

Tecnológico de Costa Rica

# III Proyecto Programado

Lenguajes de Programación

# Tabla de contenidos

---

Introducción .....	3
Descripción del problema.....	4
Diseño del programa. ....	5
Decisiones tomadas.....	5
Análisis de resultados .....	6
Manual de usuario.....	7
Bibliografía.....	11

## Introducción

Uno de los principales problemas que tienen los estudiantes a la hora de realizar investigaciones es el manejo de documentos. En una investigación típica, se pueden consultar decenas de documentos con el fin de recopilar toda la información relevante. Algunos de esos documentos están disponibles en Internet, mientras que otros son descargados a las computadoras en formatos como el PDF.

Ahora, la administración de los PDF descargados suele ser una tarea bastante tediosa, ya que si se quiere buscar un documento específico, se tiene que abrir y revisar la información del documento, lo cual es un proceso sumamente lento.

Es por esta razón que se desea desarrollar un sistema de búsqueda de PDF en Lisp, con el objetivo de simplificar la búsqueda de documentos, basado en diversos términos o parámetros.

## Descripción del problema

El objetivo de esta tarea es familiarizarse con el desarrollo de aplicaciones usando el lenguaje Lisp, mediante la creación de una base de datos de información de archivos PDF.

La meta-información o metadatos (datos sobre los datos) de los archivos PDF está codificada según el formato del archivo y la versión de PDF que tenga. Deberán escribir funciones de Lisp que permitan leer un directorio con archivos PDF, y decodificar la información presente en el “header” de cada documento. Para decodificar un archivo binario como un PDF, se deberán escribir rutinas para poder leer archivos binarios; y posteriormente usar esa información para facilitar la búsqueda de documentos, de acuerdo a los metadatos.

Posteriormente, se deberán crear una base de datos que se almacenará en memoria, la cual tendrá una estructura para poder almacenar la información de los PDF, tal como Título, Autor, Asunto, Palabras clave, y Fecha de creación, entre otros. La estructura de la base de datos puede ser definida por lo estudiantes.

La idea es repetir el proceso de extracción de información a todos los pdfs que se encuentren en un directorio específico (el cual podrá ser especificado por el mismo usuario).

Se deberán crear funciones para poder hacer las siguientes consultas sobre la base de datos:

- Mostrar la información de todos los PDF.
- Obtener la información de todos los PDF que tengan un determinado título (el cual será especificado por el usuario).
- Obtener la información de todos los PDF que tengan un determinado autor.
- Obtener la información de todos los PDF que tengan cierta palabra clave.
- Obtener la información de todos los PDF que tengan cierta fecha de creación.

## Diseño del programa.

### *Funciones:*

- Carga path-carpeta: Esta función se encarga de acceder a la carpeta indicada por el usuario y enviando a la función de lectura cada archivo en la carpeta.
- Consulta-general: esta función se encarga de generar una consulta general de todas las instancias a-pdf que se encuentran en la tabla de hash.
- Consulta-titulo: esta función se encarga de realizar una consulta con respecto al titulo.
- Consulta-autor: esta función se encarga de realizar una consulta con respecto al autor.
- Consulta-llaves: esta función se encarga de realizar una consulta con respecto a las llaves.
- Consulta-fecha: esta función se encarga de realizar una consulta con respecto a la fecha.

## Decisiones tomadas

### *Base de Datos*

La base de datos se carga en tiempo de ejecución mediante la función (carga “directorio”).

Esto se basa en la lectura del archivo en binario, los archivos deben estar en formato pdf 1.5 ya que se facilita la identificación y lectura de los metadatos.

Se realiza la lectura y luego se decodifica para poder identificar las etiquetas y obtener la información respectiva.

## Análisis de resultados

A continuación se brinda una reseña de las tareas que se completaron y de las tareas que no se lograron completar, además de algunas propuestas de solución para los problemas que no se pudieron resolver.

Tarea	Estado	Propuesta de solución
Lectura de un directorio con archivos específico	Completa	
Lectura del archivo PDF	Completa	
Extracción de metadatos	Completa	
Extracción de datos de todos los PDF del directorio	Completa	
Lectura del directorio que especificó el usuario	Completa	
Creación de base de datos	Completa	
Almacenamiento de la base de datos en memoria	Completa	
Mostrar la información de todos los PDF	Completa	
Obtener la información de todos los PDF que tengan un determinado título	Completa	
Obtener la información de todos los PDF que tengan un determinado autor	Completa	
Obtener la información de todos los PDF que tengan cierta palabra clave	Completa	
Obtener la información de todos los PDF que tengan cierta fecha de creación	Completa	

## Manual de usuario

Primero se ejecuta gcl (Gnu Common Lisp)

```
dunix@dunix:~/Documentos/Lenguajes/III tareas$ gcl
GCL (GNU Common Lisp) 2.6.7 ANSI Sep 17 2010 12:46:19
Source License: LGPL(gcl,gmp), GPL(unexec,bfd,xgcl)
Binary License: GPL due to GPL'ed components: (XGCL READLINE BFD UNEXEC)
Modifications of this banner must retain notice of a compatible license
Dedicated to the memory of W. Schelter

Use (help) to get some basic information on how to use GCL.
Temporary directory for compiler files set to /tmp/
>
```

Luego se carga el archivo fuente TP3.lisp

```
>(load "TP3.LISP")

Loading TP3.LISP
Finished loading TP3.LISP
T
>
```

Se cuenta con un manual de ayuda.

```
>(ayuda)
<<<<Manual de ayuda al usuario>>>>
Se mostraran las funciones que puede usar
carga path-carpeta
consulta-general
consulta-titulo titulo
consulta-autor autor
consulta-llave llave
consulta-fecha fecha
NIL
>
```

Se visualiza como se cargan los archivos.

```
>(carga "carpeta")
iniciando carga de archivos
cargando..... 1.pdf
cargando..... 2.pdf
cargando..... 3.pdf
cargando..... 4.pdf
cargando..... 5.pdf
(2 3 4 5 6)
```

Consulta general.

```
>(consulta-general)
1.pdf Ejemplo sp Particionamiento) jstradi) Keywords(sp, bases, datos, tec)
2.pdf StoreProcedures) Wayler Mora Salas) Keywords(bases, store, tec)
3.pdf EntidadRelacion) Jose Angel) Keywords(entidad, relacion, bases, datos, tec)
4.pdf ModeloSemantico) JOSE A. STRADI G) Keywords(semantico, bases, datos, tec)
5.pdf INTEGRIDAD) Jose Angel) Keywords(tec, integridad, bases, datos)
NIL
```

El sistema cuenta con valores opcionales, por si el usuario no ingresa un parámetro en la consulta.

```
>(consulta-titulo )
Resultados:.....
1.pdf
2.pdf
3.pdf
4.pdf
5.pdf
NIL
```



## Ejemplos de consulta.

```
>(consulta-general)
1.pdf Ejemplo_sp_Particionamiento) jstradi) D:20120604204528-06'00') Keywords(sp, bases, datos, tec)
2.pdf StoreProcedures) Wayler Mora Salas) D:20120604204316-06'00') Keywords(bases, store, tec)
3.pdf EntidadRelacion) Jose Angel) D:20120604204704-06'00') Keywords(entidad, relacion, bases, datos, tec)
4.pdf ModeloSemantico) JOSE A. STRADI G.) D:20120604204810-06'00') Keywords(semantico, bases, datos, tec)
5.pdf INTEGRIDAD) Jose Angel) D:20120604205011-06'00') Keywords(tec, integridad, bases, datos)
NIL
```

Posteriormente, deberán crear una base de datos que se almacenará en memoria, la cual tendrá una estructura para poder almacenar los metadatos de los pdfs. La estructura de la base de datos será definida por los mismos estudiantes, por lo que deberán explicar con bastante detalle en la documentación todas las decisiones de diseño detrás de la escogencia de la estructura.

Una vez que se tenga la información almacenada, el programa deberá permitir al usuario hacer las siguientes consultas sobre los pdfs:

```
>(consulta-general)
1.pdf Ejemplo_sp_Particionamiento) jstradi) D:20120604204528-06'00') Keywords(sp, bases, datos, tec)
2.pdf StoreProcedures) Wayler Mora Salas) D:20120604204316-06'00') Keywords(bases, store, tec)
3.pdf EntidadRelacion) Jose Angel) D:20120604204704-06'00') Keywords(entidad, relacion, bases, datos, tec)
4.pdf ModeloSemantico) JOSE A. STRADI G.) D:20120604204810-06'00') Keywords(semantico, bases, datos, tec)
5.pdf INTEGRIDAD) Jose Angel) D:20120604205011-06'00') Keywords(tec, integridad, bases, datos)
NIL
```

Posteriormente, deberán crear una base de datos que se almacenará en memoria, la cual tendrá una estructura para poder almacenar los metadatos de los pdfs. La estructura de la base de datos será definida por los mismos estudiantes, por lo que deberán explicar con bastante detalle en la documentación todas las decisiones de diseño detrás de la escogencia de la estructura.

```
>(consulta-general)
1.pdf Ejemplo_sp_Particionamiento) jstradi) D:20120604204528-06'00') Keywords(sp, bases, datos, tec)
2.pdf StoreProcedures) Wayler Mora Salas) D:20120604204316-06'00') Keywords(bases, store, tec)
3.pdf EntidadRelacion) Jose Angel) D:20120604204704-06'00') Keywords(entidad, relacion, bases, datos, tec)
4.pdf ModeloSemantico) JOSE A. STRADI G.) D:20120604204810-06'00') Keywords(semantico, bases, datos, tec)
5.pdf INTEGRIDAD) Jose Angel) D:20120604205011-06'00') Keywords(tec, integridad, bases, datos)
NIL
```

Posteriormente, deberán crear una base de datos que se almacenará en memoria, la cual tendrá una estructura para poder almacenar los metadatos de los pdfs. La estructura de la base de datos será definida por los mismos estudiantes, por lo que deberán explicar con bastante detalle en la documentación todas las decisiones de diseño detrás de la escogencia de la estructura.

Una vez que se tenga la información almacenada, el programa deberá permitir al usuario hacer las siguientes consultas sobre los pdfs:

```
>(consulta-general)
1.pdf Ejemplo_sp_Particionamiento) jstradi) D:20120604204528-06'00') Keywords(sp, bases, datos, tec)
2.pdf StoreProcedures) Wayler Mora Salas) D:20120604204316-06'00') Keywords(bases, store, tec)
3.pdf EntidadRelacion) Jose Angel) D:20120604204704-06'00') Keywords(entidad, relacion, bases, datos, tec)
4.pdf ModeloSemantico) JOSE A. STRADI G.) D:20120604204810-06'00') Keywords(semantico, bases, datos, tec)
5.pdf INTEGRIDAD) Jose Angel) D:20120604205011-06'00') Keywords(tec, integridad, bases, datos)
NIL
```

Posteriormente, deberán crear una base de datos que se almacenará en memoria, la cual tendrá una estructura para poder almacenar los metadatos de los pdfs. La estructura de la base de datos será definida por los mismos estudiantes, por lo que deberán explicar con bastante detalle en la documentación todas las decisiones de diseño detrás de la escogencia de la estructura.

Una vez que se tenga la información almacenada, el programa deberá permitir al usuario hacer las siguientes consultas sobre los pdfs:

```

>(consulta-general)
1.pdf Ejemplo_sp_Particionamiento de 31 bytes D:20120604204520-06'00') Keywords(sp, bases, datos, tec)
2.pdf StoreProcedures) Wayer Mora Salas) da de do D:20120604204316-06'00') Keywords(bases, store, tec)
3.pdf EntidadRelacion) Jose Angel) D:20120604204704-06'00') Keywords(entidad, relacion, bases, datos, tec)
4.pdf ModeloSemantico) JOSE A. STRADI G) D:20120604204810-06'00') Keywords(semantico, bases, datos, tec)
5.pdf INTEGRIDAD) Jose Angel) D:20120604205011-06'00') Keywords(tec, integridad, bases, datos)
NIL

>(consulta-fecha "2009")
Resultados:.....
NIL

>(consulta-fecha "2012")
Resultados:.....
1.pdf
2.pdf
3.pdf
4.pdf
5.pdf
NIL
>

```

Posteriormente, deberán crear una base de datos que se almacenará en memoria, la cual tendrá una estructura para poder almacenar los metadatos de los pdfs. La estructura de la base de datos será definida por los mismos estudiantes, por lo que deberán explicar con bastante detalle en la documentación todas las decisiones de diseño detrás de la escogencia de la estructura.

Una vez que se tenga la información almacenada, el programa deberá permitir al usuario hacer las siguientes consultas sobre los pdfs:

- Mostrar la información de todos los pdfs, usando un formato de tabla con las siguientes columnas: nombre del archivo, título, autor, asunto, palabras clave y fecha de creación
- Obtener la información de todos los pdfs que tengan un determinado título (el cual será

## Bibliografía

- The Common Lisp Cookbook Project. (2007, 28 de enero). **The Common Lisp Cookbook - Strings**. Recuperado el 29 de mayo del 2012, de <http://cl-cookbook.sourceforge.net/strings.html>
- stackoverflow.com. (2010, 19 de diciembre). **An efficient collect function in Common Lisp**. Recuperado el 29 de mayo del 2012, de <http://stackoverflow.com/questions/4480994/an-efficient-collect-function-in-common-lisp>
- es.scribd.com. (2012). **FUNCIONES PARA MANEJAR CADENAS DE TEXTO**. Recuperado el 29 de mayo del 2012, de <http://es.scribd.com/doc/86425568/16/FUNCIONES-PARA-MANEJAR-CADENAS-DE-TEXTO>
- David B. Lamkins. (2001). **Chapter 19 - Streams**. Recuperado el 29 de mayo del 2012, de <http://psg.com/~dlamkins/sl/chapter19.html>
- LispWorks Ltd. (2005). **Function READ-BYTE**. Recuperado el 29 de mayo del 2012, de [http://www.lispworks.com/documentation/lw60/CLHS/Body/f\\_rd\\_by.htm](http://www.lispworks.com/documentation/lw60/CLHS/Body/f_rd_by.htm)
- sno.phy.queensu.ca. (2011, 1 de diciembre). **PDF Tags**. Recuperado el 31 de mayo del 2012, de <http://www.sno.phy.queensu.ca/~phil/exiftool/TagNames/PDF.html#Metadata>
- L. Leurs. (2000). **La estructura y manipulación de los ficheros PDF**. Recuperado el 31 de mayo del 2012, de [http://gusgsm.com/la\\_estructura\\_y\\_manipulacion\\_de\\_los\\_ficheros\\_pdf](http://gusgsm.com/la_estructura_y_manipulacion_de_los_ficheros_pdf)
- cliki.net. (2012). **cl-binary-file**. Recuperado el 01 de junio del 2012, de <http://www.cliki.net/cl-binary-file>
- stackoverflow.com. (2010, 18 de noviembre). **How to read a pdf file using lisp**. Recuperado el 01 de junio del 2012, de <http://stackoverflow.com/questions/4214403/how-to-read-a-pdf-file-using-lisp>
- es.scribd.com. (2012). **MANUAL PROGRAMACION LISP**. Recuperado el 01 de junio del 2012, de <http://es.scribd.com/doc/2418100/MANUAL-PROGRAMACION-LISP>
- Peter Seibel. (2009). **Practical Common Lisp**. Recuperado el 02 de junio del 2012, de <http://www.gigamonkeys.com/book/>