7.3. Các Plugin xử lý tài liệu độc quyền

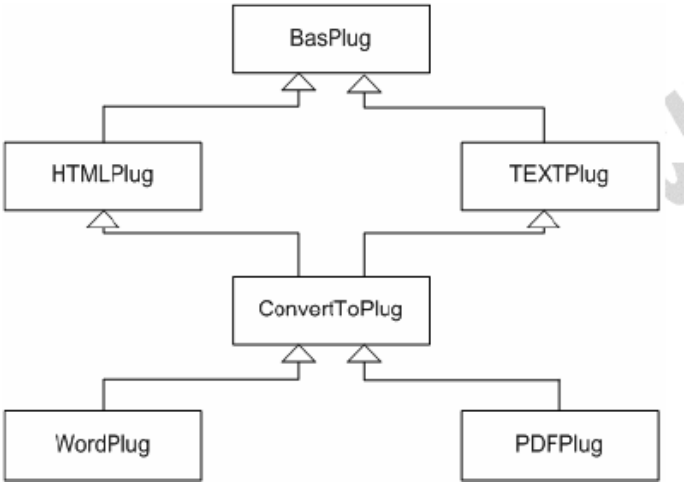
Đối với tài liệu độc quyền như word, pdf, ta dùng các plugin tương ứng là WordPlug và PDFPlug. Các plugin này thực hiện 2 thao tác:

1. Chuyển tài liệu nguồn sang dạng html hay plain text

2. Sử dụng plugin HTMLPlug hay TEXTPlug chuyển kết quả ở bước 1 sang dạng XML của Greenstone.

Để chuyển tài liệu nguồn sang dạng html hay plain text, Greenstone dùng những chương trình có sẵn như pdftohtml, wvware trong thư mục "greenstone\bin\windows".

Các plugin WordPlug và PDFPlug kế thừa từ plugin ConvertToPlug. Tùy chọn convert_to của plugin ConvertToPlug cho biết chuyển sang dạng tài liệu nào.



Hình 1. 1 – Cây kế thừa của các plugin xử lý các tài liệu độc quyền

1.3.7.4. Gán thông tin metadata từ một tập tin mô tả

Các thông tin metadata cho một tài liệu có thể được đặc tả trong một tập tin XML metadata. xml. Nếu tùy chọn use_metadata_files của plugin RecPlug được chỉ ra, plugin này sẽ gán thông tin metadata có trong tập tin metadata. xml vào tập tin XML chuyển đổi từ tài liệu nguồn.

```
<!DOCTYPE GreenstoneDirectoryMetadata [
<!ELEMENT DirectoryMetadata (FileSet*)>
<!ELEMENT FileSet (FileName+,Description)>
<!ELEMENT FileName (#PCDATA)>
<!ELEMENT Description (Metadata*)>
<!ELEMENT Metadata (#PCDATA)>
<ATTLIST Metadata name CDATA #REQUIRED>
<ATTLIST Metadata mode (accumulate|override) "override">
]>
```

Hình 1. 2 - Định nghĩa kiểu tài liệu XML của tập tin metadata. Xml

```
<?xml version="1.0" ?>
<!DOCTYPE GreenstoneDirectoryMetadata SYSTEM
"http://greenstone.org/dtd/GreenstoneDirectoryMetadata/1.0/GreenstoneDirecto
ryMetadata.dtd">
<DirectoryMetadata>
    <FileSet>
        <FileName>vidu.*</FileName>
        <Description>
            <Metadata name="Title">Đây là ví dụ</Metadata>
            <Metadata name="Place" mode="accumulate">Sách giáo
khoa</Metadata>
        </Description>
    </FileSet>
    <FileSet>
        <FileName>vidu-1.jpg</FileName>
        <Description>
            <Metadata name="Title">Đây là ví dụ 1</Metadata>
            <Metadata name="Subject">Thư viện số</Metadata>
        </Description>
    </FileSet>
</DirectoryMetadata>
```

Hình 1. 3 -Ví dụ một tập tin metadata. xml

Ví dụ trên chứa 2 cấu trúc metadata. Ở mỗi cấu trúc, trường FileName đặc tả tên các tập tin cần gán thông tin metadata. Ở cấu trúc thứ nhất, thông tin metadata sẽ được gán cho các tập tin được bắt đầu với chữ "vidu". Những thông tin metadata Title và Place có giá trị tương ứng là "Đây là ví dụ", "Sách giáo khoa". Ở cấu trúc thứ hai, metadata Title của tập tin vidu-1. jpg có giá trị "Đây là ví dụ 1" sẽ override thông tin metadata Title đã được đặc tả ở cấu trúc metadata thứ nhất. Tập tin vidu-1. jpg còn được cung cấp thêm metadata Subject với giá trị là "Thư viện số".

Với một metadata có nhiều giá trị phải dùng thuộc tính mode = "accumulate", nếu không metadata đặc tả sau sẽ override metadata trước (giá trị mặc định là mode = "override")

Đối với plugin RecPlug, nếu tùy chọn use_metadata_files được chọn, RegPlug sẽ tìm trong thư mục tài liệu nguồn tập tin metadata. xml, sau đó gán thông tin metadata trong t ập tin metadata. xml này cho các tập tin và thư mục con trong thư mục tài liệu nguồn.

1.3.7.5. Chia cấu trúc tài liệu nguồn

Có thể chia tài liệu nguồn có cấu trúc phân cấp thành nhiều vùng (section), mỗi vùng được bao bằng cặp thẻ <Section> </Section>, các cặp thẻ <Section></Section> có thể lồng nhau.

```
<!--
<Section>
    <Description>
        <Metadata name="Title">( Thông tin metadata mô tả thông tin của
Section)</Metadata>
     </Description>
-->
(Phần nội dung của Section)
<!--
</Section>
-->
```

Hình 1. 4 - Minh họa cách chia section cho tài liệu

Giữa cặp thẻ <Section> </Section> ta có thể thêm cặp thẻ <Description> </Description> để mô tả thông tin cho section. Ví dụ trên mô tả thông tin metadata Title cho section. Ta chỉ có thể thêm các thẻ section vào tài liệu nguồn dạng html, word vì:

- Đối với tập tin html, các văn bản trong vùng <!-- --> được xem như dòng ghi chú, do đó các thẻ <Section> trong tập tin html không ảnh hưởng đến nội dung hiển thị của tập tin html này.
- Đối với tập tin word, việc chuyển sang XML của Greenstone phải qua bước trung gian là chuyển sang tập tin html nên việc chèn các thẻ <Section> cũng không ảnh hưởng đến nội dung hiển thị cuối cùng.

Mục đích của việc thêm cặp thẻ <Section> </Section> vào tài liệu nguồn là để sau khi xây dựng bộ sưu tập, khi hiển thị nội dung của tài liệu bằng trình duyệt web, ta sẽ thấy cấu trúc phân cấp của tài liệu và có thể nhanh chóng xem nội dung một đoạn nào đó trong tài liệu nhờ cấu trúc phân cấp này.

Để plugin HTMLPlug xử lý cặp thẻ <Section> </Section>, ta phải chọn tùy chọn description_tags. Ví dụ một tài liệu có cấu trúc phân cấp:

Tìm hiểu PP LT hướng khía cạnh
Lời cảm ơn
(Nội dung phần "Lời cảm ơn")
...
Mục lục
(Nội dung phần "Mục lục")
...
Danh mục các ảnh-Sơ đồ
(Nội dung phần "Danh mục các ảnh-Sơ đồ")
...
Giới thiệu
(Nội dung phần "Giới thiệu")
...
Nội dung
(Nội dung phần "Nội dung")
...
Phần 1: Tìm hiểu phương pháp lập trình hướng khía cạnh
Chương 1: Tổng quan về đề tài
(Các nội dung trong chương 1)
...
Chương 2: Phương pháp lập trình hướng khía cạnh (AOP)
(Các nội dung trong chương 2)
...
Chương 3: Ngôn ngữ lập trình AspectJ
(Các nội dung trong chương 3)
...

Sau khi xây dựng bộ sưu tập từ tài liệu nguồn đã chèn các cặp thẻ <Section> </Section>, trình duyệt web sẽ hiển thị nội dung tài liệu này như sau:

Hình 1. 5 - Hiển thị nội dung tài liệu trên trình duyệt web

1.3.8. Classifier

Classifier dùng để xây dựng cấu trúc duyệt tài liệu trên web của một bộ sưu tập. Tương tự các plugin, các classifier được đặc tả trong tập tin cấu hình collect. cfg của mỗi bộ sưu tập.

Trong pha cuối cùng của quá trình xây dựng bộ sưu tập (nén và tạo chỉ mục trên tài liệu), các classifier được script buildcol. pl gọi sẽ lưu cấu trúc duyệt tài liệu vào cơ sở dữ liệu bộ sưu tập.

Cú pháp: classify <Tên classifier> <Các tham số>

Ví dụ: classify AZList -metadata Title -buttonname TitleA-Z

Trong dòng đặc tả có một tham số quan trọng là metadata xác định rằng các tài liệu của bộ sưu tập sẽ được sắp xếp theo metadata đã được chỉ ra. Với ví dụ trên, các tài liệu được sắp xếp theo tựa đề của tài liệu (Title)

Tham số buttonname xác định tên nút xuất hiện trên thanh duyệt. Với dòng đặc tả trên, khi ta click vào nút TitleA-Z trên thanh duyệt, các tài liệu của bộ sư u tập được liệt kê theo thứ tự từng vùng alphabet.

Hình 1. 6 - Dùng AZList để liệt kê các tài liệu theo từng vùng alphabet

Các classifier được đặt trong thư mục greenstone\perllib\classify. Để biết thông tin của classifier, dùng lệnh: classinfo. pl <Tên classifier>. Ta có thể viết các classifier mới.

Những nút trên thanh duyệt, ngoại trừ nút Search, được quản lý bởi các classifier. Khi định nghĩa một classifier trong tập tin collect. cfg, những nút liên quan sẽ xuất hiện trên thanh duyệt.

1.3.8.1. Phân loại

Nhóm classifier liệt kê tài liệu dưới dạng danh sách (list)

Classifier AZList : liệt kê tài liệu theo từng vùng alphabet

Hình 1. 7 - Minh họa classifier AZList

Classifier List: liệt kê tài liệu thành một danh sách sắp thứ tự alphabet

Hình 1. 8 - Minh họa classifier List

Classifier DateList: liệt kê tài liệu theo từng vùng thời gian

Hình 1. 9 - Minh họa classifier DateList

Nhóm classifier liệt kê tài liệu dưới dạng phân cấp (hierarchy)

Classifier Hierarchy : liệt kê các tài liệu dưới dạng phân cấp

Hình 1.10 - Minh họa Classifier Hierarchy

1.3.9. Định dạng cách hiển thị tài liệu

Những trang web trong Greenstone không được thiết kế trước mà được phát sinh và hiển thị ra web browser. Một phần giao diện của các trang web này được quản lý bởi các chuỗi định dạng. Chuỗi định dạng được đặc tả trong tập tin cấu hình của bộ sưu tập collect. cfg. Để đặc tả chuỗi định dạng ta dùng từ khóa format, theo sau là tên của những thành phần mà chuỗi định dạng sẽ tác động.

Ta có thể định dạng 2 thành phần sau:

- Danh sách tài liệu được phát sinh bởi classifier hoặc danh sách tài liệu nhận được trong quá trình tìm kiếm

- Những thành phần trên trang web hiển thị tài liệu hoặc hiển thị các đoạn của một tài liệu

1.3.9.1. Định dạng danh sách tài liệu

Cú pháp: format <kiểu danh sách> <chuỗi html định dạng>

Từ khóa chỉ kiểu danh sách gồm 2 phần:

Phần thứ nhất gồm các loại:

- Search: danh sách kết quả tìm kiếm tài liệu
- CLi: với i là số nguyên > 0

Đây là danh sách được phát sinh bởi các classifier. CL1, CL2, CL3 … tương ứng với classifier thứ nhất, thứ hai, thứ ba... được đặc tả trong tập tin cấu hình collect. cfg.

Phần thứ hai gồm các loại:

- VList: danh sách theo chiều dọc
- HList: danh sách theo chiều ngang
- DateList: danh sách phân loại theo thời gian

Ví dụ :

format SearchVList … : định dạng kế t quả tìm kiếm tài liệu, áp dụng cho các danh sách hiển thị theo chiều dọc.

format CL1HList… : định dạng danh sách tài liệu phát sinh từ classifier thứ nhất, áp dụng cho các danh sách hiển thị theo chiều ngang

Ví dụ một đoạn trong tập tin cấu hình collect. cfg :

```
1  classify Hierarchy -metadata Subject -buttonname Subjects
2  classify AZList -metadata Title –buttonname Title
3  classify List -metadata Howto –buttonname HowTo
4  format CL3Vlist "<br>[link][Howto][/link]"
5  format DocumentImages true
6  format DocumentText "<h3>[Title]</h3>\\n\\n<p>[Text]"
7  format DocumentButtons "Expand Text|Expand contents|Detach|Highlight"
```

Kết quả hiển thị trên trình duyệt web :

Hình 1. 11 - Kết quả hiển thị tài liệu trên trình duyệt

Ta thấy ở dòng 4 trong tập tin cấu hình có nội dung:

format CL3VList “
[link][Howto][/link]”

Dùng để định dạng danh sách tài liệu được phát sinh từ classifier thứ ba trong tập tin cấu hình (classifier List), áp dụng cho các danh sách được bố trí theo chiều dọc. Chuỗi "
[link][Howto][/link]" là chuỗi html định dạng cách hiển thị của danh sách. Với chuỗi định dạng này thì mỗi phần tử trong danh sách sẽ xuất hiện trên một dòng (“
”), được đại diện bằng dòng chữ (giá trị của metadata Howto) và liên kết đến tài liệu gốc (“[link] [/link]”). Trong chuỗi định dạng ta có thể sử dụng các thẻ html và một số từ khóa khác mà Greenstone hỗ trợ, ví dụ [link] [/link] đại diện cho 1 liên kết, [tên metadata] đại diện cho giá trị của metadata, [Text] đại diện cho nội dung của văn bản…

1.3.9.2. Định dạng các thành phần của trang web hiển thị tài liệu

Cú pháp: format <tên thành phần> <giá trị>

Dưới đây là bảng các thành phần trên trang web

Bảng 1.2 – Các thành phần trên trang Web

Tên thành phần	**Giá trị**	**Ý nghĩa**
DoccumentImages	True/false	True: hiển thị ảnh ở phía trên bên trái của trang tài liệu. False: không hiển thị. Giá trị mặc định là false.
DoccumentHeading	Chuỗi định dạng	Định dạng phần header của tài liệu trong trang tài liệu nếu DoccumentImages có giá trị false giá trị mặc định là: [Title]
DoccumentContents	True/false	Hiển thị bảng nội dung tài liệu được phân cấp hoặc các nút next/previous và đoạn chữ "page k of n" nếu tài liệu không phân cấp.
DoccumentButtons	Chuỗi	Quản lý các nút hiển thị trên trang tài liệu. Giá trị mặc định Detach| HighLight
DoccumentText	Chuỗi định dạng	Định dạng nội dung hiển thị trên trang tài liệu. Giá trị mặc định: <table width=537><tr>[Text]<td></td></tr > </table>
DoccumentArrowsBottom	True/false	Hiển thị nút next/previous trên trang tài liệu Giá trị mặc định: true
DoccumentUseHTML	True/false	True: mỗi tài liệu được hiển thị trong

		một frame. False: không hiển thị tài liệu dưới dạng frame.

Ví dụ:

Dòng 6 trong tập tin cấu hình trên định dạng cách hiển thị nội dung tài liệu:

format DocumentText “<h3>[Title]</h3>\\n\\n<p>[Text]”

Dòng 7 ta xác định các nút dùng trong trang tài liệu:

format DocumentButtons " Expand Text|Expand

contents|Detach|Highlight "

Ta có thể xem hình 1. 11 ở trên để thấy rõ hơn kết quả mà chuỗi định dạng mang lại.

CHƯƠNG 2: Xây dựng bộ sưu tập

2.1. Tổng quan quá trình xây dựng bộ sưu tập

Quá trình xây dựng một bộ sưu tập trải qua 3 pha chính

Pha 1 : Tạo cấu trúc chung cho bộ sưu tập

Pha 2 : Chuyển định dạng tài liệu nguồn sang định dạng XML

Pha 3 : Nén và tạo chỉ mục trên các tài liệu của bộ sưu tập

Ở mỗi pha ta dùng chương trình do Greenstone hỗ trợ để xây dựng bộ sưu tập

Pha 1 : dùng chương trình mkcol. pl

Pha 2 : dùng chương trình import. pl

Pha 3 : dùng chương trình buildcol. pl

Các chương trình trên được đặt trong thư m ục "greenstone\bin\script". Để thực thi các chương trình này trong môi trường DOS, ta dùng cú pháp lệnh như sau: perl –S <tên chương trình><các tham số>

2.1.1. Chương trình mkcol. pl

Công dụng :

Chương trình mkcol. pl dùng để tạo cấu trúc chung cho một bộ sưu tập, tạo tập tin cấu hình mặc định cho bộ sưu tập collect. cfg đặt trong thư mục con "etc" của bộ sưu tập.

Cú pháp :

mkcol. pl [Các tùy chọn] <Tên bộ sưu tập>

Các tùy chọn:

- creator <string>: địa chỉ email của người tạo bộ sưu tập
- optionfile <string>: lấy những tùy chọn từ một tập tin nào đó.
- maintainer <string>: địa chỉ email của người quản lý bộ sưu tập.

- collectdir <string>: thư mục chứa bộ sưu tập. Giá trị mặc định là "greenstone\collect"
- public <string>: cho phép bộ sưu tập được truy cập rộng rãi hay không. Giá trị mặc định là "true".
- title <string>: tựa đề của bộ sưu tập
- about <string>: thông tin mô tả bộ sưu tập
- plugin <string>: tên plugin được dùng
- quiet : không hiển thị các thông báo của chương trình
- win31compat <string> : cho biết tên thư mục của bộ s ưu tập có tuân theo quy ước của Windows 3. 1 hay không (tên thư mục có độ dài tối đa 8 kí tự). Giá trị mặc định là "true".

Hình 2. 1 - Quá trình xây dựng bộ sưu tập

2.1.2. Chương trình import. pl

Công dụng:

Chuyển định dạng tài liệu nguồn sang định dạng XML của Greenstone, tạo tập tin tóm tắt thông tin archive.inf.

Cú pháp :

import. pl [Các tùy chọn] <Tên bộ sưu tập>

Các tùy chọn :

- archivedir <string>: đường dẫn đến các tập tin sau khi import, mặc định là " greenstone\collect\<tên thư mục chứa bộ sưu tập>\archives"
- collectdir <string>: thư mục chứa các bộ sưu tập, mặc định là "greenstone\collect"
- debug: chạy chương trình ở chế độ debug, chỉ xuấ t kết quả ra màn hình, không tạo các tập tin kết quả sau khi import
- faillog <string>: đường dẫn đến tập tin log lưu tên của những tập tin không import được. Mặc định là "greenstone\collect\<tên thư mục chứa bộ sưu tập>\etc\fail. log"
- groupsize <int>: số tài liệu được nhóm thành một tập tin XML, mặc định là 1
- gzip: dùng gzip để nén những tài liệu XML kết quả. Chú ý phải thêm plugin ZIPPlug vào danh sách các plugin trong tập tin cấu hình.
- importdir <string>: đường dẫn đến các tập tin nguồn
- keepold : không xóa nộ i dung của thư mục archive (mặc định)
- maxdocs <int>: số tài liệu tối đa được import.
- OIDtype <enum>: phương thức dùng để phát sinh ra ID duy nhất cho mỗi tài liệu. Giá trị mặc định là hash. Các giá trị có thể là hash, incremental, assigned, dirname.
- out <string>: tên tập tin hoặc handle để in ra các dòng thông báo. Giá trị mặc định là STDERR.

- removeold: xóa những nội dung cũ của thư mục archives.
- saveas <enum> : định dạng của tập tin sau khi import. Mặc định là GA.

GA: định dạng theo Greenstone

METS: định dạng theo METS

- sortmeta <metadatum>: sắp xếp những tài liệu theo thứ tự alphabet của
- metadata. Tùy chọn này sẽ bị bỏ qua nếu groupsize >1
- statsfile <string>: tên tập tin hay handle để in ra các dòng thống kê của quá trình import. Mặc định là STDERR
- verbosity <int>: quản lý mật độ xuất các thông báo ra màn hình.

Các giá trị : 0: không xuất, 3: nhiều. Giá trị mặc định là 2.

2.1.3. Chương trình buildcol. pl

Công dụng :

Nén văn bản, tạo chỉ mục trên tài liệu XML, lưu thông tin thể hiện bộ sưu tập vào cơ sở dữ liệu (icon, tiêu đề, thông tin classifier tạo ra…).

Cú pháp :

buildcol. pl [Các tùy chọn] <Tên bộ sưu tập>

Các tùy chọn :

- remove_empty_classifications : giấu đi những classifier và những nút phân cấp rỗng (chúng không chứa những tài liệu nào).
- archivedir <string>: đường dẫn đến thư mục archives.
- builddir <string>: đường dẫn đến thư mục building chứa các chỉ mục đã được tạo.
- collectdir <string> : đường dẫn thư mục chứa các bộ sưu tập, mặc định là " greenstone\collect".

- debug: chạy chương trình ở chế độ debug, chỉ xuất các kết quả ra màn hình, không tạo ra các tập tin kết quả.
- faillog <string>: đường dẫn đến tập tin log, mặc định là "greenstone\collect\<tên thư mục chứa bộ sưu tập>\etc\fail. log"
- index <string>: xác định loại chỉ mục sẽ được xử lý. Nếu tùy chọn này không được chọ n thì các chỉ mục trong tập tin cấu hình collect. cfg sẽ được xử lý.
- keepold: không xóa nội dung hiện tại ở thư mục building.
- maxdocs <int>: số tài liệu tối đa được xử lý.
- mode <enum>: chỉ ra các công việc được thực hiện trong quá trình building, giá trị mặc định là all.

Một số giá trị:

all: xử lý tất cả các công việc

compress_text: chỉ nén văn bản

build_index: chỉ tạo chỉ mục cho văn bản

infodb:chỉ xây d ựng cơ sở dữ liệu metadata

- no_text: không lưu những văn bản được nén.
- out <string>: tên tập tin hoặc handle để xuất những thông báo tình trạng, mặc định là STDERR.
- verbosity <int> quản lý mật độ xuất những thông báo

Một số giá trị thường dùng:

0: không xuất thông báo

3: xuất đầy đủ các thông báo

Giá trị mặc định là 2

2.2. Cấu trúc thư mục của Greenstone

Ta gọi thư mục cài đặt Greenstone là GSDLHOME. Cấu trúc thư mục của Greenstone như sau:

Hình 2. 2 – Cấu trúc thư mục của Greenstone

Bảng 2.1 – Danh sách thư mục của GreenStone

Tên thư mục	**Mục đích**
bin	Chứa mã thực thi
bin\script	Chứa các script Perl để tạo bộ sưu tập
perllib	Chứa những module Perl hỗ trợ cho quá trình xây dựng bộ sưu tập
Perllib\plugin	Chứa mã nguồn của các plugin hỗ trợ xử lý tài liệu
Perllib\classify	Chứa mã nguồn các claasify hỗ trợ việc hiển thị kết quả tìm kiếm tài liệu
cgi-bin	Chứa các CGI script của GreenStone
tmp	Chứa các tập tin tạm của GreenStone
etc	Chứa các tập tin cấu hình, tập tin log, cơ sở dữ liệu quản lý người dùng
src	Chứa mã nguồn C++
src/colservr	Chứa mã nguồn C++
src/recpt	Chứa mã nguồn C++
packages	Chứa mã nguồn của những gói phần mềm hỗ trợ cho

	GreenStone
Packages\mg	Chứa mã nguồn của MG – phần mềm dùng để nén và tạo chỉ mục trong GreeStone
mappings	Chứa bảng chuyển đổi chuẩn Unicode thành các chuẩn khác
macros	Chứa các tập tin macro dùng cho giao diện GreenStone
collect	Chứa các bộ sưu tập
lib	Chứa mã nguồn C++ dùng cho collection server và receptionist
images	Chứa các tập tin ảnh dùng cho giao diện của GreenStone
docs	Chứa các tài liệu về GreenStone

2.3. Cấu trúc thư mục của một bộ sưu tập

Trong pha 1 của quá trình xây dựng bộ sưu tập, sau khi thực thi chương trình mkcol. pl, bộ sưu tập được tạo ra với cấu trúc các thư mục như sau:

Bảng 2.3 – Danh sách thư mục của bộ sưu tập

Tên thư mục	**Mục đích**
archives	Chứa các tập tin sau khi import
building	Chứa các tập tin trong quá trình nén, tạo chỉ mục, cơ sở dữ liệu cho bộ sưu tập
etc	Chứa tập tin cấu hình collect.cfg
images	Chứa các ảnh dành riêng cho bộ sưu tập

import	Chứa các tài liệu nguồn cần xây dựng bộ sưu tập
index	Chứa các tập tin sau khi nén, tạo chỉ mục, cơ sở dữ liệu lấy từ thư mục building
perllib	Chứa các thư viện Perl hôc trợ cho bộ sưu tập.

2.4. Cấu trúc tài liệu theo định dạng XML

Trong pha import, Greenstone chuyển tài liệu nguồn sang tài liệu XML. Dưới đây là phần định nghĩa kiểu tài liệu XML của Greenstone (DTD – Document Type Definition)

```
<!DOCTYPE GreenstoneArchive [
<!ELEMENT Section (Description, Content, Section*)>
<!ELEMENT Description (Metadata*)>
<!ELEMENT Content (#PCDATA)>
<!ELEMENT Metadata (#PCDATA)>
<ATTLIST Metadata name CDATA #REQUIRED>
]>
```

Ví dụ một tài liệu của Greenstone sau khi import:

```
<?xml version="1. 0" ?>
<!DOCTYPE GreenstoneArchive SYSTEM
"http://greenstone. org/dtd/GreenstoneArchive/1. 0/GreenstoneArchive. dtd">
<Section>
 <Description>
 <Metadata name="gsdlsourcefilename">ec158e. txt</Metadata>
 <Metadata name="Title">Freshwater Resources in Arid Lands</Metadata>
 <Metadata
name="Identifier">HASH0158f56086efffe592636058</Metadata>
 <Metadata name="gsdlassocfile">cover. jpg:image/jpeg:</Metadata>
```

```
<Metadata name="gsdlassocfile">p07a. png:image/png:</Metadata>
</Description>
<Section>
<Description> <Metadata name="Title">Preface</Metadata> </Description>
<Content> This is the text of the preface </Content>
</Section>
<Section>
<Description>
<Metadata name="Title">First and only chapter</Metadata>
</Description>
<Section>
<Description> <Metadata name="Title">Part 1</Metadata> </Description>
<Content> This is the first part of the first and only chapter </Content>
</Section>
<Section>
<Description> <Metadata name="Title">Part 2</Metadata> </Description>
<Content> This is the second part of the first and only chapter </Content>
</Section>
</Section>
</Section>
```

Tài liệu XML của Greenstone có một thẻ gốc là <Section> </Section>. Tài liệu được chia thành nhiều vùng (section) được bao bọc bởi cặp thẻ <Section> </Section>, các cặp thẻ <Section> </Section> có thể lồng nhau. Mỗi Section có một thẻ Description và một thẻ Content. Thẻ Decription có thể chứa một hay nhiều thẻ Metadata. Các thẻ metadata có cấu trúc :

<Metadata name = “Tên metadata”>Giá trị của metadata</metadata>

Ta thường biên mục tài liệu theo chuẩn Dublin Core, ví dụ :

<Metadata name = “dc. Title”>Tìm hiểu nguồn mở Greenstone</Metadata>

dc là từ viết tắt của cụm từ Dublin Core

Greenstone thiết kế sẵn một số bộ thẻ để biên mục, với Dublin Core ta có bộ thẻ dc. Nếu không có metadata nào trong các chuẩn của Greenstone thích hợp để mô tả tài liệu, có thể dùng các bộ metadata do ta định nghĩa. Ví dụ có thể mô tả cho phần tiêu đề của một cuốn sách như sau:

<Metadata name = “BookTitle”>Lập trình C++</Metadata>

Mỗi tài liệu trong Greenstone có một ID nhất định được hệ thống phát sinh (OID – Object Identifier) để xác định những section hay section con bằng cách đánh số các section này. Ví dụ : section con thứ ba của section thứ hai của tài liệu có OID = HASHa72X là HASHa72X. 2. 3

Hình 2. 4 – Minh họa cấu trúc phân cấp của tài liệu

Cấu trúc phân cấp của tài liệu được dùng cho chỉ mục tìm kiếm tài liệu, có 3 mức chỉ mục: document, section, paragraph.

Chỉ mục document : tìm kiếm một số từ trong tất cả các tài liệu.

Chỉ mục section : tìm kiếm một số từ trong từng section.

Chỉ mục paragraph xem mỗi đoạn văn như là một tài liệu riêng biệt, thích hợp cho mục đích tìm kiếm tập trung.

Hình sau minh họa cách tìm kiếm tài liệu theo chỉ mục document và section.

Hình 2. 5 - Tìm kiếm tài liệu theo chỉ mục document và section

Trong hình trên, chapters và section titles xác định chỉ mục theo section, còn entire documents xác định chỉ mục theo document.

2.5. Tập tin cấu hình bộ sưu tập

Tập tin cấu hình của mỗi bộ sưu tập có tên collect. cfg được đặt trong thư mục "greenstone\collect\<tên bộ sưu tập>\etc" dùng để quản lý giao diện, cách thức xử lý tài liệu, cách hiển thị nội dung tài liệu…

Trong quá trình xây dựng bộ sưu tập, khi ta thực thi chương trình mkcol. pl, một tập tin cấu hình đơn giản cho bộ sưu tập được tạo, chứa các giá trị mặc định cho bộ sưu tập. Thông tin trong tập tin cấu hình bộ sưu tập bao gồm:

Bảng 2.4 – Thông tin cấu hình bộ sưu tập.

Tên	Ý nghĩa
creator	Email người tạo bộ sưu tập
maintainer	Email người quản lý bộ sưu tập
public	Xác định bộ sưu tập có cho phép công khai truy cập không
beta	Xác định bộ sưu tập có phải là phiên bản beta hay

	không
indexes	Danh sách các chỉ mục tìm kiếm
defaultindex	Chỉ mục mặc định
subcollection	Định nghĩa bộ sưu tập con dựa trên thông tin metadata
indexsubcollection	Xác định những bộ sưu tập con sẽ chỉ mục
defaultsubcollection	Chỉ mục của bộ sưu tập con mặc định
languages	Danh sách các ngôn ngữ để xây dựng chỉ mục
defaultlanguage	Xác định ngôn ngữ mặc định của bộ sưu tập
collectionmeta	Xác định metadata ở cấp bộ sưu tập
plugin	Xác định các Plugin được dùng trong quá trình xây dựng bộ sưu tập
format	Chuỗi định dạng giao diện Web
classify	Xác định các classifier dùng trong quá trình xây dựng bộ sưu tập

Ví dụ nội dung một tập tin cấu hình :

```
creator trangvuthithu@ymail.com
maintainer trangvuthithu@ymail.com
public True
beta True
indexes document:text
defaultindex document:text
plugin ZIPPlug
plugin GAPlug
plugin TextPlug
plugin HTMLPlug
plugin EMAILPlug
plugin ArcPlug
plugin RecPlug
classify AZList metadata Title
collectionmeta collectionname "Sample Collection"
```

collectionmeta iconcollection ""

collectionmeta collectionextra "Đây là phần mô tả của collection"

collectionmeta. document:text "documents"

Dòng 1 và 2 xác định email người tạo và quản lý bộ sưu tập là trangvuthithu@ymail. com. Dòng 3 xác định bộ sưu tập này được công khai sử dụng. Dòng 4 xác định đây là bộ sưu tập phiên bản beta. Dòng 5 xác định chỉ mục tìm kiếm tài liệu trong bộ sưu tập là document:text, tìm kiếm trên toàn văn bản của các tài liệu. Dòng 6 xác định chỉ mục tìm kiếm tài liệu mặc định là document:text. Từ dòng 7 đến dòng 13 xác định các plugin được dùng. Dòng 14 xác định classifier được dùng. Từ dòng 15 đến dòng 18 xác định thông tin chung của bộ sưu tập bao gồm : tên bộ sưu tập (collectionname), đường dẫn đến biểu tượng đại diện cho bộ sưu tập (iconcollection), mô tả bộ sưu tập (collectionextra), tên đại diện cho chỉ mục tìm kiếm tài liệu (tên này sẽ xuất hiện trong chức năng Search của Greenstone để người dùng có thể chọn cách tìm kiếm tài liệu).

Greenstone còn hỗ trợ chức năng tìm kiếm tài liệu trên nhiều bộ sưu tập, định nghĩa như sau:

supercollection <tên b ộ sưu tập 1> <tên bộ sưu tập 2> …<bộ sưu tập n>

Khi này, trong quá trình tìm kiếm tài liệu, Greenstone sẽ tìm trong cả n +1 bộ sưu tập: bộ sưu tập hiện tại, bộ sưu tập 1, bộ sưu tập 2, …, bộ sưu tập n

CHƯƠNG 3: Hiệu chỉnh giao diện Web GreenStone

3.1. Giới thiệu

Để chỉnh sửa giao diện ta thao tác trên các tập tin sau:

- Tập tin cấu hình của một bộ sưu tập collect. cfg
- Những tập tin macro :

Những tập tin macro có phần mở rộng là. dm, lưu trong thư mục "greenstone\macros". Macro là ngôn ngữ dành riêng cho Greenstone để xử lý giao diện web. Một macro có định dạng sau: _tên macro_. Ví dụ : _imagecollect_ là một macro.

Các trang web của Greenstone không được thiết kế sẵn, các trang này được phát sinh từ các tập tin macro. Ví dụ tập tin home. dm sẽ phát sinh ra trang chủ của Greenstone, tập tin help. dm sẽ phát sinh ra trang trợ giúp của Greenstone…

Mỗi tập tin macro chứa một hay nhiều package. Mỗi package chứa một loạt các macro. Một macro có thể được viết cho một bộ sưu tập với từ khóa [c=tên bộ sưu tập], nghĩa là ta sẽ override macro mặc định của Greenstone.

Cần tạo các macro trong package đúng. Ví dụ: muốn tạo macro "imagecollect" cho bộ sưu tập "fenian" và muốn macro này làm việc cho tất cả màn hình hiển thị, đặt macro này trong package Global :

```
package Global
_imagecollect_[c=fenian]{_gsimage_(_httppagecollect_, _httpiconccollectof_, _httpiconccollecton_, collect, _textimagecollect_)}
```

Sau đây là danh sách các package có sẵn và vai trò của chúng (tên các package có phân biệt chữ hoa, chữ thường)

Bảng 3.1 - Danh sách các package

Tên package	Tên tập tin	Vai trò
package Gobal	base.dm	Chứa những macro hoạt động trong

	english.dm	tất cả màn hình
package Style	style.dm	Quản lý kiểu hiển thị cho tất cả các trang như màu sắc, phông chữ
package document	english.dm document.dm	Chứa những macro hiển thị tài liệu
package query	query.dm english.dm	Chứa những macro cho việc hiển thị trang Search
package preferences	pref.dm english.dm	Chứa những macro cho việc hiển thị trang Preferences
package help	help.dm english.dm	Chứa những macro cho việc hiển thị trang Help
package about	about.dm english.dm	Chứa những macro cho việc hiển thị trang About
package browse	browse.dm english.dm	Chứa những macro cho giao diện duyệt tài liệu
package home	home.dm	Chứa những macro cho giao diện trang chủ

- Những tập tin ảnh :

Tất cả những tập tin ảnh được chứa trong thư mục "greenstone\images".

- Main. cfg :

Chứa các khai báo các tập tin macro được dùng trong Greenstone. Nếu tạo tập tin macro mới, để tập tin macro này có hiệu lực, cần thêm tên tập tin macro đó vào nội dung tập tin main. cfg. Tập tin main. cfg được lưu trong thư mục "greenstone\etc".

Giao diện Greenstone gồm 5 phần như hình vẽ bên dưới :

Hình 3.1 – Các thành phần trong giao diện Web của GreenStone

Sau đây ta sẽ đi chi tiết từng phần của giao diện theo thứ tự hình trên.

3.2. Ảnh tiêu đề bộ sưu tập

Được chỉ ra trong tập tin cấu hình collect. cfg :

iconcollection "_httpprefix_/đường dẫn đến tập tin ảnh/tên tập tin ảnh"

Ý nghĩa:

- iconcollection: từ khóa để tạo ảnh cho bộ sưu tập
- _httpprefix_: thư mục Greenstone

Ví dụ 1 đoạn trong tập tin collect. cfg :

```
collectionmeta iconcollection "_httpprefix_/collect/demo/images/en/demo.gif"
```

3.3. Các nút duyệt trang

3.3.1. Cách hiển thị

Sự trình bày của những nút duyệt trang được quản lý bởi macro _javalinks_ được định nghĩa trong package Global của tập tin base. dm như sau:

javalinks {_imagehome__imagehelp__imagepref_}

Macro này hiển thị 3 nút HOME, HELP và PREFERENCE như ở hình 4. 1. Macro _imagehome_ quản lý nút HOME, _imagehelp_ quản lý nút HELP và _imagepref_ quản lý nút PREFERENCE

Vị trí của những nút này được chỉ ra bởi macro trình bày trang _pagebanner_ trong tập tin macro style. dm. Ví dụ sau hiển thị những nút duyệt bên phải của phần header được minh họa trong hình 3. 2

Hình 3.2 – Các nút duyệt tài liệu

Có thể thay đổi vị trí của những nút này, thêm mới hoặc xóa nút duyệt trang.

Thêm một nút mới

1. Tạo một macro ảnh mới trong package Global trong tập tin base. dm,

macro này phải cùng định dạng với những macro ảnh khác. Cú pháp định nghĩa một

macro ảnh như sau:

_<tên macro ảnh>_{ _gsimage_ (macro url, macro ảnh "of", macro ảnh "on",

chuỗi, macro chuỗi mô tả ảnh)}

Với _gsimage_ là một macro chuẩn của Greenstone. Các macro url, macro

ảnh "of", macro ảnh "on", chuỗi, macro chuỗi mô tả ảnh do người dùng định nghĩa

Ví dụ tạo một macro có tên là _imagecollect_ như sau:

imagecollect {_gsimage_(_httppagecollect_, _httpiconccollectof_,

httpiconccollecton, collect, _textimagecollect_)}

2. Tạo một macro _httppage_ (marco url) trong package Global của tập tin

base. dm. Đây là URL để mở khi nút được nhấn. Ví dụ :

httppagecollect {http://www. aladin. wrlc. org/dl/}

3. Tạo những macro _httpicon_ (định nghĩa các macro ảnh "of", ảnh "on"), những macro này chỉ ra vị trí của những ảnh, những macro này trong package Global của tập tin english. dm.

4. Tạo những tập tin ảnh có tên được định nghĩa ở bước thứ 3, đặt những tập tin ảnh này trong thư mục images của Greenstone.

5. Tạo macro chuỗi mô tả ảnh (macro text) cho nút. Macro này đặ t trong package Global của tập tin english. dm. Đây là đoạn "alt" trong thẻ <img> của nút. Ví dụ :

textimagecollect {WRLC Special Collections}

6. Thêm macro ảnh này đến macro _javalinks_ trong package Global của tập tin base. dm, ví dụ

javalinks {_imagehelp_
imagehome
imagecollect}

Macro này hiển thị 3 nút theo chiều dọc như minh họa ở hình 3. 2

Xóa nút duyệt trang

Giả sử ta cần xóa nút HOME, trong tập tin base. dm, ở package Global, ta thay đổi macro _javalinks_ như sau:

javalinks {_imagehelp_ _imagepref_}

javalinks [v=1] {

imagehelp

imagepref
 }

Ở macro _javalinks_ trên, macro _imagehome_ đã được xóa.

Thay đổi nút duyệt trang

Có thể thay đổi nội dung trên nút duyệt trang bằng cách chỉnh sửa các ảnh của nút, hoặc thay đổi liên kết của nút duyệt trang bằng cách chỉnh sửa macro _httppageX_(ví dụ _httppagecollect_), thay đổi chuỗi mô tả ảnh bằng cách chỉnh sửa macro _textimageX_ (ví dụ _textimagecollect_) hoặc thay đổi vị trí của nút này so với nút khác bằng cách chỉnh sửa macro _javalinks_

3.3.2. Vị trí đặt các ảnh

Những macro xác định vị trí ảnh của những nút duyệt trang được chỉ ra trong package Global trong tập tin english. dm. Theo quy ước các macro này bắt

đầu với _httpicon_, theo sau là kí tự "c" và tên nút. Mỗi nút được tạo bởi một cặp ảnh "on" và "off". Để thay đổi những nút này, ta cần tạo 2 ảnh.

```
## "HOME" ## top_nav_button ## chome ##
_httpiconchomeof_ {_httpimg_/chomeof. gif}
_httpiconchomeon_ {_httpimg_/chomeon. gif
##"HELP"##top_nav_button##chelp##
_httpiconchelpof_{_httpimg_/chelpof. gif}
_httpiconchelpon_ {_httpimg_/chelpon. gif}
```

Chú ý trong các tập tin macro, theo sau dấu "#" là dòng ghi chú. Macro _httpimg_ là một macro chuẩn để chỉ thư mục ảnh của Greenstone (greenstone\images).

Để đặt những hình ảnh nằm trong thư mục khác với thư mục mặc định "greenstone\images", ta có thể đặt đường dẫn như sau :

```
#Collections
_httpiconccollectof_{_httpprefix_/collect/auhist/images/ccollectof. gif}
_httpiconccollecton_ {_httpprefix_/collect/auhist/images/ccollecton. gif}
```

Khi này 2 ảnh ccollectof. gif và ccollecton. gif nằm trong thư mục images của bộ sưu tập auhist. Macro _httpprefix_ là macro chuẩn chỉ thư mục Greenstone.

3.4. Ảnh tiêu đề trang web

Ảnh tiêu đề trang được quản lý bởi macro _imagethispage_. Ta đặt macro này trong package ta muốn hiển thị ảnh. Ví dụ, nếu ta muốn hiển thị một ảnh tiêu đề cho trang Help :

```
package help
_imagethispage_ [c=auhist] {_iconthispage_}
_iconthispage_ [c=auhist] {<img src="_httpiconhhelp_" alt="_textHelp_"
width=_widthhhelp_ height=_heighthhelp_>}
```

Khi ta định nghĩa _imagethispage_, ta phải định nghĩa _iconthispage_. Không có ảnh tiêu đề cho những trang hiển thị tài liệu.

Vị trí của những ảnh tiêu đề được định nghĩa trong package document của tập tin english. dm. Tất cả macro này bắt đầu với _httpicon_, theo sau là "h" và tên nút. Ta quy ước đặt tên cho các ảnh này là h_imagename. Ví dụ: h_title. gif, h_subj. gif…

Độ rộng và chiều cao là số pixel. Ta có thể thay đổi chúng nếu cần thay đổi độ rộng và chiều cao của ảnh.

Để thay đổi vị trí lưu trữ ảnh, ta cần thay đối đường dẫn đến những tập tin ảnh trong cặp ngoặc { }.

3.5. Các nút duyệt tài liệu

3.5.1. Giới thiệu

Các nút duyệt tài liệu, ngoại trừ nút Search, được quản lý bởi các classifier. Khi định nghĩa một classifier trong tập tin collect. cfg của bộ sưu tập, những nút liên quan sẽ xuất hiện trên thanh duyệt.

Những nút này được định nghĩa trong package Global, thường trong tập tin base. dm.

3.5.2. Vị trí đặt các ảnh

Vị trí của những tập tin ảnh được chỉ ra trong những macro của package Global, thường trong tập tin english. dm. Tất cả những macro này bắt đầu với _httpicon_ và theo sau là ký tự "t" và tên nút. Ta gọi nó là "T button".

Một nút trên thanh duyệt được tạo bởi 3 ảnh, ảnh "off", ảnh "on", và ảnh "gr". Nếu ta muốn thay đổi các hiển thị của những nút này, ta cần tạo ra 3 ảnh:

```
## "titles AZ" ## nav_bar_button ## ttitl ##
_httpiconttitlgr_ {_httpimg_/ttitlgr. gif}
_httpiconttitlof_ {_httpimg_/ttitlof. gif}
_httpiconttitlon_ {_httpimg_/ttitlon. gif}
_widthttitlx_ {87}
## "authors AZ" ## nav_bar_button ## tauth ##
_httpicontauthgr_ {_httpimg_/tauthgr. gif}
_httpicontauthof_ {_httpimg_/tauthof. gif}
_httpicontauthon_ {_httpimg_/tauthon. gif}
```

widthtauthx {110}

Ta cũng cần thay đổi cách hiển thị của những ảnh khoảng trống (space image) đặt giữa những nút này. Tập tin ảnh khoảng trống này được đặt tên là "tspace. gif" và nó được chỉ ra trong package Global trong tập tin base.dm:

httpicontspace {_httpimg_/tspace. gif}

heighttspace {17}

Độ rộng của những ảnh này là số pixel. Nếu muốn có một dòng chữ dài, ví dụ như "Place Names", và nút này rộng hơn những nút khác, ta thiết đặt độ rộng trong cặp ngoặc{}. Khi xây dựng bộ sưu tập, Greenstone sẽ tự động tính toán khoảng trống giữa những nút và tập tin tspace. gif sẽ phủ toàn bộ khoảng trống này.

3.5.3. Thêm nút mới

Có 2 cách để thêm một nút mới

- Dùng một nút đã có và thay đổi đoạn chữ trên ảnh
- Tạo một macro mới cho nút mới và tạo những tập tin ảnh cho nút mới này.

3.5.3.1. Tạo một nút mới dựa vào nút đã có

Ví dụ, ta cần một nút "Places" và dùng một nút đã có "From" để tạo nút mới này. Cách làm như sau:

- Định nghĩa một classifier cho nút mới trong tập tin collect. cfg.

classify AZList -metadata dc. Title -buttonname From

- Trong một chương trình xử lý ảnh, thay đoạn text "From" thành "Places" trên tất cả 3 ảnh, có thể thay đổi màu sắc, và đặt những ảnh này trong thư mục images. tfromgr.gif, tfromof.gif, tfromon.gif.

3.5.3.2. Tạo macro mới cho một nút

Ví dụ muốn tạo mộ t nút mới "Museums". Cách làm như sau:

Tạo một classifier trong tập tin collect.cfg:

```
classify AZList -metadata Museums -buttonname Museums
```

Tạo một macro ảnh trong package Global dùng cùng định dạng với những macro ảnh khác

```
_imagemuseum_ {_gsimage_(_httpbrowseMuseum_,_httpicontMuseumof_,
_httpicontMuseumon_,Museums,_textimageMuseum_)}
```

Xác định vị trí của những ảnh và độ rộng của những ảnh này

```
#Museum
_httpicontmuseumgr_ {_httpprefix_/collect/auhist/images/tmuseumgr.gif}
_httpicontmuseumof_ {_httpprefix_/collect/auhist/images/tmuseumof.gif}
_httpicontmuseumon_ {_httpprefix_/collect/auhist/images/tmuseumon.gif}
_widthtmuseumsx_ {110}
```

Tạo 3 ảnh và lưu chúng trong thư mục images

3.5.4. Xóa nút duyệt tài liệu

Xóa nút duyệt tài liệu bằng cách xóa macro ảnh _imageX_ (ví dụ _imagemuseum_) trong package Global.

3.5.5. Thay đổi nút duyệt tài liệu

Thay đổi nút duyệt tài liệu bằng cách thay đổi macro _httpbrowseX_ (ví dụ _httpbrowseMuseum_) hay _httpicontX_ (ví dụ _httpicontmuseumgr_) hay _textimageX_ (ví dụ _textimageMuseum_)…

3.6. Hiển thị văn bản

Có 2 cách hiển thị văn bản. Chúng được quản lý trong tập tin collect.cfg.

3.6.1. Hiển thị loại CL list

Hiển thị danh sách các tài liệu được được quản lý bởi những chuỗi format trong tập tin collect.cfg. Ví dụ, chuỗi format sau sẽ hiển thị danh sách các tiêu đề của tài liệu là classifier đầu tiên, với một thumbnail và phần mô tả chung, được minh họa ở hình 3.3

Hình 3.3 – Hiển thị danh sách các tài liệu

3.6.2. Hiển thị nội dung

Khi ta click lên trên tiêu đề trong danh sách các tiêu đề, ta thấy hoặc là trang nội dung hoặc là một record metadata mô tả đối tượng số hóa. Sự hiển thị trang này được quản lý bởi Format DocumentHeading và Format DocumentText trong tập tin collect.cfg.

3.7. Override các Macro

Thay vì sửa trực tiếp trong các tập tin macro mặc định của Greenstone, có thể tạo một tập tin macro mới, sau đó override những macro muốn sửa đổi. Khi override các macro, phải chỉ ra macro này nằm trong package nào. Ngoài override các macro có sẵn trong Greenstone, có thể thêm các macro mới vào tập tin này.

Ví dụ cách override một macro: giả sử ta muốn thay đổi giao diện trang chủ của Greenstone như hình vẽ dưới đây ta làm như sau :

- Tạo một tập tin macro mới đặt tên là zitlibweb.dm. Tập tin lưu trong thư mục macro của Greenstone
- Soạn thảo nội dung của tập tin macro này. Ta thay đổi những macro cần thiết.
- Thêm vào nội dung tập tin main.cfg tên của tập tin macro vừa tạo.

Lúc này giao diện trang chủ của Greenstone sẽ thay đổi lập tức.

Chú ý:

- Nếu chỉ muốn áp dụng giao diện mới cho bộ sưu tập nào đó thôi thì ta thêm dòng sau trước mỗi macro [c=tên bộ sưu tập]

3.8. Thêm một trang mới

Để tạo một trang mới ta cần tạo một tập tin macro mới. Ví dụ thêm một trang web tên là mypage :

- Tạo tập tin macro mới tên mypage.dm trong thư mục "greenstone\macros"

- Để liên kết trang này với trang home, trong macro _content_ của tập tin macro home.dm, ta thêm một link như sau: My Page

- Thêm tên tập tin macro mới này vào danh sách các macro trong tập tin main.cfg

mypage.dm sẽ chứa nội dung của trang mới.

3.9. Hiển thị các bộ sưu tập

Giả sử với giao diện trang chủ ở hình 3.4, khi click vào hyperlink của bộ sưu tập luận văn, sẽ đến trang zthesis. Trang zthesis hiển thị tất cả các bộ sưu tập luận văn, mỗi bộ sưu bao gồm các luận văn của một khóa học nào đó, ví dụ bộ sưu tập luận văn khóa 7, bộ sưu tập luận văn khóa 8….

Ta tạo trang web mới zthesis bằng cách tạo một tập tin macro zthesis.dm.

Nội dung tập tin zthesis.dm như sau:

```
package zthesis
#Tạo ảnh banner
_imagecollection_ {<img src="_httpimg_/banner1.jpg">}
#Tạo liên kết
_mycollectionurl_ {_gwcgi_?a=p&p=about&c=_1_}
_mycollectionlink_ {<a href="_mycollectionurl_(_1_)">_2_</a>}
_pagetitle_ {Bộ sưu tập luận văn}
#Xóa 3 nút home,help,pref
 _javalinks_ {}
_javalinks_ [v=1] {}
#Phần nội dung
_content_ {
<center><H1><font color = blue>BỘ SƯU TẬP LUẬN VĂN</font></H1></center>
<table align= center> <tr> <td><a href="http://localhost/itlibweb/index.aspx">
<img src="_httpimg_/logo_home.jpg" border="0"></a></td>
 <td><a href="_httppagex_(home)">
<img src="_httpimg_/logo_search.jpg" border="0"></a></td> </tr> </table>
<table align ="center">
<tr> <td width =100>_mycollectionlink_(LV01,Luận văn Khóa 7)</td>
 <td width =100>_mycollectionlink_(LV02,Luận văn Khóa 8)</td> </tr>
</table>
}
```

Ta định nghĩa 2 macro mới _mycollectionurl_, _mycollectionlink_ để tạo các liên kết đến các bộ sưu tập. Sau đó trong _content_ ta tạo 2 liên kết đến 2 bộ sưu tập luận văn khóa 7, luận văn khóa 8 bằng cách dùng macro _mycollectionlink_

3.10. Macro chuẩn

Bảng 3.2 – Một số macro chuẩn

Tên	Ý nghĩa
httpprefix	Thư mục GreenStone
httpimg	Thư mục ảnh của GreenStone
httppagex	Định nghĩa một trang web nào đó của GreenStone
httpiconchalk	Định nghĩa ảnh nền cho GreenStone
content	Định nghĩa nội dung hiển thị một trang
javalinks	Trình bày nút duyệt trên cùng
gsimage	Định nghĩa các macro ảnh
imagethispage	Định nghĩa ảnh header

CHƯƠNG 4: Hệ thống Web GreenStone

4.1. Tổng quan về cơ chế xử lý

Hệ thống web của Greenstone được viết bằng ngôn ngữ C++ dùng cơ chế CGI.

Hình 4.1 - Cơ chế xử lý

Hình trên minh họa cách một số người dùng truy cập đến các bộ sưu tập của thư viện số. Người dùng được đại diện bằng các máy tính. Sau khi các bộ sưu tập được xây dựng, chúng được truy cập "online" từ phía người dùng. Để truy cập đến các bộ sưu tập, người dùng phải đi qua 2 đơn thể trừu tượng, receptionist và collection server. Receptionist có thể xem như phần giao diện để giao tiếp với các thư viện số và người dùng. Chúng cho phép người dùng nhập thông tin sau đó phân tích và gửi các yêu cầu này đến các collection server thích hợp. Collection server nhận được yêu cầu, phân tích, lấy những thông tin được yêu cầu từ các bộ sưu tập và phản hồi lại cho receptionist, sau đó receptionist hiển thị kết quả cho người dùng.

Receptionist kết nối với Collection server thông qua một nghi thức giao tiếp. Sự thực thi nghi thức này tùy thuộc vào cấu hình từng máy tính. Trong trường hợp đơn giản nhất, chỉ có một receptionist và một collection server chạy trên cùng một máy tính. Khi này receptionist và collection server hợp thành một thể thống nhất gọi là library và giữa chúng dùng một nghi thức null protocol

Hình 4.2 - Collection server và receptionist liên lạc qua nghi thức null protocol

Thông thường server phải chạy liên tục để phản hồi những yêu cầu từ phía client. Đối với collection server dùng nghi thức null protocol thì khác. Chỉ khi nào có một trang web của Greenstone được yêu cầu, chương trình library sẽ được khởi động bởi cơ chế CGI, phản hồi những yêu cầu, sau đó kết thúc ngay. Điều này có thể làm chậm tiến trình xử lý. Để khắc phục ta dùng cơ chế Fast-CGI, chương trình library sẽ được lưu lại trong bộ nhớ ở lần thực thi đầu tiên. Để dùng Fast-CGI ta phải biên dịch lại Greenstone.

4.2. Chi tiết về cơ chế xử lý

Hình dưới đây minh họa cho trang web "about this collection" của bộ sưu tập Project Gutenberg. Trang web được phát sinh từ chương trình library

Hình 4.3 - Bộ sưu tập Project Gutenberg

Trên thanh địa chỉ lúc này có chuỗi như sau:

http://www.nzdl.org/cgi-bin/library?c=gberg&a=p&p=about

Chuỗi này có ý nghĩa: người dùng muốn truy cập đến collection gberg(c=gberg), hành động(action) là muốn phát sinh ra một trang web (a = p) và trang được phát sinh là trang about (p=about).

Receptionist đầu tiên sẽ khởi tạo các thành tố của nó, sau đó phân tích các tham số CGI để quyết định gọi các hành động nào. Khi một hành động được thực hiện, receptionist thông qua nghi thức chung truy cập đến nội dung của bộ sưu tập. Những phản hồi từ phía server được receptionist dùng các thành tố như chuỗi định dạng (format), macro để phát sinh trang web kết quả hiển thị cho người dùng.

Collection server cũng trải qua quá trình kh ởi tạo các thành tố của nó, sau đó dùng các thành tố như Filter, Source, Search để phản hồi những yêu cầu gửi đến. Các thành tố như Filter, Source, Search truy cập đến hệ thống chỉ mục và

cơ sở dữ liệu quản lý thông tin bộ sưu tập để lấy kết quả phản hồi cho người dùng.

Hình sau minh họa chi tiết hệ thống Greenstone dùng nghi thức null protocol:

Hình 4.4 - Cơ chế xử lý dùng null protocol

Thống kê cho thấy receptionist dùng khoảng 15 000 dòng mã nguồn, còn collection server chỉ dùng khoảng 5000 dòng mã nguồn. Nhưng mã nguồn của collection server phức tạp hơn. Trong collection server có dùng 2 đơn thể bên ngoài, đó là MG và GDMB. MG dùng để nén văn bản, tạo chỉ mục, truy vấn dữ liệu phục vụ cho quá trình tìm kiếm tài liệu. GDBM là một hệ quản trị cơ sở dữ liệu dùng để quản lý thông tin của bộ sưu tập. Đây là 2 đơn thể cốt lõi củ a collection server.

4.3. Mã nguồn

Mã nguồn của hệ thống web Greenstone được viết bằng thư viện STL (Standard Template Library) của C++ và được đặt trong thư mục "greenstone\src".

Hình 4.5 - Cấu trúc thư mục chứa mã nguồn Greenstone

Trong thư mục src có 2 thư mục con chính là: colservr chứa mã nguồn của collection server và recpt chứa mã nguồn của receptionist.

Ngoài ra trong thư mục greenstone\lib còn chứa mã nguồn các lớp, các hàm cơ bản dùng cho cả collection server và receptionist.

4.3.1. Các lớp và hàm cơ bản

- text_t.h : định nghĩa lớp đối tượng hỗ trợ Unicode cho Greenstone.
- cfgread.h : định nghĩa các hàm đọc và ghi tập tin cấu hình.
- display.h : định nghĩa các lớp đối tượng dùng cho receptionist trong việc cài đặt, lưu trữ và mở rộng các macro.
- fileutil.h : định nghĩa các hàm thao tác với tập tin.
- gsdlconf.h : định nghĩa các hàm mức hệ thống.
- gsdltimes.h : định nghĩa các hàm thao tác trên kiểu dữ liệu thời gian.
- gsdltools.h : định nghĩa các hàm hỗ trợ hệ thống Greenstone : kiểm tra trình thực thi Perl có tồn tại không, thực thi một câu lệnh hệ thống, xác định tài liệu định dạng theo little Endian hay big Endian…
- gsdlunicode.h : định nghĩa các lớp đối tượng hỗ trợ xử lý chuỗi Unicode text_t.

4.3.2. Collection server

Collection server gồm 3 lớp cơ sở: searchclass, sourceclass,filterclass.

Lớp đối tượng searchclass là lớp cơ sở, định nghĩa các phương thức ảo để thực hiện các câu truy vấn tìm kiếm tài liệu. Greenstone dùng MG để tìm kiếm tài liệu. Do đó có một lớp mgsearchclass kế thừa từ lớp searchclass. Lớp mgsearchclass định nghĩa lại một số phương thức ảo của lớp searchclass, các phương thức này gọi đế n thư viện MG. Mã nguồn mở của MG có thể tìm trong thư mục "greenstone\packages\mg".

Lớp đối tượng sourceclass có nhiệm vụ truy cập metadata và nội dung của tài liệu. 2 hàm ảo get_metadata() và get_document() thực hiện nhiệm vụ trên. Greenstone dùng 2 thư viện MG và GDBM để cài đặt cho lớp sourceclass thông qua lớp mggdmbsourceclass.

Lớp đối tượng filterclass là lớp cơ sở để lọc các tài liệu có được từ một câu truy vấn hay từ hành động duyệt xem các tài liệu.

Định nghĩa lớp, hàm của collection server :

- browsefilter.h: định nghĩa lớp browsefilterclass kế thừa từ filterclass, dùng để truy cập đến thư viện GDBM.
- collectserver.h: bao bọc các đối tượng Filter và Source
- colservrconfig.h: cung cấp những hàm đọc file cấu hình etc\collect.cfg và index\build.cfg
- filter.h : định nghĩa lớp cơ sở filterclass
- maptools.h : định nghĩa lớp tên là stringmap được dùng trong các lớp mggdbmsourceclass và queryfilterclass.
- mggdbmsource.h: định nghĩa lớp mggdbmsourceclass kế thừa từ lớp sourceclass, truy cập đến MG và GDBM.
- mgppqueryfilter.h: định nghĩa lớp mgppqueryfilterclass kế thừa từ lớp queryfilterclass, lớp này cài đặt cho lớp queryfilterclass dùng MG++.
- mgppsearch.h: định nghĩa lớp mgppsearchclass kế thừa lớp searchclass, lớp này cài đặt cho lớp searchclass dùng MG++

- mgq.h: cung cấp các hàm giao tiếp với thư viện MG.
- mgqueryfilter.h: định nghĩa lớp mgqueryfilterclass kế thừa từ lớp
- queryfilterclass, dùng để cài đặt cho lớp queryfilterclass dùng MG
- mgsearch.h: định nghĩa lớp mgsearchclass kế thừa từ lớp searchclass, cài đặt cho lớp searchclass dùng MG.
- querycache.h: định nghĩa lớp querycache, dùng cho lớp searchclass. Những lớp kế thừa từ nó sẽ lưu tạm kết quả truy vấn để tối ưu việc truy vấn.
- queryfilter.h: định nghĩa lớp queryfilterclass kế thừa từ lớp filterclass, đây là lớp cơ sở cho các đối tượng lọc các câu truy vấn.
- queryinfo.h: hỗ trợ việc tìm kiếm tài liệu, định nghĩa các cấu trúc dữ liệu cho tham số các câu truy vấn, kết quả trả về.
- search.h: định nghĩa lớp đối tượng cơ sở searchclass.
- source.h: định nghĩa lớp đối tượng cơ sở sourceclass.

4.3.3. Receptionist

Greenstone định nghĩa một khái niệm trừu tượng đó là Action. Một Action là một hành động của người dùng tương tác với hệ thống web Greenstone hay một hành động của hệ thống Greenstone.

Bảng 5.1 – Danh sách các Action

Tên Action	Ý nghĩa
Action	Lớp cơ sở
Authenaction	Hỗ trợ chứng thực người dùng
Collectoraction	Phát sinh trang Web cho Collector
Documentaction	Nhận kết quả tìm kiếm tài liệu hoặc định dạng thông tin tài liệu

Pageaction	Phát sinh trang web dựa vào các macro
Pingaction	Kiểm tra một bộ sưu tập có online hay không
Queryaction	Thực hiện việc tìm kiếm tài liệu
Statusaction	Phát sinh trang web của Admin
Useraction	Hỗ trợ quản lý người dùng

Định nghĩa lớp, hàm trong receptionist :

- action.h : định nghĩa lớp cơ sở
- authenaction.h : định nghĩa lớp authenaction kế thừa từ lớp action hỗ trợ chứng thực người dùng.
- browserclass.h : định nghĩa lớp cơ sở cho hành động duyệt tài liệu
- browsetools.h : định nghĩa các hàm hỗ trợ cho lớp browserclass
- cgiargs.h : định nghĩa cấu trúc dữ liệu cgiarginfo và những cấu trúc dữ liệu khác hỗ trợ cho các tham số CGI
- cgiutils.h : định nghĩa các hàm hỗ trợ cho các cấu trúc dữ liệu định nghĩa trong cgiargs.h.
- collectoraction.h : định nghĩa lớp kế thừa từ lớp action cho phép người dùng xây dựng bộ sưu tập bằng Collector. Trang Collector được phát sinh từ macro collect.dm.
- datelistbrowserclass.h : định nghĩa lớp kế thừa từ browserclass, hỗ trợ các bộ sưu tập có tài liệu được duyệt theo ngày tháng.
- documentaction.h : định nghĩa lớp kế thừa từ lớp action hỗ trợ lấy tài liệu từ một câu truy vấn.
- formattools.h : định nghĩa các hàm hỗ trợ cho chuỗi định dạng (format).
- hlistbrowserclass.h : định nghĩa lớp kế thừa từ browserclass, hỗ trợ các bộ sưu tập có tài liệu được sắp xếp theo danh sách ngang.

- htmlbrowserclass.h : định nghĩa lớp kế thừa từ browserclass, hỗ trợ duyệt các tài liệu html.
- htmlgen.h : định nghĩa các hàm hỗ trợ cho việc highlight các chuỗi liên quan trong nội dung tài liệu tìm được..
- infodbclass.h : định nghĩa 2 lớp : gdbmclass và infodbclass, hỗ trợ truy cập thư viện GDBM.
- pageaction.h : định nghĩa lớp kế thừa từ lớp action hỗ trợ phát sinh trang web.
- querytools.h : định nghĩa những hàm hỗ trợ truy vấn dữ liệu.
- recptconfig.h : định nghĩa các hàm hỗ trợ đọc file cấu hình main.cfg và gsdlsite.cfg.
- statusaction.h : định nghĩa lớp kế thừa từ lớp action, hỗ trợ phát sinh trang quản lý cho admin.
- userdb.h : định nghĩa các cấu trúc dữ liệu và hàm quản lý cơ sở dữ liệu người dùng.
- usersaction.h : định nghĩa lớp kế thừa từ lớp action, hỗ trợ admin quản lý người dùng.
- vlistbrowserclass.h : định nghĩa lớp kế thừa từ lớp browserclass, hỗ trợ cho các tài liệu được liệt kê theo danh sách chiều dọc.

CHƯƠNG 5: Cấu hình thử nghiệm

5.1. Môi trường thử nghiệm

Phần cứng : Một máy tính pentum IV.

- 50MB đĩa cứng (HDD) còn trống chỗ.

Phần mềm :

- Cài Java Version 1.4 trở lên.
- Cài ImageMagick.
- Cài GreenStone.

5.2. Phát biểu bài toán ứng dụng

Trong phạm vi đề tài, em sử dụng phần mềm mã nguồn mở GreenStone xây dựng bộ sưu tập Luận văn tốt nghiệp bao gồm các đề tài tốt nghiệp của sinh viên ngành Công nghệ thông tin trường Đại học Dân lập Hải Phòng với các hướng dẫn cụ thể. Bộ sưu tập này được cấu hình và định dạng để thuận tiện cho việc tra cứu. Giao diện Web được hiệu chỉnh cho phù hợp để sử dụng tại trường Đại học Dân lập Hải Phòng.

5.3. Giải quyết bài toán

5.3.1. Xây dựng một bộ sưu tập với GLI

5.3.1.1. Giao diện thủ thử GreenStone Librarian Interface (GLI)

Giao diện Greenstone Library Interface cung cấp giao diện tương tác phản ánh các bước thực hiện như sau:

- **GATHER**: Dùng để thu gom tài liệu tập trung vào bộ sưu tập.
- **ENRICH**: Trình bày 15 yếu tố của Dublin Core để biên mục tài liệu. Động tác biên mục được làm thủ công. Trong đó, 15 yếu tố của chuẩn Dublin Core :

1. Nhan đê (Title) : Nhan đề của tài liệu.
2. Tác giả (Creator) : Tác giả của tài liệu, bao gồm cả tác giả cá nhân và tập thể.

3. Chủ đề (Subject): Chủ đề tài liệu đề cập dùng để phân loại tài liệu. Có thể thể hiện bằng từ, cụm từ/(Khung chủ đề), hoặc chỉ số phân loại/ (Khung phân loại).
4. Tóm tắt (Description): Tóm tắt, mô tả nội dung tài liệu. Có thể bao gồm tóm tắt, chú thích, mục lục, đoạn văn bản để làm rõ nội dung...
5. Nhà xuất bản (Publisher): Nhà xuất bản, nơi ban hành tài liệu có thể là tên cá nhân, tên cơ quan, tổ chức, dịch vụ...
6. Tác giả phụ (Contributor): Tên những người cùng tham gia cộng tác đóng góp vào nội dung tài liệu, có thể là cá nhân, tổ chức...
7. Ngày tháng (Date): Ngày, tháng ban hành tài liệu.
8. Loại (kiểu) (Type): Mô tả bản chất của tài liệu. Dùng các thuật ngữ mô tả phạm trù kiểu: trang chủ, bài báo, báo cáo, từ điển...
9. Khổ mẫu (Format): Mô tả sự trình bày vật lý của tài liệu, có thể bao gồm; vật mang tin, kích cỡ độ dài, kiểu dữ liệu (.doc,.html,.jpg, xls, phần mềm....)
10. Định danh (Identifier): Các thông tin về định danh tài liệu, các nguồn tham chiếu đến, hoặc chuỗi ký tự để định vị tài nguyên: URL (Uniform Resource Locators) (bắt đầu bằng http://), URN (Uniform Resource Name), ISBN (International Standard Book Number), ISSN (International Standard Serial Number), SICI (Serial Item & Contribution Identifier),...
11. Nguồn (Resource): Các thông tin về xuất xứ của tài liệu, tham chiếu đến nguồn mà tài liệu hiện mô tả được trích ra/tạo ra, nguồn cũng có thể là: đường dẫn (URL), URN, ISBN, ISSN...
12. Ngôn ngữ (Language): Các thông tin về ngôn ngữ, mô tả ngôn ngữ chính của tài liệu.
13. Liên kết (Relation): Mô tả các thông tin liên quan đến tài liệu khác. có thể dùng đường dẫn (URL), URN, ISBN, ISSN...
14. Diện bao quát (Coverage): Các thông tin liên quan đến phạm vi, quy mô hoặc mức độ bao quát của tài liệu. Phạm vi đó có thể là địa điểm, không gian hoặc thời gian, tọa độ...

15.Bản quyền (Right): Các thông tin liên quan đến bản quyền của tài liệu.

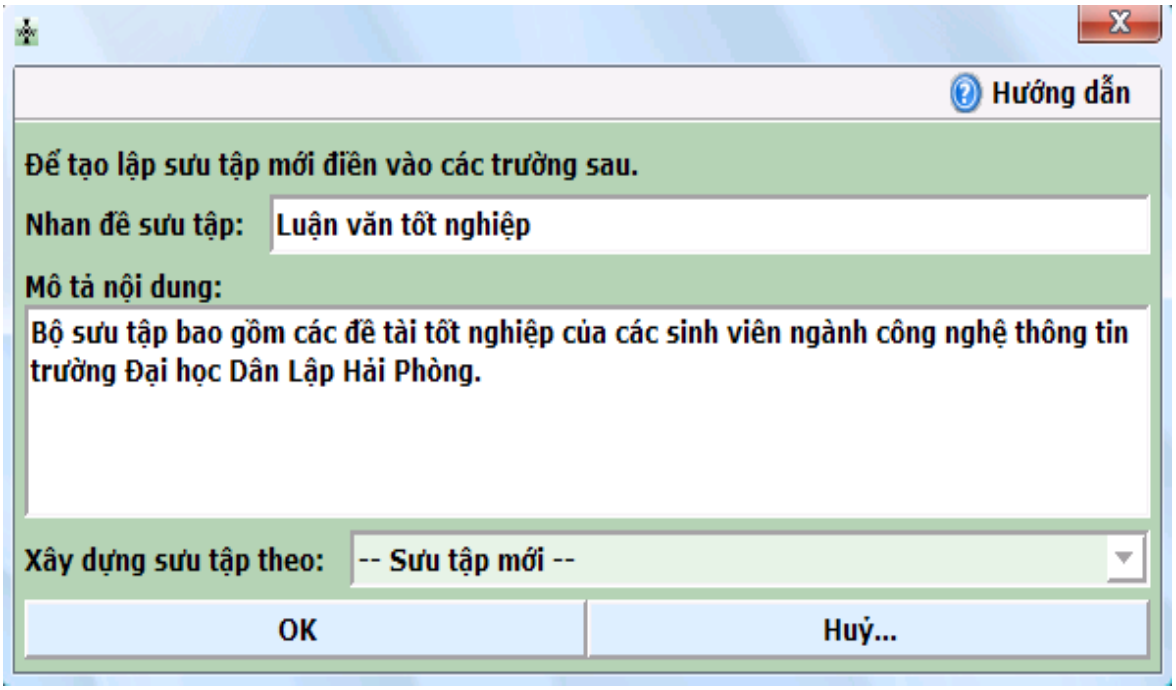
- **DESIGN**: Dùng để thiết kế giao diện bộ sưu tập kèm theo những chỉ thị tìm kiếm theo những dẫn mục cho ta chọn, chẳng hạn như: tác giả, nhan đề, năm,…
- **CREAT**: Dùng để sản xuất bộ sưu tập.
- **FORMAT** : Trình bày tài liệu.

5.3.1.2. Các bước xây dựng bộ sưu tập với GLI

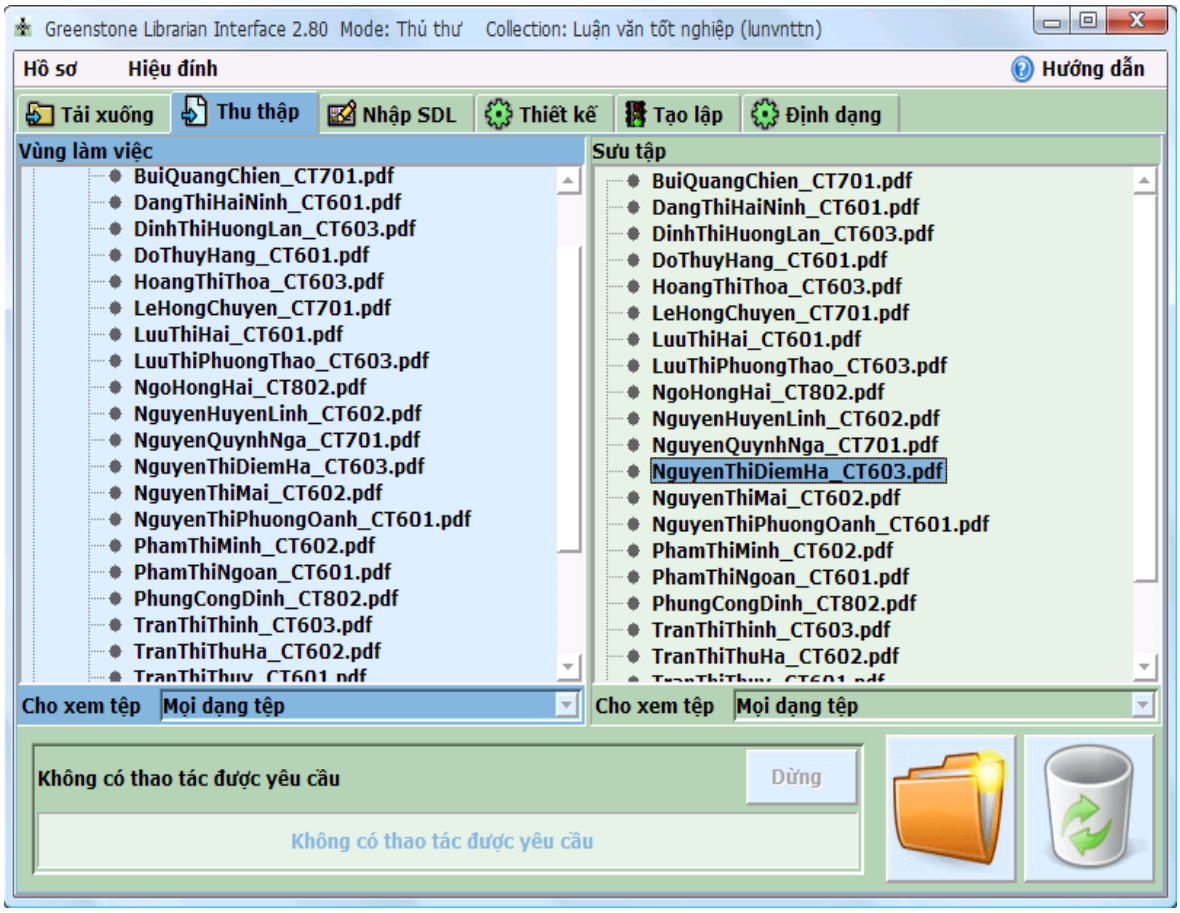
- Bước 1 : Khởi động GLI
  - Start/ Program/ Greenstone/Greenstone Librarian Interface.
  - Chọn File/ New để tạo bộ sưu tập mới: Gõ tên vào mục Collection title và mô tả tóm tắt vào phần Description of content.
- Bước 2: Chọn tài liệu cần tạo bộ sưu tập
  - Bấm Gather/ Mở đến địa chỉ chứa tài liệu trong phần workspace.
  - Chọn tập tin tài liệu kéo và thả vào cửa sổ Collection bên phải.
- Bước 3: Biên mục cho từng tài liệu đưa vào theo chuẩn Dublin Core
  - Bấm vào Enrich/ Chọn file tương ứng rồi biên mục theo 15 trường của Dubline Core.
- Bước 4: Xây dựng bộ sưu tập
  - Create/ Build collection.

5.3.2. Xây dựng bộ sưu tập Luận văn tốt nghiệp

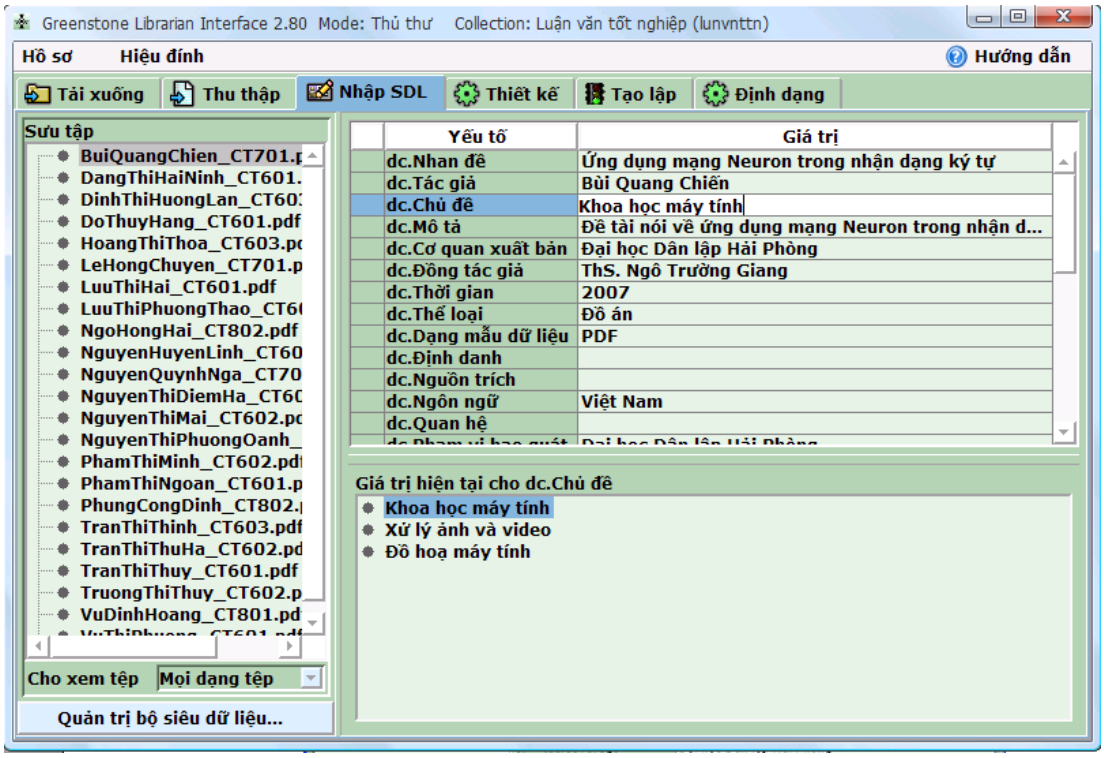
Dưới đây là một số giao diện tương ứng các bước xây dựng bộ sưu tập Luận án tốt nghiệp.



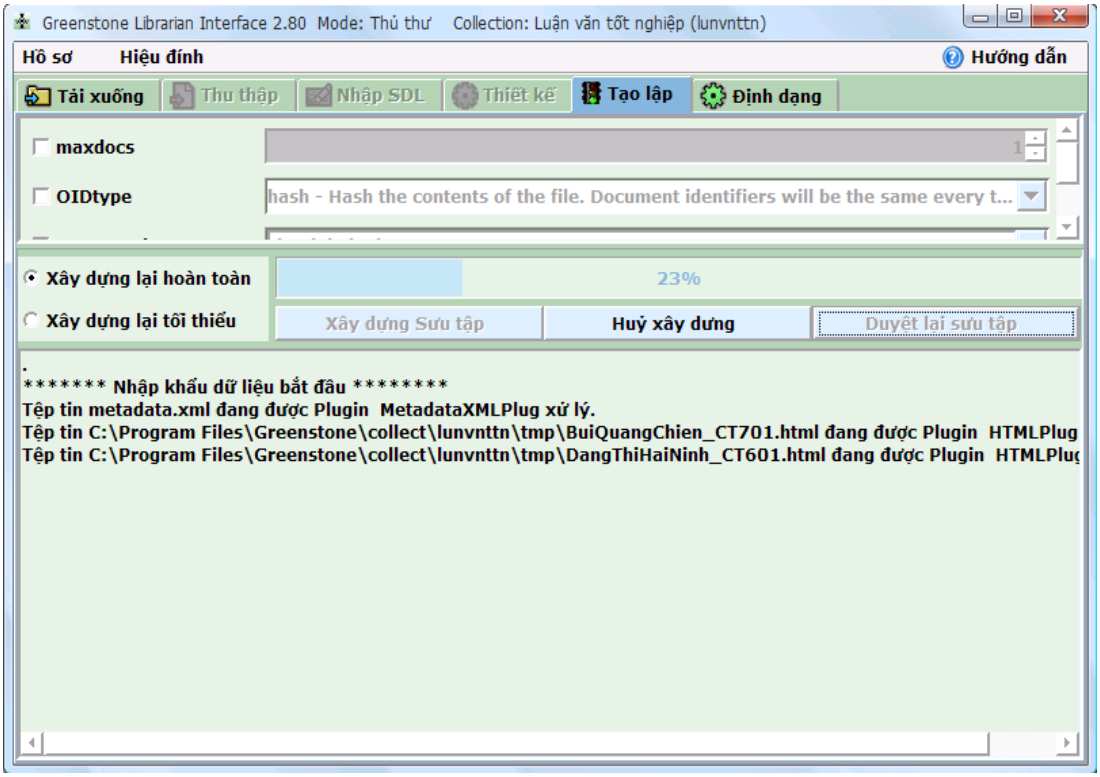
Hình 5.1 – Tạo bộ sưu tập Luận văn tốt nghiệp.



Hình 5.2 – Chọn tài liệu cho bộ sưu tập.

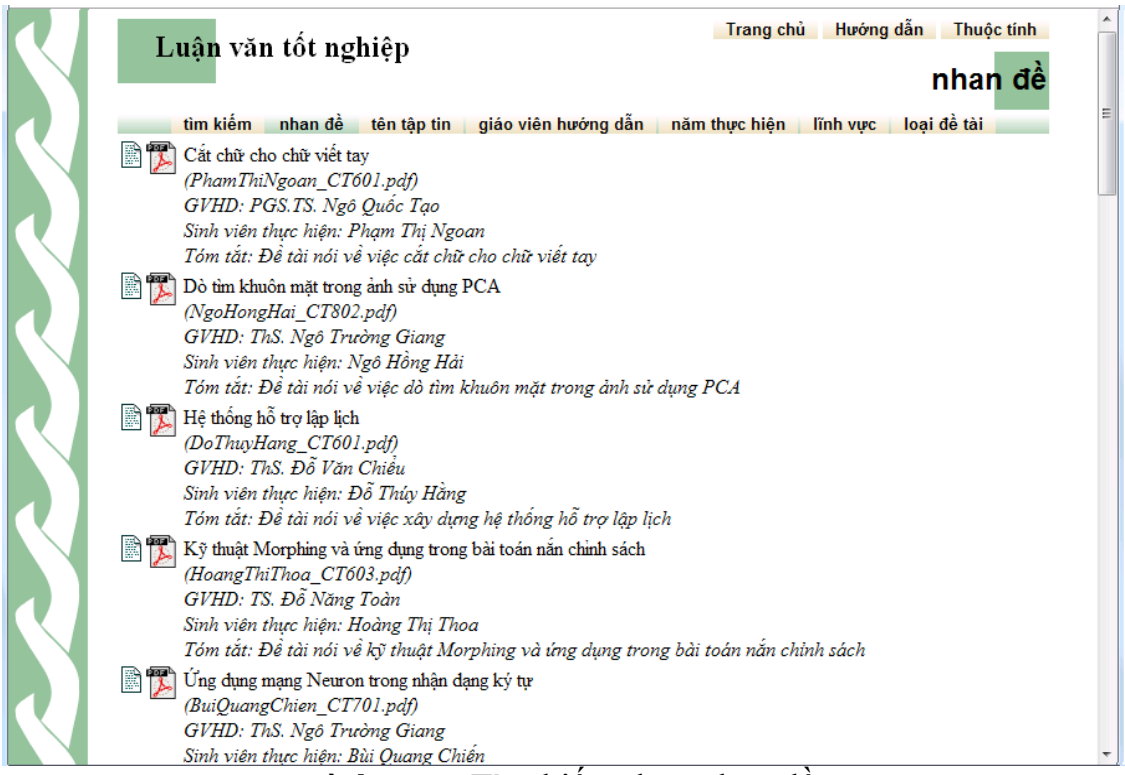


Hình 5.3 – Biên mục cho các tài liệu.

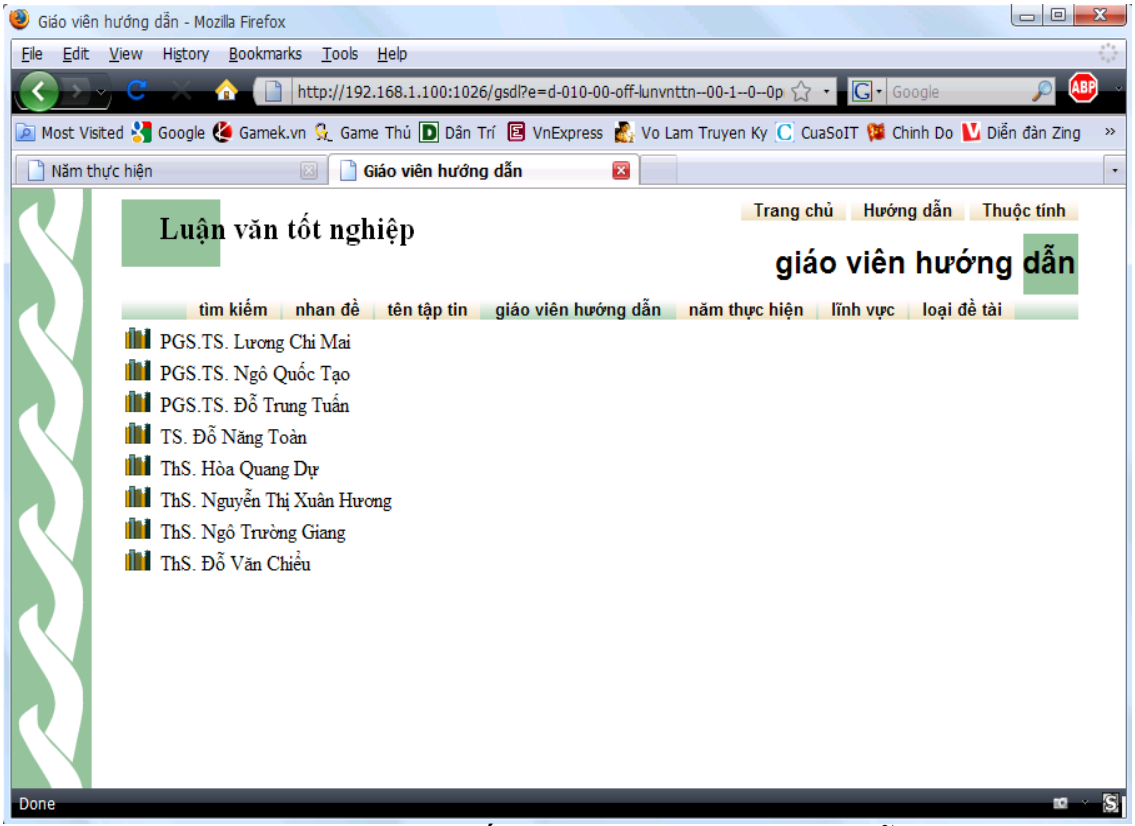


Hình 5.4 – Xây dựng bộ sưu tập.

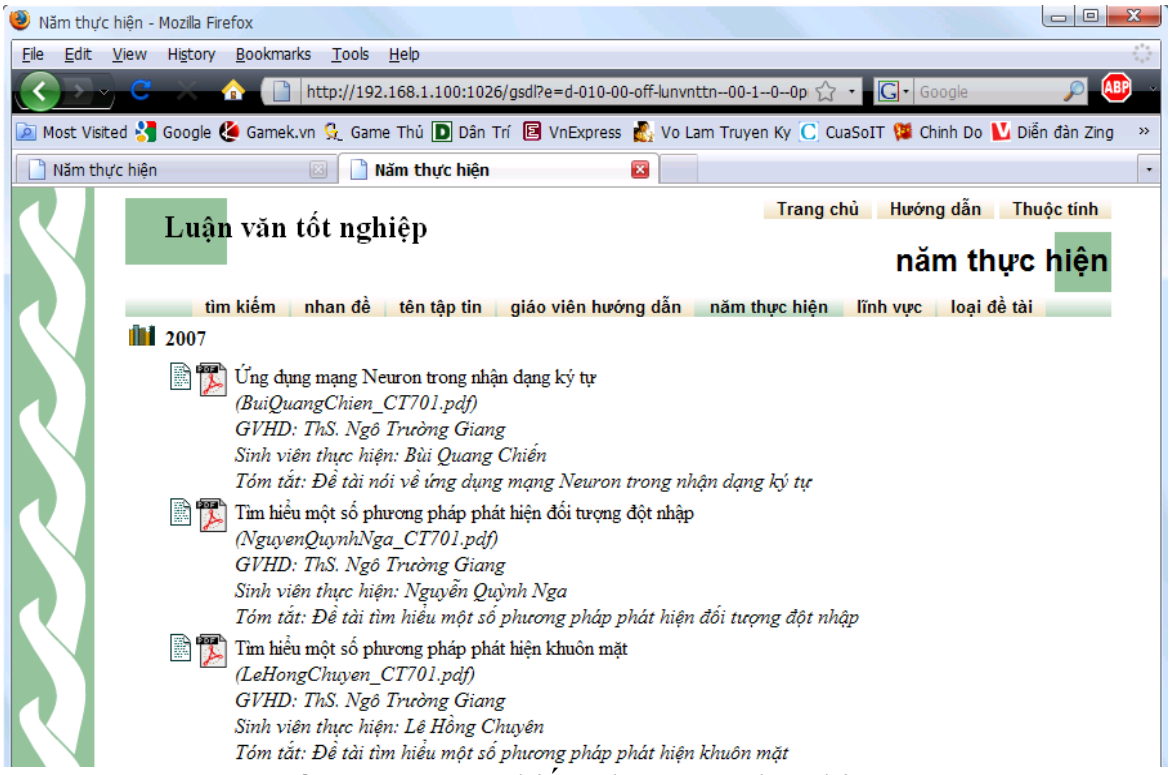
5.3.3. Một số giao diện Web



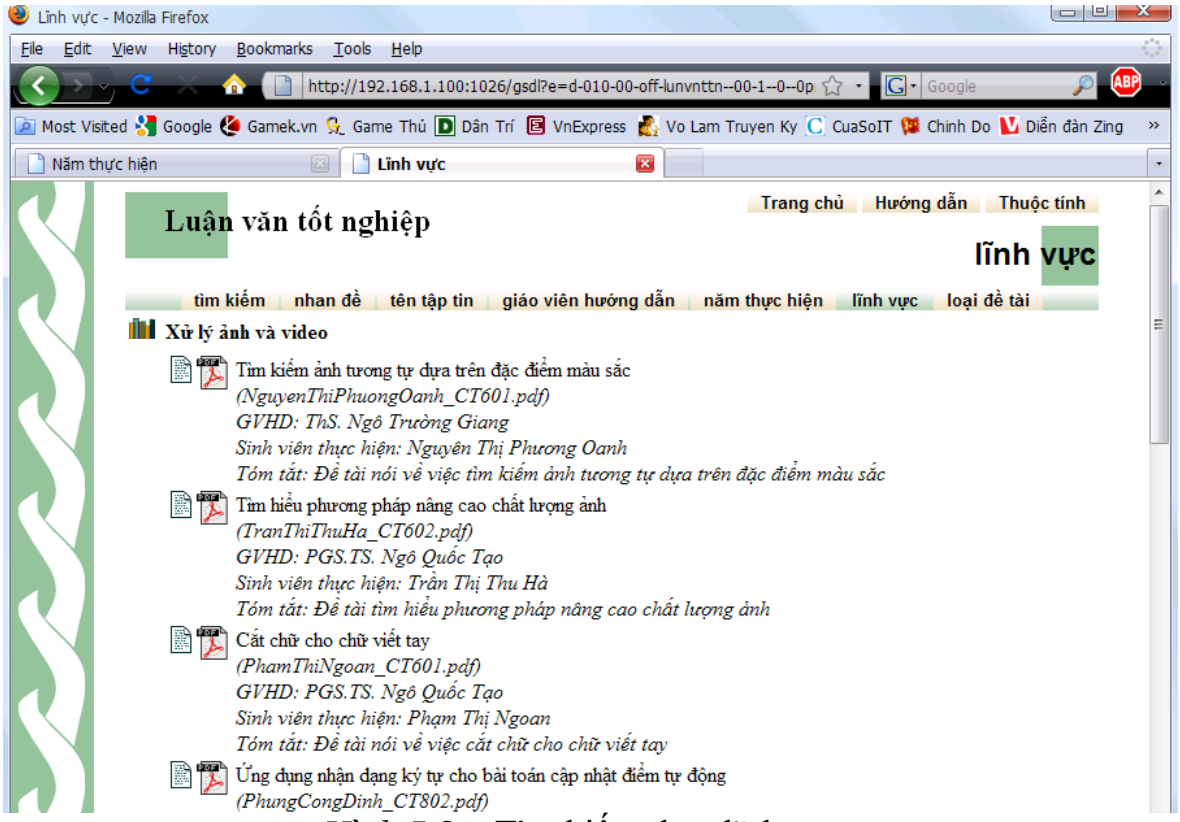
Hình 5.5 – Tìm kiếm theo nhan đề.



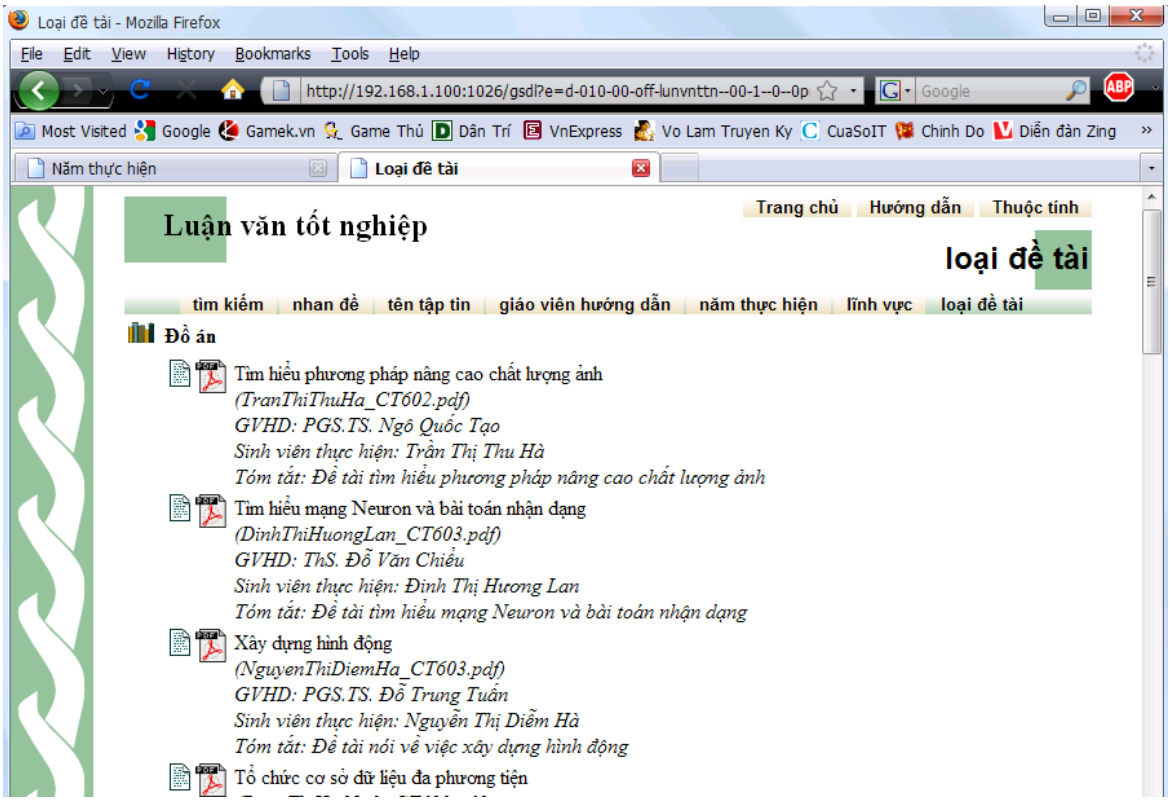
Hình 5.6 – Tìm kiếm theo giáo viên hướng dẫn.



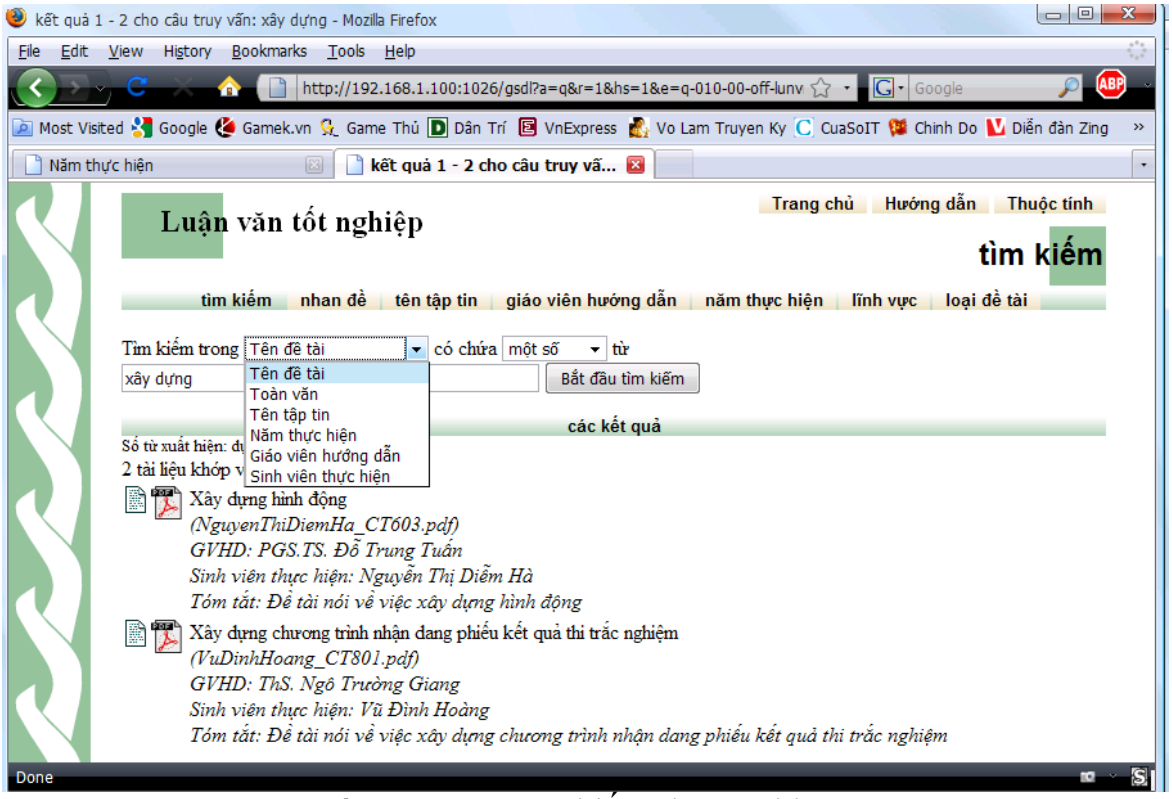
Hình 5.7 – Tìm kiếm theo năm thực hiện.



Hình 5.8 – Tìm kiếm theo lĩnh vực.



Hình 5.9 – Tìm kiếm theo loại đề tài.



Hình 5.10 – Tìm kiếm theo từ khoá.

Kết luận

Đứng trước những yêu cầu về quản lý và tiếp nhận thông tin, sự ra đời của thư viện số đã đóng góp lớn vào công cuộc phát triển tri thức của nhân loại. Khoa học ngày càng tiến bộ con người càng phải nghiên cứu để phát triển thư viện số. Với đề tài "Tìm hiểu phần mềm mã nguồn mở GreenStone", khóa luận này đã trình bày được tổng quan về thư viện số GreenStone, tập trung tìm hiểu vấn đề xây dựng bộ sưu tập và các vấn đề liên quan đến hệ thống web của Greenstone. Qua đó, em đã xây dựng được bộ sưu tập các Luận văn tốt nghiệp của sinh viên ngành Công nghệ Thông tin trường Đại học Dân lập Hải Phòng, đưa bộ sưu tập vào tra cứu thuận lợi với các tiêu chí khác nhau và giao diện web được hiệu chỉnh cho phù hợp.

Trong tương lai em sẽ tiếp tục nghiên cứu thêm để có thể hoàn thiện hơn nữa về giao diện web và xây dựng thêm nhiều bộ sưu tập để ứng dụng vào quản lý được tất cả các tài liệu nhằm xây dựng hoàn thiện Thư viện điện tử trường Đại học Dân lập Hải Phòng.

Tuy nhiên do hạn chế về điều kiện và thời gian, khoá luận sẽ không thể tránh khỏi những thiếu xót. Kính mong được sự đóng góp ý kiến của thầy cô và các bạn để em có thể hoàn thiện hơn đề tài nghiên cứu của mình trong đợt làm khoá luận tốt nghiệp này.

Trân trọng cảm ơn!

Tài liệu tham khảo

[1.] Khóa luận tốt nghiệp của Nguyễn Thành Quy và Lê Hoàng Ngọc Quỳnh – Khoa Công Nghệ Thông Tin – Trường Đại học Khoa học tự nhiên – 2005.

[2.] http://en.wikipedia.org/wiki/Greenstone_(software)

[3.] Bản tin thư viện – Phòng công tác kĩ thuật – Đại hoc Khoa học tự nhiên – Đại học Quốc gia HCM.

[4.] Ứng dụng phần mềm mã nguồn mở GreenStone trong việc tạo lập và phân phối kho tài nguyên số hóa phục vụ giảng dạy và nghiên cứu trong trường đại học – Ths.Nguyễn Thanh Minh – Giám đốc trung tâm Thông tin & Thư viện – ĐH Ngân hàng Thành phố HCM.

[5.] Thư viện và thư viện số - Ths.Nguyễn Minh Hiệp – ĐH Quốc gia TP.Hồ Chí Minh.

[6.] Thư viện số Greenstone – Ian H.Witten & Stefan Boddie – Khoa Công nghệ thông tin – Trường Đại học Waikato – New Zealand.

[7.] http://www.greenstone.org