



**UNIVERSIDAD CATOLICA
DE LA SANTISIMA CONCEPCION**
FACULTAD DE INGENIERIA

**HERRAMIENTA COMPUTACIONAL DE APOYO A LA ETAPA DE BÚSQUEDA
DE ESTUDIOS PRIMARIOS EN LAS BASES DE DATOS CIENTÍFICAS.**

PAOLA PATRICIA FREDES PEÑA

PROYECTO DE TÍTULO PRESENTADO A LA FACULTAD DE INGENIERÍA DE LA
UNIVERSIDAD CATÓLICA DE LA SANTÍSIMA CONCEPCIÓN, PARA OPTAR
AL TÍTULO DE INGENIERO CIVIL INFORMÁTICO.

Profesor Guía:

PEDRO ROSSEL CID.

Comisión informante:

CLAUDIA MARTÍNEZ ARANEDA.

Concepción, CHILE
2017

Resumen del Proyecto de Título

Contexto: Revisión Sistemática es una metodología para sintetizar la evidencia científica para responder a una pregunta delimitada que utiliza unos procesos específicos y sistemáticos para identificar, seleccionar y evaluar objetivamente la investigación original, así como para extraer e interpretar los datos de los estudios incluidos. Las fases de una RS se vuelven una tarea muy lenta y tediosa si se realiza manualmente, es algo que les quita mucho tiempo a los investigadores efectuar.

De lo anterior, surge la necesidad de desarrollar un software que permita automatizar una fase del proceso de la revisión, y a su vez eliminar tareas que pueden ser realizadas automáticamente por un sistema computacional, dando como resultado que los investigadores dediquen tiempo a otras actividades de una RS.

Este proyecto de título aborda el desarrollo de una Herramienta Computacional, centrándose en una interfaz práctica, que principalmente facilite la labor de extracción de los metadatos de los artículos que se encuentran en las bases de datos científicas que realizan los investigadores al momento de efectuar una RS, es vital que esta herramienta se enfoque a que el tiempo de extracción de las referencias bibliográficas sea menor que estar realizando este proceso manualmente y además que la información que se extraiga de la herramienta sea consistente con la información de las bases de datos científicas.

Abstract

Context: Systematic Review (SR) is a methodology use to synthesize scientific evidence in order to answer a specific question, it uses specific process to identify, select and evaluate objectively the original research, also to extract and interpret the data of the included research. The phases of a SR are slow a tedious when done manually, this take a lot of the researches time.

From this comes the necessity to develop a software that allow us to automate a phase of the reviewing process, and also delete tasks that could be done by a computer, this will allow the researchers to use time in others phases of RS.

This project approaches the development of a computer tool, focusing in the practical interphase, that will allow to facilitate the extraction of metadata from scientific data base's articles. This tool will focus in lowering the extraction time of bibliographic references, so is less than doing it manually, also the information that this tool extract must be consistent with the information in the data base.

Tabla de contenido

Capítulo 1	1
Introducción	1
1.1. Objetivo General	3
1.2. Objetivos Específicos.....	3
1.3. Justificación del Problema.....	4
1.4. Delimitación del problema	5
1.5. Metodología a utilizar	7
1.5.1 Etapa de aprendizaje.....	7
1.5.2. Método de desarrollo.....	8
1.5.3. Desarrollo de los incrementos	9
1.5.4. Validación de la Herramienta.....	10
1.6. Presentación de capítulos siguientes	11
Capítulo 2	12
Marco Teórico	12
2.1. Revisiones sistemáticas	12
2.2. Mapeos sistemáticos	16
2.3. Formato BibTeX.....	19
2.3.1. Tipos de registros bibliográficos.....	21
2.4. Web Scraping.....	24
2.4.1. Proceso de web Scraping de la herramienta.....	24
2.4.2. Conceptos para entender web scraping	25
2.4.3. Problemas con web Scraping.	31
2.5. Herramientas a utilizar	32
2.5.1. Lenguaje de programación PHP	32
2.5.2. Simple HTML DOM Parser	33
2.5.3. HTML	35
2.5.4. CSS.....	36

Capítulo 3	37
Estado del Arte	37
3.1. Base de Datos Epistemonikos.....	37
3.2. Mendeley.....	40
3.3. JabRef	44
3.4. StArt.....	46
3.4. Sobre las aplicaciones encontradas.	53
Capítulo 4	54
Desarrollo de la herramienta Computacional.....	54
4.1 Primer Incremento de la Herramienta.....	54
4.1.1. Búsquedas en las bases de datos y Resultados.	55
4.1.2. Análisis de las bases de datos.....	75
4.1.3. Estructura de las bases de datos	79
4.1.4. Procesamiento.....	81
4.2. Segundo Incremento de la Herramienta.....	83
4.2.1. Búsqueda por fecha	83
4.2.2. Construcción de interfaz gráfica	84
4.2.3. Guardar metadatos en un archivo	86
Capítulo 5	87
Pruebas del Sistema	87
5.1. Resultados entre la cantidad de referencias bibliográficas de las bases de datos v/s la herramienta computacional para una búsqueda en el título.	88
5.2. Resultados entre la cantidad de referencias bibliográficas de las bases de datos v/s la herramienta computacional para una búsqueda en el Abstract.	89
5.3. Resultados entre la cantidad de referencias bibliográficas de las bases de datos v/s la herramienta computacional para una búsqueda en el Keyword.....	90
5.4. Resultados entre la cantidad de referencias bibliográficas de las bases de datos v/s la herramienta computacional para una búsqueda en el título y abstract.	91
5.5. Resultados entre la cantidad de referencias bibliográficas de las bases de datos v/s la herramienta computacional para una búsqueda en el título y keyword.....	92
5.6. Resultados entre la cantidad de referencias bibliográficas de las bases de datos v/s la herramienta computacional para una búsqueda en el abstract y keyword.	93

5.7. Resultados entre la cantidad de referencias bibliográficas de las bases de datos v/s la herramienta computacional para una búsqueda en el título, abstract y keyword.....	94
Capítulo 6	96
Conclusión	96
6.1. Principales dificultades.....	96
6.2. Problemas durante el proceso de extracción.....	97
6.3. Objetivos alcanzados.....	99
Bibliografía	100
Apéndice A	102
Estructura HTML de las Páginas Web.....	102
A.1. Taylor and Francis	102
B.2. Emerald.	108
B.3. Wiley & Sons.	113
B.4. SpringerLink.....	118
B.5. IEEEXplore.	122
B.6. Scopus.....	126
B.7. Science Direct.	129
Apéndice B	134
Especificación de requisitos	134
B.1. Descripción general	134
B.2. Actores de la Herramienta	134
B.3. Requisitos del usuario	135
B.4. Requisitos del sistema	135
B.5. Requisitos no funcionales	136
B.6. Diagrama de flujos de datos.....	149
B.6.1. Consultar por artículos en ACM.	149
B.6.2. Consultar por artículos en Wiley & Sons.....	150
B.6.3. Consultar por artículos en Springer.	151
B.6.4. Consultar por artículos en Emerald.	152
B.6.5. Consultar por artículos en Taylor and Francis.	153
B.6.6. Consultar por artículos en Scopus.	154

Apéndice C	155
Código Fuente del Segundo Incremento de la Herramienta	155
C.1. Archivo: server.php	155
C.2. Archivo: acm.php	158
C.3. Archivo:taylorfrancis.php.....	163
C.4. Archivo:springer.php.....	169
C.5. Archivo:emerald.php	173
C.6. Archivo:wileysons.php	176
C.7. Archivo:scopus.php	180
Apéndice D	184
Manual de Usuario.....	184

Índice de Figuras

Figura 1: Metodología de Kitchenham para realizar una revisión sistemática.....	2
Figura 2: Número de revisiones sistemáticas que fueron realizadas y publicadas en la base de datos Science Direct.....	4
Figura 3: Modelo Incremental Pressman.....	9
Figura 4: Formato general BibTeX.....	19
Figura 5: Ejemplo tipo de registro "Article".....	21
Figura 6: Ejemplo tipo de registro "Book".....	22
Figura 7: Ejemplo tipo de registro "InCollection".....	22
Figura 8: Ejemplo tipo de registro "InProceedings".....	23
Figura 9: Ejemplo del proceso de "web scraping".....	25
Figura 10: Esquema de protocolo HTTP.....	27
Figura 11: Ejemplo de cabecera de petición.....	28
Figura 12: Estructura de solicitud HTTP.....	28
Figura 13: Representación Web, HTML, DOM.....	30
Figura 14: Árbol DOM creado a partir de un documento HTML.....	34
Figura 15: Búsqueda Avanzada Epistemonikos.....	38
Figura 16: Tipos de Filtros de Epistemonikos.....	38
Figura 17: Exportar Resultados Epistemonikos.....	39
Figura 18: Mendeley incrustado en la barra de navegación.....	41
Figura 19: Búsqueda en Science direct.....	42
Figura 20: Uso de la herramienta "Web Importer".....	43
Figura 21: Opciones de bases de datos para realizar búsquedas en línea.....	44
Figura 22: Búsqueda en la bases de datos IEEEExplore.....	45
Figura 23: Búsqueda en la bases de datos Springer.....	46
Figura 24: Ejemplo de llenado del protocolo en la fase de planificación.....	48
Figura 25: Ejemplo ingreso string de búsqueda.....	49
Figura 26: Importar resultados obtenidos de la búsqueda en ACM.....	50
Figura 27: Menú fase selección.....	51
Figura 28: Panel de un estudio en específico.....	52

Figura 29: Búsqueda por Campos en Emerald.	64
Figura 30: Búsqueda por campo en Wiley & Sons	66
Figura 31: Búsqueda por campos en Science Direct.	72
Figura 32: Búsqueda por campos en Scopus.....	74
Figura 33: Búsqueda avanzada de ACM.....	76
Figura 34: Página de resultados obtenidos en la búsqueda de ACM.....	77
Figura 35: Ejemplo de referencia bibliográfica de ACM.....	78
Figura 36: Extracto de código (cantidad de resultados) de ACM.....	80
Figura 37: Extracto de código para obtener todos los ítems de ACM.....	80
Figura 38: Extracto de código para obtener el doi de ACM.	81
Figura 39: Extracto de código del archivo acm.php.....	82
Figura 40: URL + filtro fecha ACM.	83
Figura 41: Ingreso cadena de búsqueda.	84
Figura 42: Seleccionar bases de datos.	84
Figura 43: Filtros de la Herramienta Computacional.	85
Figura 44: Guardar en un archivo.....	86
Figura 45: Página Taylor and Francis.....	103
Figura 46: Página de resultados de Taylor and Francis.	104
Figura 47: Extractos de códigos de la Pagina Taylor and Francis.	105
Figura 48: Peticiones HTTP solicitar referencias bibliográficas Taylor and Francis.....	106
Figura 49: Página Emerald.....	109
Figura 50: Extractos de códigos de página Emerald.....	110
Figura 51: Peticiones HTTP solicitar referencias bibliográficas página Emerald.....	111
Figura 52: Página Wiley & Sons.....	114
Figura 53: Extractos de códigos de la página Wiley & Sons.	116
Figura 54: Peticiones HTTP solicitar referencias bibliográficas Wiley & Sons.....	117
Figura 55: Página de SpringerLink.	119
Figura 56: Extractos de códigos de la página SpringerLink.	120
Figura 57: Peticiones HTTP solicitar referencias bibliográficas SpringerLink.	121
Figura 58: Página IEEEExplore.....	123

Figura 59: Extractos de códigos de la página IEEEExplore.....	124
Figura 60: Peticiones HTTP solicitar referencias bibliográficas IEEEExplore.....	125
Figura 61: Página de Scopus.....	127
Figura 62: Extracto de código de la página Scopus.	128
Figura 63: Página Science direct.....	130
Figura 64: Extracto de código de Science Direct.	131
Figura 65: Peticiones HTTP solicitar referencias bibliográficas Science Direct.	132
Figura 66: Página de bienvenida de WampServer	185
Figura 67: Interfaz Gráfica de la herramienta computacional.	186
Figura 68: Ingreso de la cadena de búsqueda.....	187
Figura 69: Mensaje de alerta si no ingresa cadena de búsqueda.	187
Figura 70: Mensaje de alerta si no se encuentran resultados de la búsqueda.....	187
Figura 71: Seleccionar bases de datos.	188
Figura 72: Mensaje de alerta bases de datos.....	188
Figura 73: Seleccionar Campos.	189
Figura 74: Mensaje de alerta para los campos.	190
Figura 75: Seleccionar rango de Fechas.	190
Figura 76: Resultados Guardados en un Archivo.	191

Índice de Tablas

Tabla 1: Cuadro Comparativo de las Revisiones Sistemáticas y Mapeos Sistemáticos.....	17
Tabla 2: Campos de cabecera del protocolo HTTP.....	29
Tabla 3: Formato de cadenas de búsquedas por campo en la base de datos ACM.....	56
Tabla 4: Porcentaje de resultados correctos arrojados por ACM.	59
Tabla 5: Formato de cadenas de búsquedas por campos en Taylor and Francis.....	60
Tabla 6: Porcentaje de resultados correctos arrojador por Taylor and Francis.	62
Tabla 7: Formato de cadena de búsqueda por campo título en Springer.....	63
Tabla 8: Porcentaje de Resultados correctos arrojador por Springer.....	63
Figura 29: Búsqueda por Campos en Emerald.	64
Tabla 9: Porcentaje de resultados correctos arrojados por Emerald.....	65
Tabla 10: Porcentaje de resultados correctos arrojados por Wiley & Sons.....	67
Tabla 11: Formato de cadenas de búsquedas por campo en IEEEXplore.	68
Tabla 12: Porcentaje de resultados correctos arrojados por IEEEXplore.....	71
Tabla 13: Porcentaje de resultados correctos arrojados por Science Direct.	73
Tabla 14: Porcentaje de resultados correctos arrojados por Scopus.....	74
Tabla 15: Pruebas del sistema de una búsqueda en el título.....	88
Tabla 16: Pruebas del sistema de una búsqueda en el abstract.....	89
Tabla 17: Pruebas del sistema de una búsqueda en el keyword.	90
Tabla 18: Pruebas del sistema de una búsqueda en el título y abstract.....	91
Tabla 19: Pruebas del sistema para el título y keyword.....	92
Tabla 20: Pruebas del sistema para el abstract y keyword.	93
Tabla 21: Pruebas del sistema para el título, abstract y keyword.	94

Capítulo 1

Introducción

En las últimas décadas se ha podido observar la enorme y variada cantidad de artículos científicos existentes, lo que dificulta cada vez más a los investigadores conocer por completo la información. La cantidad de producción científica que se dispone es demasiada extensa y además ésta crece a un ritmo vertiginoso, resultando difícil mantenerse actualizado en los temas de interés. En este escenario en el que los investigadores se encuentran, las revisiones sistemáticas son el mejor esfuerzo para recopilar, sintetizar y analizar de manera confiable y eficaz la evidencia científica sobre un tema en particular, con el objetivo de establecer conclusiones más generales que los estudios individuales o como un comienzo de actividades de investigación futura.

De acuerdo a la definición de Kitchenham, la revisión sistemática es un método que permite a los especialistas obtener resultados relevantes y cuantificables; esto lleva a la identificación, selección y producción de evidencias basadas en la investigación de un tópico en particular [5].

En la Figura 1 se muestran las fases principales de una revisión sistemática, las actividades contenidas en cada una de ellas, y también se señala la actividad “Selección de estudios”, para poder resaltar en donde la herramienta computacional permite la automatización del protocolo que sigue una revisión sistemática.

En la sección 2.1 se encuentra explicado cada una de las fases principales de una revisión sistemática.

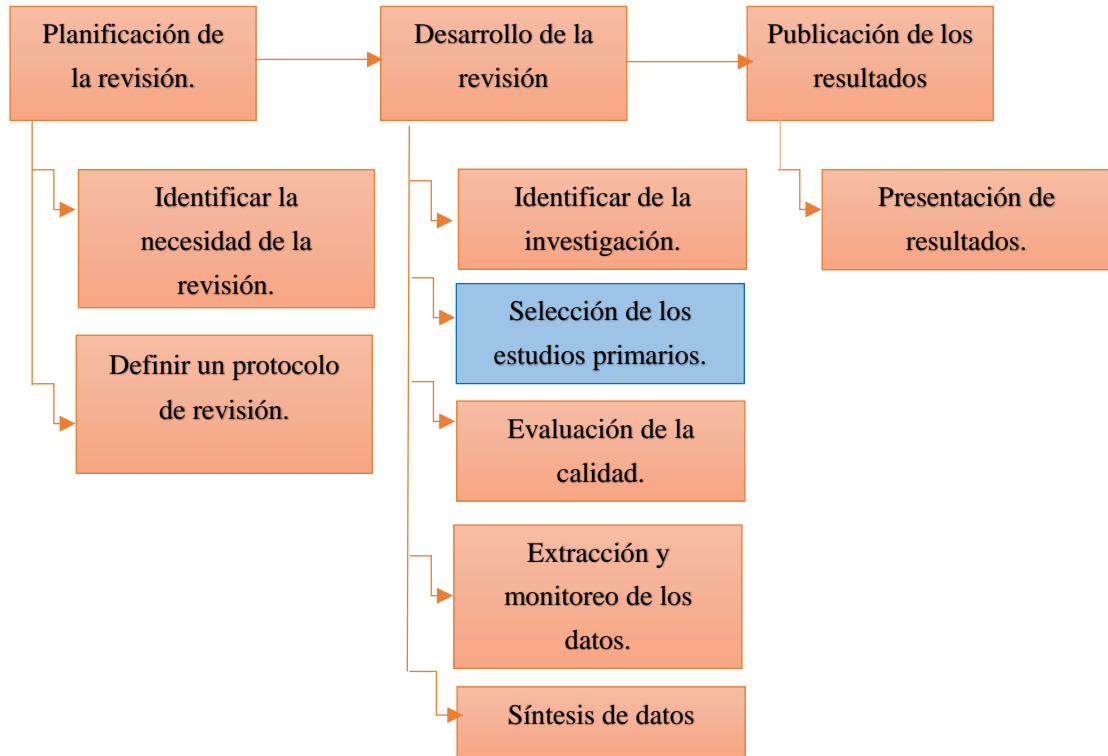


Figura 1: Metodología de Kitchenham para realizar una revisión sistemática (Fuente Propia).

El mayor problema que existe en una revisión sistemática es que el proceso es largo y engorroso, requiere de dedicación y tiempo ya que generalmente las fases de una revisión sistemática se realizan manualmente.

A raíz de estas necesidades es donde surge este proyecto, cuyo objetivo principal es desarrollar una herramienta computacional que apoye a la etapa de “Selección de los estudios” del proceso de la revisión sistemática. Esta herramienta será capaz de recopilar variados metadatos de los artículos en distintas bases de datos científicas que sean relevantes al tema de interés, para que luego estos metadatos sean ingresados a un archivo con formato BibTeX, permitiéndoles a los investigadores acceder a la información de manera rápida sin que ellos tengan que estar manualmente accediendo a la bases de datos científicas y seleccionando las referencias de los artículos de interés, ayudando a los investigadores agilizar el proceso de búsqueda, sin tener la necesidad de invertir demasiado tiempo en esta etapa.

1.1. Objetivo General

Construir una herramienta computacional que apoye la etapa de búsqueda de estudios primarios en las bases de datos científicas.

1.2. Objetivos Específicos

- 1.- Estudiar la forma en que se realizan las revisiones sistemáticas y mapeos sistemáticos.
- 2.- Revisar Bases de datos científicas
- 3.- Estudiar conceptos para el desarrollo
- 4.- Desarrollar Herramienta computacional.

1.3. Justificación del Problema

La necesidad de desarrollar esta herramienta computacional surge debido a la cantidad de revisiones y mapeos sistemáticos realizadas por los investigadores, en particular en el área de Ingeniería de Software.

En la Figura 2 se ve claramente que a medida que pasan los años las revisiones sistemáticas han ido en aumento. El número de revisiones sistemáticas que se muestra en la Figura 2 es solo de la base de datos Science Direct en el área Computer Science [2].

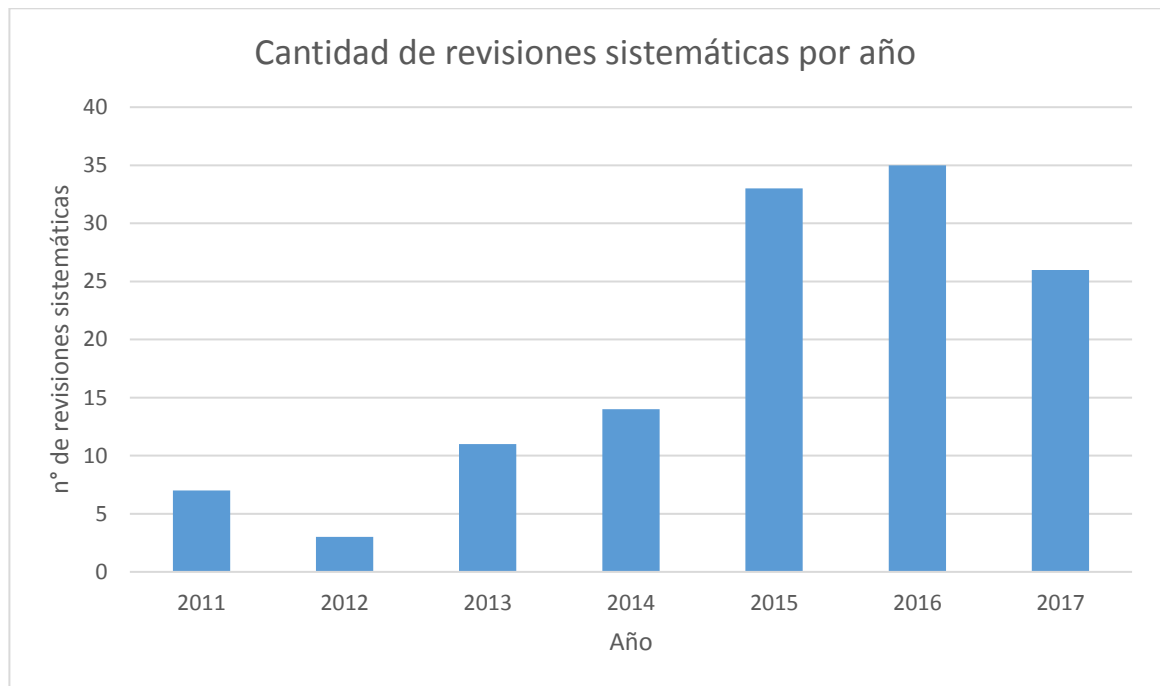


Figura 2: Número de revisiones sistemáticas que fueron realizadas y publicadas en la base de datos Science Direct.

Este hecho pone de manifiesto la importancia creciente de las revisiones sistemáticas que se están realizando en todo el mundo. Por esta razón, esta herramienta se convertiría en un apoyo a dicha labor, especialmente en la búsqueda de los artículos. Si bien existen varias herramientas que apoyen a las distintas etapas de las revisiones sistemáticas son pocas las herramientas que apoyen la etapa llamada “Selección de los estudios”, por lo que esto sería otra razón para que se desarrolle esta herramienta.

Por otro lado, cabe mencionar que esta herramienta busca apoyar a los investigadores de tal forma que su usabilidad les brinde algunos beneficios como reducir el tiempo en el desarrollo del protocolo, favoreciendo la ejecución de una revisión sistemática y mejorando en el acceso a los datos agilizando el proceso de “Selección de los estudios”.

1.4. Delimitación del problema

El proyecto está orientado solo a la etapa de “Selección de los estudios”, recopilando los datos de los artículos en un formato llamado BibTeX; no se pretende organizar y estructurar dichos datos (autores, año, nombre, páginas), si no que al ingresar el string de búsqueda en el motor de búsqueda en las diferentes bases de datos científicas, que recopile las referencias de los artículos en un archivo .bib.

La herramienta tampoco se preocupa de demostrar que se han extraído correctamente los datos; el motor de búsqueda no distingue entre palabras coherentes o incoherentes.

Las bases de datos sobre las que se decidirá hacer el proyecto serán las siguientes, al principio se ocuparán estas ya que son las más ocupadas en las ciencias de la computación.

- ACM
- Springer Link
- Scopus
- Wiley & Sons
- Taylor and Francis
- Emerald

1.5. Metodología a utilizar

Con respecto a la metodología que se ocupó durante el proyecto se dividió en tres etapas: la primera de ellas es la etapa de aprendizaje, que refiere a todos los temas necesarios para poder desarrollar la herramienta computacional, la segunda etapa, es el método de desarrollo que se va aplicar, y por último la tercera etapa trata la validación de la herramienta.

1.5.1 Etapa de aprendizaje

Para el presente proyecto, se realizará un estudio a fondo de revisiones sistemáticas y mapeos sistemáticos, y se deberá entender cada una de las etapas, en especial la etapa “Selección de los estudios”, que es donde la herramienta brindará ayuda; por otro lado se deberá estudiar cómo se generará un string de búsqueda con palabras claves, y será necesario realizar búsquedas en las distintas bases de datos en las que se trabajarán. Posteriormente, se deberá estudiar la forma en que se extraerán los metadatos de los artículos en las distintas bases de datos científicas. La forma en que se extraerán dichos metadatos será mediante “web scraping”, que es una herramienta que permite la extracción de información de distintas páginas webs, permitiendo que el proceso sea de forma automatizada.

Una vez que se extraigan los datos de los artículos éstos serán guardados en un archivo, es por esto que es necesario conocer y estudiar el formato BibTeX que es donde se guardarán las referencias de los artículos.

Por otro lado, para desarrollar la herramienta, es fundamental conocer el lenguaje de programación PHP ya que este es el que contiene la biblioteca robot necesaria para poder construir la herramienta.

1.5.2. Método de desarrollo

El método de desarrollo a aplicar en este proyecto será el modelo Incremental. Según Pressman 2010, “*El modelo de proceso incremental se centra en que en cada incremento se entrega un producto que ya opera. Los primeros incrementos son versiones desnudas del producto final, pero proporcionan capacidad que sirve al usuario y también le dan una plataforma de evaluación [3]*”. Como se muestra en la Figura 3, obtenido desde [3], se puede observar que, al final de cada iteración se le entrega al usuario una versión completa y mejorada del producto, y esto permite que los usuarios puedan ver en una etapa temprana que tan bien el software favorece su trabajo. Como se obtiene una rápida retroalimentación con éste, el desarrollador podrá conseguir nuevas ideas de requerimiento, distinguiendo los cambios que puede tener el software. Esto quiere decir que si hay algún error en algunos de los incrementos o no están completos, se modificarán algunas de las funciones que necesita el usuario, con el objetivo de reflejar la nueva comprensión de los requerimientos. Por estas características, se ocupará esta metodología con la finalidad de precisar la forma en que se va a desarrollar la herramienta; se necesitará que en cada incremento o versión el desarrollador adquiera retroalimentación del usuario cada vez que le va entregando los incrementos, y así el cliente podrá ir evaluando el desarrollo de la herramienta en una etapa temprana para constatar si se entrega lo que se requiere.

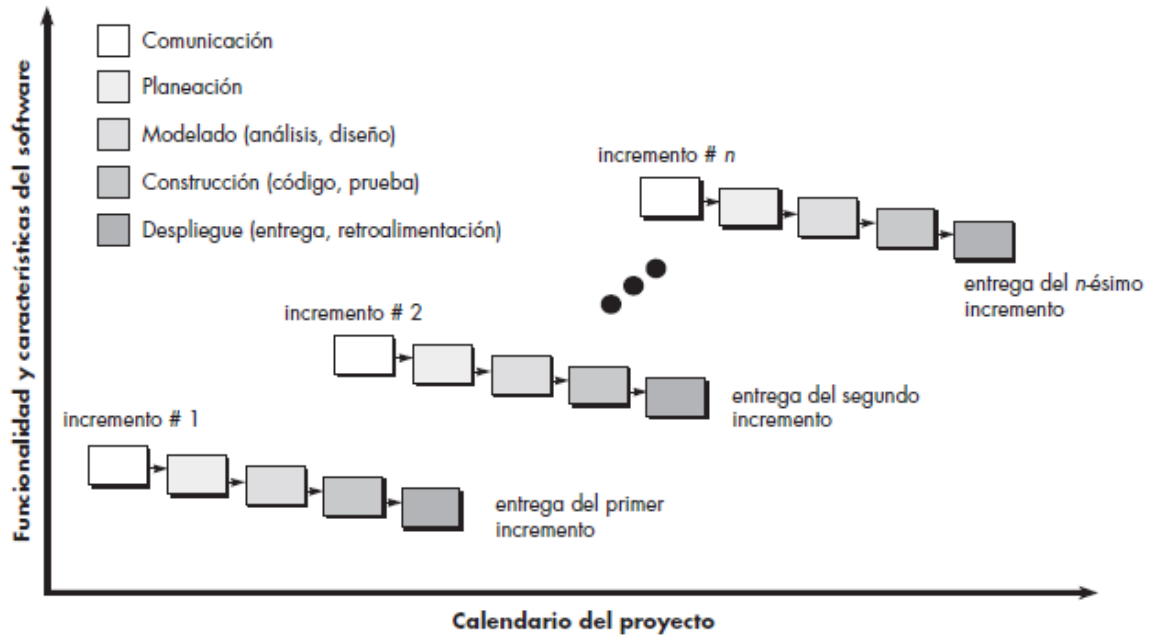


Figura 3: Modelo Incremental Pressman.

1.5.3. Desarrollo de los incrementos

Se procederá a desarrollar el primer incremento del sistema, en esta etapa del desarrollo se tendrán que realizar búsquedas en las distintas bases de datos, seguidamente se definirá correctamente el string de búsqueda para cada una de ellas y posterior a esto se analizarán cada una de las páginas webs (bases de datos). Luego, se efectuará la conexión con cada una de éstas. Posterior a esto, se recopilarán todas las referencias de los resultados de la búsqueda. Enseguida, se aplicará una etapa de pruebas para apreciar la consistencia de la cantidad de archivos arrojados en las bases de datos con la cantidad de metadatos que se extraerán con la herramienta computacional. Finalizada esta etapa, se desarrollará el segundo incremento del sistema, en esta etapa se ingresarán los metadatos de los artículos en un archivo con formato .BIB y se aplicarán las pruebas necesarias para ver la consistencia de la

cantidad de metadatos que se guardarán en el archivo con la cantidad de metadatos de la búsqueda que se realizó y por último se implementará la interfaz gráfica para luego ser validada con el usuario.

1.5.4. Validación de la Herramienta

Como instancia final, cada uno de los incrementos que se desarrollarán pasará por una etapa de pruebas, proceso que permitirá determinar si el sistema realmente hará lo que se desea, además de descubrir defectos antes de la utilización

Una vez que la herramienta haya sido construida se validará con revisiones sistemáticas que se realizaron anteriormente, la cantidad de artículos que arrojan las bases de datos tienen que ser consistentes con la cantidad de metadatos que extrae la herramienta, además esto no bastará, se tendrá que hacer una validación si los metadatos que se extraen son los correctos.

1.6. Presentación de capítulos siguientes

Este informe se encuentra dividido en distintos capítulos, explicando una situación diferente experimentada durante el desarrollo del proyecto.

- Capítulo 2: Marco Teórico.- En este capítulo se muestran todos los conceptos relevantes y necesarios que conciernen al proyecto, que permitirá comprender el tema elegido para su posterior desarrollo.
- Capítulo 3: Estado del Arte.- En este capítulo se darán a conocer aquellos proyectos que existen hoy en día que son similares a la herramienta propuesta, para poder analizar las diferencias y similitudes con respecto a la herramienta computacional.
- Capítulo 4: Evolución del desarrollo de la herramienta Computacional.- En este capítulo se expone detalladamente el proceso de construcción de los incrementos de la herramienta.
- Capítulo 5: Pruebas del sistema.- En este capítulo se van a realizar pruebas en el sistema para poder ver la consistencia de la cantidad de datos de los artículos que se extraigan, con la cantidad de datos que se realizan en las búsquedas de las bases de datos científicas.
- Capítulo 6: Conclusiones.- En este capítulo se hablarán de las observaciones del sistema, los problemas que tuvo la herramienta y explicar cómo se fueron solucionando. Y por último se analizarán los requisitos futuros por si el sistema más adelante cambia.

Capítulo 2

Marco Teórico

En el marco teórico del proyecto de “Herramienta Computacional que apoye la etapa de búsqueda de estudios primarios en las bases de datos científicas”, se pueden identificar los siguientes tópicos que permitirán comprender el problema, y las temáticas asociadas a él, de tal forma de entregar la información necesaria para dar inicio al desarrollo del proyecto.

2.1. Revisiones sistemáticas

Una revisión sistemática se define como una manera de evaluar e interpretar toda la investigación disponible relevante respecto de una interrogante de investigación particular, en un área temática o fenómeno de interés [4]. Esta actividad es considerada como el primer paso a llevar a cabo en la obtención de conocimiento, entendimiento y delimitación de un tema de investigación científico.

En particular, este método propone tres etapas fundamentales, cada etapa con sus respectivas actividades que se deben realizar para llevar a cabo una revisión sistemática, como se muestran en la Figura 1.

Estas etapas son las siguientes:

Etapa 1: Planificación de la revisión

Esta etapa tiene como propósito específico definir los parámetros más importantes que serán tomados en cuenta cuando se lleve a cabo la revisión. Se debe establecer las razones que justifican llevarla a cabo, la manera en que se hará la búsqueda de trabajos y la forma en que éstos serán revisados. Finalmente, se evaluará la planificación realizada.

Se han considerado para esta etapa las siguientes actividades.

➤ *Identificación de la necesidad de la revisión.*

La necesidad de una revisión sistemática surge del requerimiento de un investigador de resumir toda la información existente sobre algún fenómeno interés de manera rigurosa e imparcial, con el objeto de iniciar otras actividades de investigaciones futuras. Antes de emprender una revisión sistemática, el investigador debe asegurarse de que ésta es necesaria. En particular, es recomendable identificar y analizar cualquier revisión sistemática existente acerca del fenómeno de interés con un criterio de evaluación apropiado [4].

➤ *Definición de un protocolo de revisión.*

El protocolo de revisión especifica los procedimientos que serán usados para emprender la revisión sistemática. La idea es impedir, en la medida de lo posible, que la selección de los estudios individuales pueda estar guiada por las expectativas del investigador.

Durante esta etapa se realiza la revisión de cada uno de los estudios encontrados y que potencialmente pueden ser incluidos en la revisión [4].

Etapa 2: Desarrollo de la revisión

En esta etapa se lleva a cabo la revisión propiamente tal. Su desarrollo está guiado por la planificación de la revisión. Sin embargo, y ya que es un proceso flexible, es posible incluir cambios a la planificación que mejoren su desempeño. A continuación se definen las sub-etapas que contempla el desarrollo de la revisión.

➤ Identificación de la Investigación

La principal actividad en esta etapa implica la especificación de un string de búsqueda y su aplicación a las distintas bases de datos.

La estrategia de búsqueda es una compensación entre encontrar todos los estudios primarios que sean relevantes para la investigación y no conseguir un abrumador número de falsos positivos, que luego deben ser excluidos manualmente. Un falso positivo es un resultado que es erróneamente positivo cuando no debería serlo; en este caso, significa que se encuentra un artículo que es de interés para la investigación, y más tarde resulta que no es así y por lo tanto tiene que ser eliminado. La cadena de búsqueda se desarrolla a partir del área a cubrir y la pregunta de investigación. El uso de múltiples bases de datos es una necesidad para poder cubrir toda la literatura relevante, pero también crea duplicados, los cuales deben ser identificados y eliminados [4], [5].

➤ Selección de estudios primarios

La selección de los estudios debe hacerse en base al protocolo de revisión definido. Este proceso será guiado por los criterios de inclusión y exclusión. Dependiendo de los intereses del proyecto, es recomendable registrar los motivos de exclusión.

El conjunto de estudios es procesado en relación con los criterios de selección. Para algunos estudios, es suficiente leer el título o el resumen de los artículos, mientras que otros

artículos necesitan un análisis más a fondo, por ejemplo, la metodología o conclusiones para determinar su estado.

➤ *Evaluación de la calidad*

Es importante evaluar la calidad de los estudios primarios, especialmente cuando los estudios reportan resultados contradictorios.

➤ *Extracción y gestión de datos*

Una vez que se decide el conjunto de los estudios primarios, en esta sub-etapa se extrae los datos de interés en los estudios, ya sean resúmenes, ideas o partes de los documentos. Esta extracción se debe realizar en base al protocolo de revisión definido. Además, se debe registrar la información necesaria que los investigadores consideren pertinente. Un formulario de extracción de datos está diseñado para recoger la información necesaria a partir de los informes de los estudios primarios [4].

➤ *Síntesis de datos*

En esta sub-etapa, al igual que en las anteriores, se debe aplicar el protocolo definido en la revisión, y consiste en registrar la información extraída de los estudios primarios siguiendo alguna estrategia.

Etapa 3: Publicación de los resultados

Esta etapa corresponde a la utilización de los resultados una vez que disponemos de ellos. Es muy importante la difusión de los resultados obtenidos producto de una revisión sistemática.

En este caso, junto con la obtención del documento exigido en cada currículo en particular, sería conveniente comunicar los resultados a través de la participación en conferencias, publicación de un artículo o de un informe técnico. Los informes detallados de los procedimientos de estudio es crítica para la capacidad de analizar y evaluar la calidad de las revisiones sistemáticas.

2.2. Mapeos sistemáticos

El mapeo sistemático es un método definido para construir clasificaciones y conducir análisis temático a los efectos de tener un mapa visual del conocimiento existente dentro de un tema amplio. El análisis de los resultados se realiza categorizando los descubrimientos y contando la frecuencia de bibliografía dentro de cada categoría para determinar la cobertura de las distintas áreas de un tema de investigación en específico [4].

Un estudio de mapeo sigue el mismo proceso de las revisiones sistemáticas, pero tienen diferentes criterios de inclusión/exclusión y calidad de los datos, debido a que el alcance es más amplio, los datos recogidos y la síntesis tienden a ser más cualitativos con respecto a las revisiones sistemáticas.

Las revisiones sistemáticas y el mapeo sistemático son complementarios y tienen como objetivo identificar los vacíos de la investigación, por lo que un mapeo sistemático es considerado por muchos como un paso imprescindible para decidir en qué áreas concretas del campo es interesante abordar una revisión sistemática más detallada.

A continuación se mostrará un cuadro comparativo, proporcionando un resumen de las características de los mapeos sistemáticos en comparación de las revisiones sistemáticas.

Tabla 1: Cuadro Comparativo de las Revisiones Sistemáticas y Mapeos Sistemáticos.

Revisión Sistemática	Mapeo Sistemático
Objetivos	
Identificación de las mejores prácticas en relación con los procedimientos, métodos o tecnologías mediante la unión de información de los estudios comparativos.	Clasificación y análisis temático de la literatura sobre un tema de interés.
Pregunta de Investigación	
Específico – relacionado con los resultados de los estudios empíricos. De la forma: Es el método A mejor o no que B?.	Genérico – relacionadas con las tendencias de la investigación. De la forma: qué investigadores, qué actividad, qué tipo de estudios, etc.
Proceso de Búsqueda	
Definido por la pregunta de investigación que identifica las tecnologías que se están investigando	Definido por el área de interés.
Alcance	
Enfocado. Son incluidos solo documentos efectivos relacionados con una pregunta de investigación específica y se extrae de cada documento información detallada sobre los resultados individuales de investigación	Extenso. Se incluyen todos los documentos relacionados con un área de interés, pero solo se recopilan datos de clasificación acerca de éstos.

Requisito de la estrategia de búsqueda	
Extremadamente riguroso – todos los estudios relevantes deben ser encontrados. Por lo general, los investigadores que realizan las revisiones sistemáticas necesitan utilizar técnicas que simplemente buscar fuentes de datos, como referencias en estudios primarios identificados o acercarse a los investigadores en el campo para averiguar si están llevando a cabo nuevas investigaciones en el área	Muchas veces menos riguroso solo si las tendencias de investigación son de interés, por ejemplo, los autores pueden buscar solo un conjunto específico de publicaciones, limitarse a los artículos journal, o restringir a una o dos bibliotecas digitales.
Evaluación de la calidad	
Importante para asegurar que los resultados se basen en mejores pruebas de calidad.	No esencial. También complicado por la naturaleza inclusiva de la búsqueda que puede incluir estudios teóricos, así como estudios empíricos de todos los tipos que permitan evaluar los estudios primarios complicados
Resultados	
Los resultados de los estudios primarios se agregan para responder la(s) pregunta(s) específica(s) de investigación, posiblemente con calificadores.	Un conjunto de documentos relacionados con un área temática categorizado en una variedad de dimensiones y recuento del número de documentos en varias categorías.

2.3. Formato BibTeX

BibTeX es un programa auxiliar de LaTeX, que fue escrito por Oren Patashnik diseñado para facilitar el manejo de la bibliografía. Permite generar listas de referencias bibliográficas formateadas de manera consistente y cambiar fácilmente determinadas características de las mismas [6][7].

En los ficheros .bib se usa el carácter “@”, seguido de una palabra representativa del tipo de registro que se trate, para indicar que empieza un registro. Asimismo, se usan llaves para delimitar el contenido del registro, así como se muestra en la Figura 4.

```
@tipoentrada{label ,
campo1 = "Valor campo 1",
campo2 = "Valor Campo 2",
campo3 = "Valor Campo 3",
[...]
campon = "Valor Campo n",
}
```

Figura 4: Formato general BibTeX.

Donde “*tipoentrada*” es el nombre que identifica a un concreto tipo de registro. En el contenido de éste, hay dos elementos:

- La clave de identificación (label): es siempre su primer elemento. Se usa para distinguir a un registro concreto de los otros registros. La clave puede consistir en cualquier combinación de letras, números y ciertos signos de puntuación. No se admiten vocales acentuadas y los caracteres “¡¿ñÑçÇ” ni ciertos signos de puntuación como la coma.

- Los campos: BibTeX no le interesa la estructura de un determinado registro. Para escribir un campo dentro de éste se puede usar independientemente cualquiera de las dos formas siguientes:

NombreCampo = { Contenido },

NombreCampo = “ Contenido “,

- *Nombre del campo:* Normalmente será uno de los campos, obligatorios u opcionales, previstos para el tipo de registro de que se trate.
- *Delimitadores de contenido:* Como delimitadores se pueden usar las llaves o las comillas dobles. Cuando se ocupa comillas dobles como delimitadores dentro de la cadena delimitada no se puede usar comillas dobles.
- *La coma final:* Indica que el contenido del campo ha terminado y puede empezar otro distinto.

Para un determinado estilo bibliográfico, los campos presentes en un registro se incluirán en uno de los siguientes grupos:

- *Campos obligatorios:* Son aquellos que el estilo en cuestión espera encontrar en un determinado tipo de registro, de manera que si en un registro concreto alguno de ellos no está presente, se generará una advertencia en la salida estándar y en el fichero.
- *Campos opcionales:* Son aquellos para los que el estilo en cuestión ha previsto algún tipo de acción, pero se ha previsto también su posible ausencia, y ésta no tiene especial trascendencia.

2.3.1. Tipos de registros bibliográficos.

A continuación se describirán los nombres de los distintos tipos de registros previstos en bibtex que ocupará la herramienta.

- *Article*: Campos Obligatorios: *author*, *title*, *journal* y *year*.

Campos Opcionales: *volumen*, *number*, *pages* y *month*.

```
@ article { ahu61 ,
  author = { Arrow, Kenneth J. and Leonid Hurwicz and Hirofumi Uzawa },
  title = { Constraint qualifications in maximization problems },
  journal = { Naval Research Logistics Quarterly },
  volume = { 8 },
  year = 1961 ,
  pages = { 175-191 }
}
```

Figura 5: Ejemplo tipo de registro "Article".

- *Book*: Campos obligatorios: *author o editor*, *title*, *Publisher* y *year*

Campos opcionales: *volumen o number*, *series*, *address*, *edition* y *month*.

```

@book{ab94,
  author * = {Charalambos D. Aliprantis and Kim C. Border},
  year = {1994},
  title = {Infinite Dimensional Analysis},
  publisher = {Springer},
  address = {Berlin}
}

```

Figura 6: Ejemplo tipo de registro "Book".

➤ *InCollection*: Campos Obligatorios: *author*, *title*, *booktitle*, *Publisher* y *year*.

Campos Opcionales: *crossref*, *editor*, *volumen o number*, *series*, *type*,
chapter, *pages*, *address*, *edition* y *month*.

```

@incollection{m85,
  author = {Maskin, Eric S.},
  year = {1985},
  title = {The theory of implementation in {N}ash equilibrium: a survey},
  booktitle = {Social Goals and Social Organization},
  editor={Leonid Hurwicz and David Schmeidler and Hugo Sonnenschein},
  pages={173-204},
  publisher = {Cambridge University Press},
  address={Cambridge}
}

```

Figura 7: Ejemplo tipo de registro "InCollection".

➤ *InProceedings*: Campos Obligatorios: *author*, *title*, *booktitle* y *year*

Campos Opcionales: *crossref*, *editor*, *volumen o number*, *series*,
pages, *address*, *month*, *organization* y *publisher*.

```
@inproceedings{ah2006,  
  author={Aggarwal, Gagan and Hartline, Jason D.},  
  year={2006},  
  title={Knapsack auctions},  
  booktitle={Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms},  
  pages={1083-1092},  
  publisher={Association for Computing Machinery},  
  address={New York}  
}
```

Figura 8: Ejemplo tipo de registro "InProceedings".

2.4. *Web Scraping*

El web scraping es una técnica que permite leer el contenido de un sitio web mediante un software empleando un robot con la finalidad de extraer datos de manera automática. La extracción de datos Web se lleva a cabo habitualmente en documentos estructurados, los cuales comúnmente usan lenguaje de marcado (HTML, XML, etc.) para mostrar su estructura interna [8][9].

2.4.1. *Proceso de web Scraping de la herramienta.*

El proceso de extracción de datos orientados a una página web puede ser dividido en 4 pasos [8] como se muestra en la Figura 9.

Paso 1: El programa extractor realiza un requerimiento interactuando con el sitio web mediante una dirección de URL.

Paso 2: El programa extractor obtiene una respuesta, necesita la estructura de la sección HTML como parámetro de entrada, dicha sección es conocida como contenedor e incluye detalladamente la estructura de etiquetas que contiene la información a ser extraída.

Paso 3: A partir de la respuesta, la herramienta podrá parsear y extraer los datos; esto consiste en analizar la información obtenida leyéndola y almacenándola siguiendo una determinada estructura para una posterior manipulación de la misma.

Paso 4: Finalmente los datos extraídos son enviados a aplicaciones externas, en este caso a un archivo con formato BibTeX.

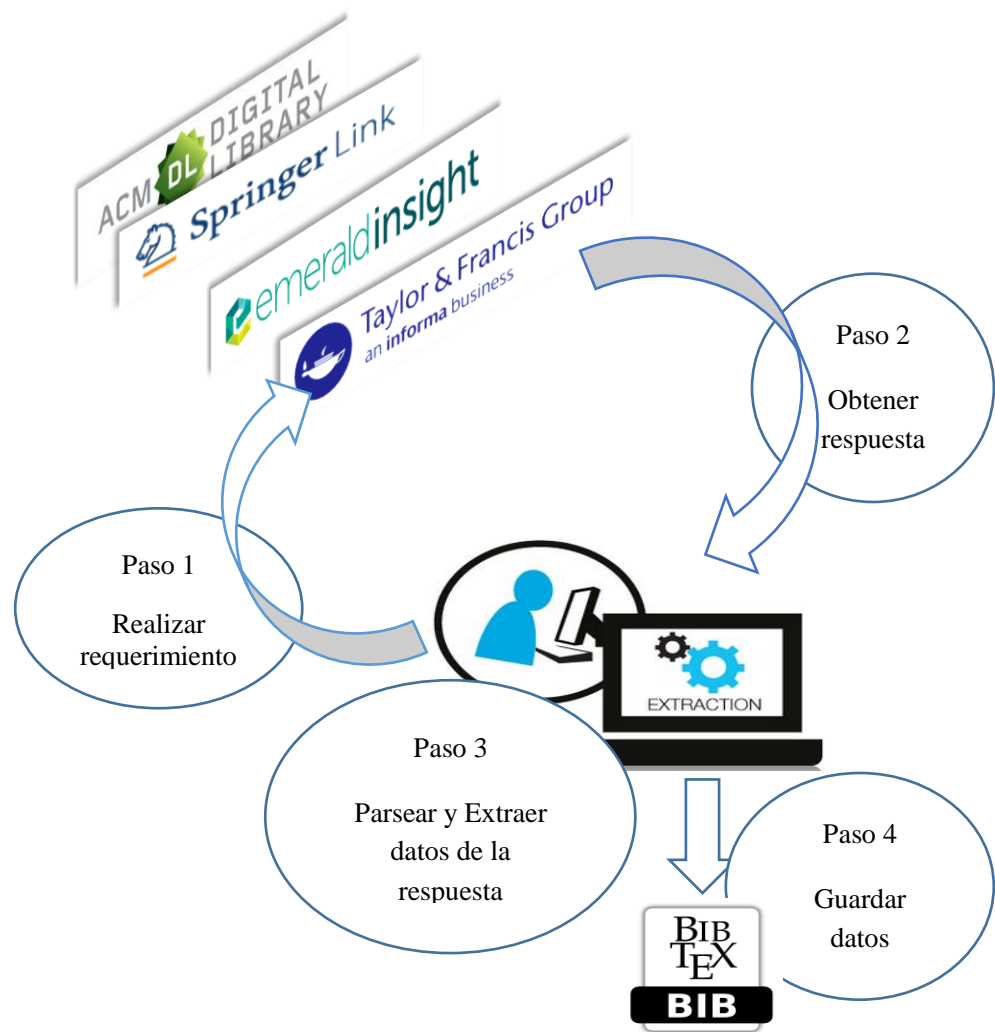


Figura 9: Ejemplo del proceso de "web scraping".

2.4.2. Conceptos para entender web scraping

Para empezar, se verá de qué forma se envía la información desde el servidor al ordenador y cómo se interceptará antes de procesarla.

El web scraping se basa en dos acciones o tareas:

- Realizar una petición HTTP.
- Procesar la respuesta HTML para extraer los datos.

Se analizará estos dos conceptos fundamentales

2.4.2.1. Protocolo HTTP

HTTP es un protocolo de transferencia de hipertexto que se usa en la web, es el acrónimo de Hypertext Transfer Protocol, o Protocolo de transferencia de hipertexto. Este protocolo fue desarrollado por las instituciones internacionales W3C y IETFF y se usa en todo tipo de transacciones a través de internet.

El HTTP facilita la definición de la sintaxis y semántica que utilizan los distintos softwares web tanto clientes como servidores y proxis para interactuar entre sí.

Este protocolo opera por petición y respuesta entre el cliente y el servidor. A menudo las peticiones tienen que ver con archivos, ejecución de un programa, consulta a una base de datos y otras funcionalidades [10].

Toda la información que opera en la web mediante este protocolo es identificado mediante el URL o dirección.

Es un protocolo sin estado, es decir, que no guarda ninguna información sobre conexiones anteriores. El desarrollo de aplicaciones web necesita frecuentemente mantener estado. Para esto se usan las cookies, que es información que un servidor puede almacenar en el sistema cliente. Esto le permite a las aplicaciones web instituir la noción de "sesión", y también permite rastrear usuarios ya que las cookies pueden guardarse en el cliente por tiempo indeterminado [11].

HTTP ha pasado por múltiples versiones del protocolo, muchas de las cuales son compatibles con las anteriores. El cliente le dice al servidor al principio de la petición la versión que usa, y el servidor usa la misma o una anterior en su respuesta. La versión que se usa actualmente es la 2.0.

❖ *Cabeceras del protocolo HTTP*

HTTP es un protocolo cliente-servidor, lo que significa que el cliente envía una petición al servidor y espera un mensaje de respuesta del servidor. Casi todo lo que se ve en el navegador es transmitido a la computadora sobre HTTP. Las cabeceras HTTP son la parte principal de las solicitudes y respuesta HTTP, y ellos llevan la información sobre el navegador del cliente, la página requerida, el servidor y más, como se muestra en la Figura 10.

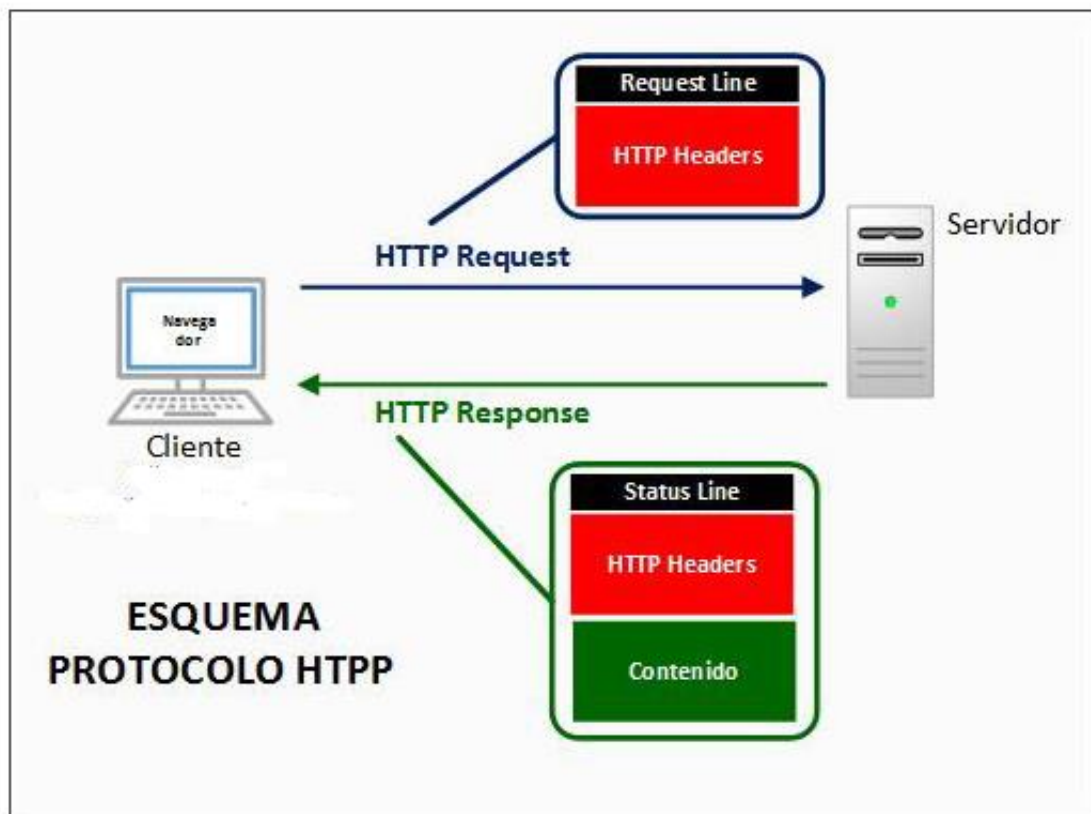


Figura 10: Esquema de protocolo HTTP

Ejemplo:

Cuando se escribe una URL en la barra de direcciones, el navegador envía una solicitud HTTP y éste puede lucir así: como se muestra en la Figura 11.

```
GET /tutorials/other/top-20-mysql-best-practices/ HTTP/1.1
Host: net.tutsplus.com
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.1.5) Gecko/20091102 Firefox/3.5.5 (.NET CLR 3.5.30729)
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
Cookie: PHPSESSID=r2t5uvjq435r4q7ib3vtdjq120
Pragma: no-cache
Cache-Control: no-cache
```

Figura 11: Ejemplo de cabecera de petición.

La primera línea es la solicitud HTTP, la cual tiene la información básica. El resto es la cabecera HTTP.

❖ Estructura de una petición HTTP

La primera línea de solicitud HTTP consiste en 3 partes, como se muestra en la Figura 12.

method	path	protocol
GET	/tutorials/other/top-20-mysql-best-practices/	HTTP/1.1

```
Host: net.tutsplus.com
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
Cookie: PHPSESSID=r2t5uvjq435r4q7ib3vtdjq120
Pragma: no-cache
Cache-Control: no-cache
```

HTTP headers as Name: Value

Figura 12: Estructura de solicitud HTTP.

- El “method” indica que tipo de solicitud es. Los métodos más comunes son: GET y POST.
- El path es generalmente la parte del URL que viene después del host o dominio. Por ejemplo cuando se solicita <http://net.tutsplus.com/tutorials/other/top-20-mysql-best-practices/> la parte del path es: "/tutorials/other/top-20-mysql-best-practices/".
- El “protocol” contiene HTTP y la versión, la cual es usualmente 2.0 en los navegadores.

A continuación, se define una serie de campos o parámetros que van en el encabezado de cada petición que podrán o no ser rellenados en función del tipo de petición que se realice al servidor. Estos campos son los que se muestran en la Tabla 2.

Tabla 2: Campos de cabecera del protocolo HTTP.

Nombre de campo	Descripción
Accept	Indica el formato de datos que se espera o que se admite en el cuerpo de la respuesta.
Accept-Encoding	Indica el tipo de compresión de datos que se espera en la respuesta. No todos los navegadores envían este campo.
Accept-Language	Indica el código del lenguaje esperado en la respuesta.
Content-Encoding	Describe la codificación de contenido que se utiliza en el cuerpo de entidad.
Content-Language	Indica el idioma del mensaje de respuesta que se devuelve al destinatario.
Content-Type	Indica el tipo de soporte del cuerpo entidad enviado al destinatario.

2.4.2.2. Respuesta HTML

Las páginas webs son documentos escritos en HTML. Los documentos webs están representados por un árbol estructurado llamado DOM (Document Object Model) y el objetivo de HTML es definir el modo en que el navegador web mostrará el contenido [12], como se muestra en la Figura 13.

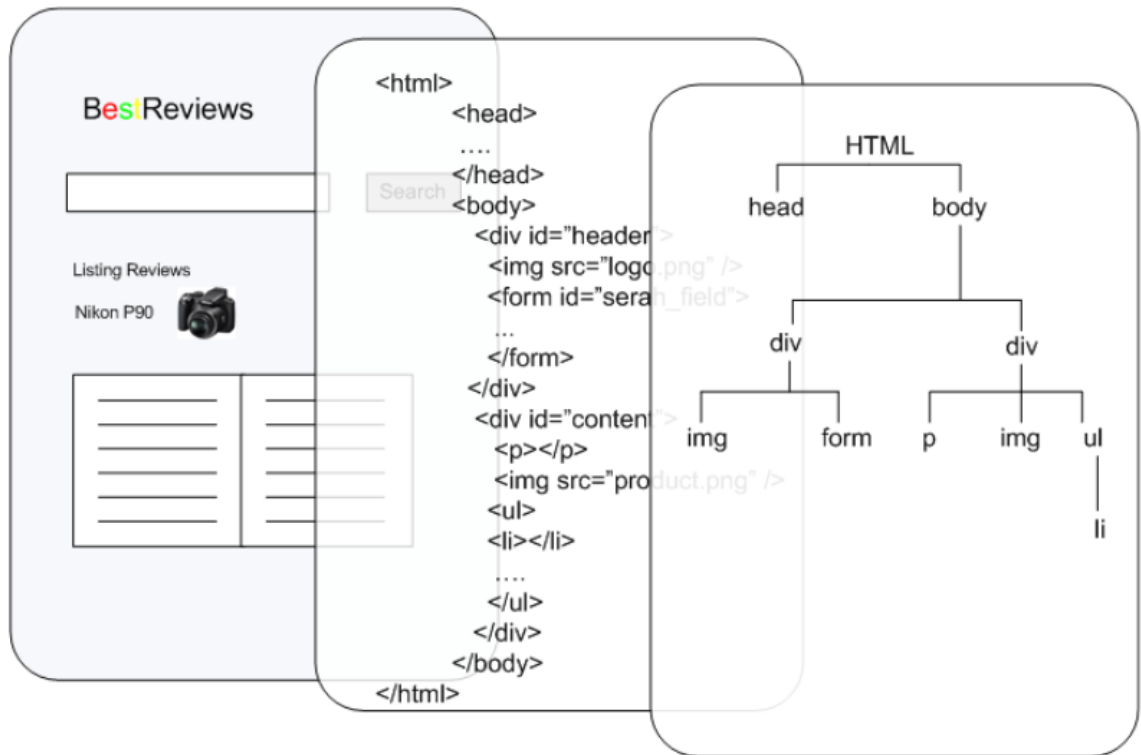


Figura 13: Representación Web, HTML, DOM.

La información extraída se transforma en datos estructurados que pueden ser almacenados y analizados para su uso posterior, teniendo como principal beneficio el hecho de recolectar eficientemente datos de diferentes orígenes con un mínimo de esfuerzo humano [12].

2.4.3. Problemas con web Scraping.

Existen diversas dificultades de la técnica del web scraping, que impiden obtener los datos, estos factores pueden hacer que realizar la obtención sea más complicada. Algunos de estos factores son [8]:

- Código de HTML mal estructurados: En este tipo de páginas, se puede obtener información pero sin clasificar, de modo que su obtención implica un posterior procesamiento haciendo la práctica más complicada o incluso impidiendo obtener los datos necesarios en el caso de grandes volúmenes de datos.
- Bloqueo al acceso masivo: Los accesos a una web que no se corresponden con acciones humanas, como son el número de páginas solicitadas por minuto, pueden provocar el bloqueo de la IP.
- Sistemas que usan cookies: Sistemas basados en sesiones que usan cookies de navegador para rastrear lo que hace el usuario.

2.5. *Herramientas a utilizar*

2.5.1. *Lenguaje de programación PHP*

PHP es un acrónimo recursivo que significa *PHP Hypertext Pre-processor* (inicialmente PHP Tools, o *Personal Home Page Tools*). Es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Es usado principalmente en interpretación del lado del servidor pero actualmente puede ser utilizado desde una interfaz de líneas de comandos o en la creación de otros tipos de programas [13].

Es un lenguaje multiplataforma con capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad. Posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.

Mediante la utilización de PHP se puede fácilmente recopilar datos de formularios, interaccionar con bases de datos y páginas web, generar páginas con contenidos dinámicos, enviar y recibir cookies, entre otras cosas más [14].

Para el desarrollo de esta herramienta computacional, se utilizará PHP debido a que este lenguaje contiene la librería PHP CURL que permitirá crear conexiones con distintos protocolos para poder conectar la aplicación que se realizó con otros entornos [13].

CURL es una biblioteca de funciones para conectar con servidores y trabajar con ellos. Permite transferir información con sintaxis de URL, es decir, sirve para realizar acciones sobre archivos que existen en URLs de Internet, soportando los protocolos más comunes, como http, ftp, https, etc.

A continuación se mostrará un ejemplo básico de un Curl en PHP [15].

```
1) $ch = curl_init("http://www.php.net/");  
2) $response = curl_exec($ch);  
3) curl_close($ch);  
4) echo $response;
```

La primera línea de código es para inicializar una sesión Curl usando `curl_init()`, en la segunda línea de código es para ejecutar la sesión con la función `curl_exec()`, la tercera línea de código es para finalizar la sesión con `curl_close()` y la última línea de código es para mostrar el contenido de la página a la cual se está conectando que en este caso es la página de PHP.

Al ejecutar este pequeño trozo de código, lo que está haciendo es abrir una conexión con la página de PHP, descarga el contenido como si de un navegador se tratase, e imprimirlo como si fuese contenido propio. Normalmente suele complicarse, porque hay páginas webs que devuelven una respuesta u otra según el navegador que se encuentra, o en un idioma u otro dependiendo del valor de las cabeceras HTTP.

2.5.2. Simple HTML DOM Parser

Con el documento HTML que se obtiene gracias a PHP Curl se podrá avanzar hacia el parseo de dicho documento. Para ellos Simple HTML DOM Parser dispone de una serie de clases y métodos que permiten simplificar al máximo esta tarea obteniendo un código más fácil de leer.

A partir de aquí ya se tendría un árbol DOM en el que el elemento raíz sería la etiqueta `<html>` y los nodos descendientes serían la etiqueta `<head>` y la etiqueta `<body>` donde se encontrarían la información buscada. Los nodos descendientes de `<head>` y `<body>` serían

las etiquetas que se encontrasen dentro de cada uno de ellos respectivamente como se muestra en la Figura 14.

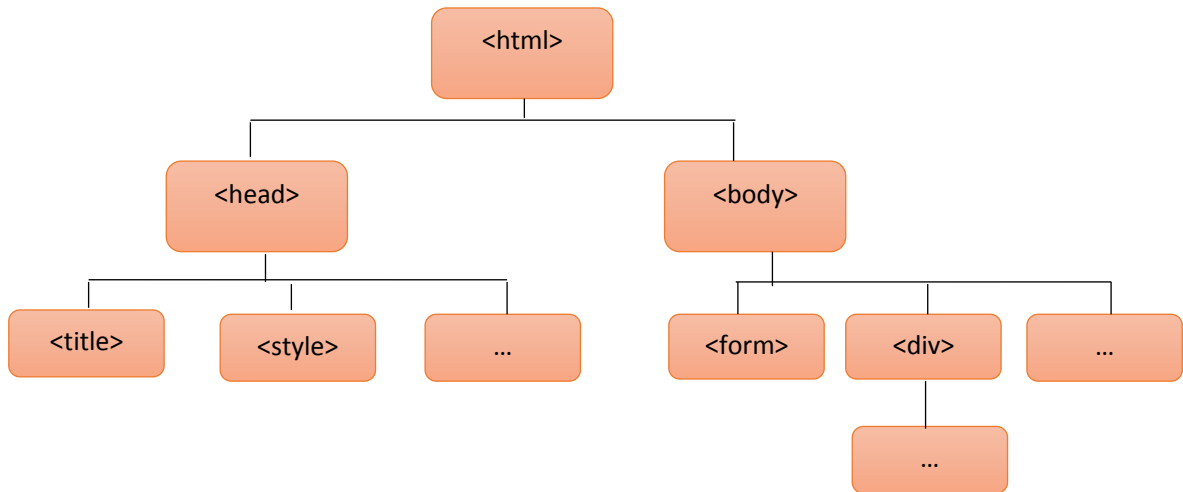


Figura 14: Árbol DOM creado a partir de un documento HTML.

Una vez obtenido el árbol DOM (información estructurada), el siguiente paso consiste en extraer solo la información que interesa; esta información se encontrará dentro de uno o varios nodos descendientes de <body>, por tanto, la tarea es encontrar esos nodos y obtener la información que contienen. El nombre de los nodos que se tienen que buscar tendrá que obtenerse del análisis de las páginas web que se presentarán en el Anexo A.

La documentación de la herramienta Simple HTML DOM Parser se encuentra en: <http://simplehtmldom.sourceforge.net/>.

2.5.3. *HTML*

La sigla HTML viene de un acrónimo que significa “Hypertext Markup Language” que, en resumen, es un lenguaje de maquetación orientado al desarrollo de sitios webs.

Si se habla de HTML puro, su principal función radicaría en cómo se distribuirá la información que se va a dar a conocer en el sitio web; cabe mencionar que para mostrar el contenido, los navegadores web interpretan las etiquetas de código HTML y muestran el contenido dependiendo de la etiqueta en la cuales están insertos [16].

➤ *Sintaxis de HTML*

El HTML es un lenguaje que basa su sintaxis en un elemento de base al que llamamos etiqueta. La etiqueta presenta dos partes:

- ✓ Una apertura de forma general <etiqueta>.
- ✓ Un cierre de tipo </etiqueta>.

Todo lo incluido en el interior de esa etiqueta experimentará las modificaciones que caracterizan a esa etiqueta. Así por ejemplo:

Las etiquetas y definen un texto en negrita. Si en el documento de HTML se escribe una frase con el siguiente código.

```
<b> Paola Bot </b>
```

El resultado será: **Paola Bot**

➤ *Partes de un documento HTML*

Un documento HTML ha de estar delimitado por la etiqueta <html> y </html>. Dentro de este documento, se pueden distinguir dos partes principales:

- ✓ El encabezado, delimitado por <head> y </head> donde se colocarán etiquetas de índole informativo como por el ejemplo el título de la página.

- ✓ El cuerpo, delimitado por las etiquetas `<body>` y `</body>`, que será donde se pondrá el texto e imágenes delimitados a su vez por otras etiquetas como la del ejemplo anterior.

El resultado es un documento con la siguiente estructura:

```
<html>
  <head>
    Etiquetas y contenidos del encabezado
    Datos que no aparecen en nuestra página pero que son importantes para
    catalogarla: Título,
    Palabras clave,...
  </head>
  <body>
    Etiquetas y contenidos del cuerpo
    Parte del documento que será mostrada por el navegador: Texto e imágenes
  </body>
</html>
```

2.5.4. CSS

CSS es el acrónimo de “Cascade Style Sheet” que se utiliza para dar estilo a alguna página o sistema web. Este lenguaje de estilos describe cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir definiendo la presentación del entorno web en un plano totalmente estético. Con CSS se puede controlar el color del texto, el estilo de las fuentes, el espacio entre párrafos, cómo se dimensionan y se exponen las columnas, así como una variedad de otros efectos [17]. Una de las características principales de este lenguaje es su capacidad de mantenibilidad, un documento CSS que complementa a una página web puede ser fácilmente modificado si se desea cambiar algún aspecto de dicha página: solo modificando algunas líneas de código, se puede cambiar el color de fondo, el aspecto de la página, etc. [18]. Para el caso del proyecto, el lenguaje CSS será utilizado para la estética de la interfaz de usuario de la herramienta computacional.

Capítulo 3

Estado del Arte

3.1. Base de Datos Epistemonikos

Epistemonikos fue creada en 2009 por Gabriel Rada (Director del Programa de Salud Basada en la Evidencia [PROSABE- Programa de Salud Basada en Evidencia], Pontificia Universidad Católica de Chile) y Daniel Pérez (propietario de Etiqs, empresa de desarrollo de software en Santiago, Chile), quienes trabajaron junto a su equipo para lograr el lanzamiento oficial fue el año 2012. Se trata de una base de datos que pretende recopilar toda la evidencia existente en el área de la salud. Posee una interfaz de búsqueda simple e intuitiva, esto se complementa con una estrategia de búsqueda amplia, y una selección y categorización rigurosa de la información[19].

El objetivo de Epistemonikos es identificar y poner en un mismo sitio lo que se llama evidencia en salud, y dejar fuera el resto de la información. Mediante búsquedas automatizadas, esta base de datos identifica todas las revisiones sistemáticas, revisiones panorámicas, guías de práctica clínica y resúmenes estructurados existentes. Además incorpora todos los estudios primarios incluidos en revisiones sistemáticas que son ingresadas en la base de datos [20]. Por ejemplo si se encuentra un estudio primario, es posible acceder a todas las revisiones sistemáticas que han incluido a éste. De esta forma, Epistemonikos favorece el movimiento desde evidencias de menor a mayor jerarquía.

Epistemonikos proporciona búsquedas avanzadas, donde el investigador consigue realizar búsquedas por campos, siendo estos: *Title*, *Abstract*, *Title/Abstract* y *Authors*, como se muestra en la Figura 15.



Figura 15: Búsqueda Avanzada Epistemonikos

Luego de realizar la búsqueda la herramienta le da la posibilidad al usuario refinar dicha búsqueda, por algunos filtros como: año de la publicación, tipo de publicación etc., como se muestran en la Figura 16.

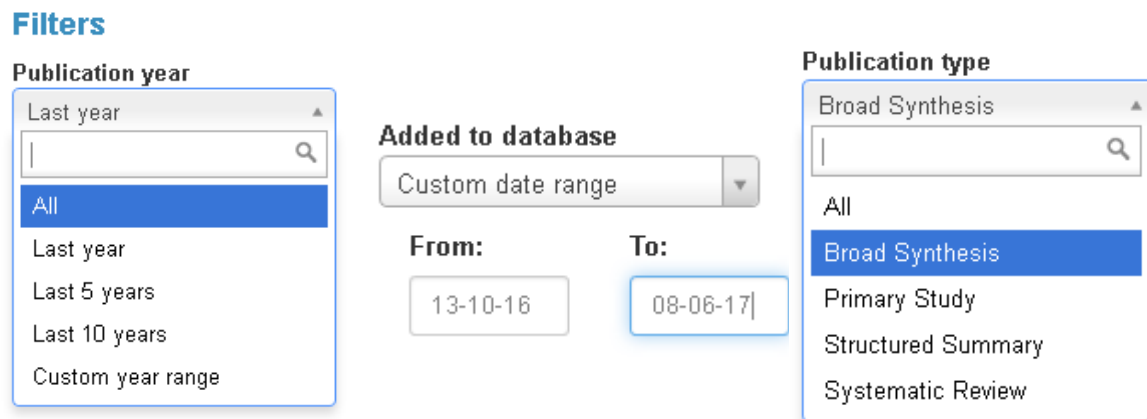


Figura 16: Tipos de Filtros de Epistemonikos.

En la Figura 17 se puede apreciar cómo se realiza la extracción. Como pueden observar, no existe consistencia entre la cantidad de registros arrojados con la cantidad de metadatos que se pueden extraer, se realizó una búsqueda en la base de datos arrojando 20815 registros, luego se quiso exportar todos los resultados dejando extraer solo 100, esto implica que el investigador tendrá que exportar todos los resultados manualmente.



Figura 17: Exportar Resultados Epistemonikos.

3.2. *Mendeley*

Mendeley es un gestor bibliográfico que combina una versión web con una versión de escritorio. Es un sistema para almacenar y organizar documentos que permite compartir las referencias bibliográficas [21].

➤ Características de Mendeley

1. Permite extraer automáticamente los metadatos y las referencias de los artículos desde archivos PDF. Recupera información adicional desde algunos sitios web (Pubmed, ScienceDirect, GoogleScholar, entre otros).
2. Guarda y organiza datos de referencia bibliográfica en la red y en el computador.
3. Sincroniza los datos de las referencias bibliográficas para generar citas y bibliografías mientras se escribe un artículo científico.
4. Integración con editores de texto como Microsoft Word.
5. Captura referencias bibliográficas desde el navegador a MyLibrary usando la herramienta “Web Importer”.
6. Permite la colaboración para compartir y anotar en grupo artículos científicos en PDF.

En resumen, Mendeley es una herramienta para cualquier investigador que quiera localizar información, compartir datos, trabajar con referencias bibliográficas, visualizar y anotar documento PDF.

En esta sección se hablará solo de la característica “Web Importer”, ya que es una aplicación de Mendeley que se asimila a la herramienta computacional que se construirá.

Para utilizar Web Importer hay que instalar el plugin en el navegador, como se muestra en la Figura 18. Luego, se podrá ver un artículo en la red e importar los datos directamente a Mendeley Web, simplemente haciendo click en el ícono “Import to Mendeley”.

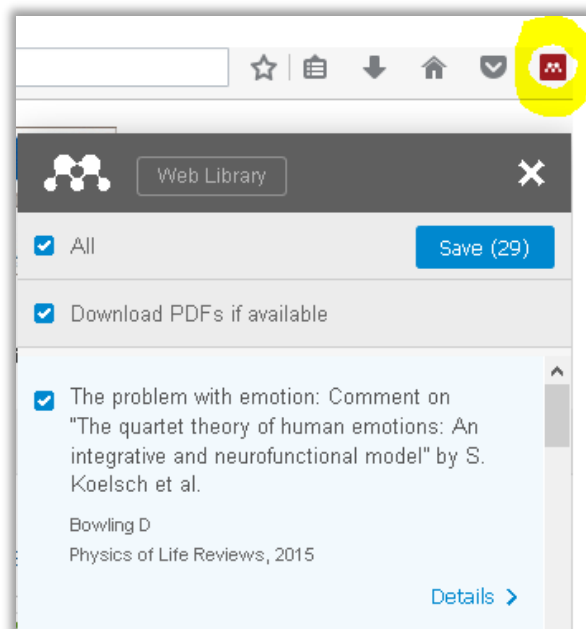


Figura 18: Mendeley incrustado en la barra de navegación.

Mendeley importa de la web los metadatos del autor, título, fechas y otros metadatos dependiendo del artículo. Algunas de las bases de datos con las que trabaja para importar metadatos bibliográficos son las siguientes:

1. Scopus
2. SpringerLink
3. Wiley Online Library
4. PubMed
5. ScienceDirect
6. IEEE Xplore

De estas bases de datos algunas permiten rescatar varias referencias de una sola vez, pero en la mayoría de los casos hay que ir visualizando referencia por referencia y extrayéndolas una a una.

A continuación se realizará una búsqueda en la bases de datos ScienceDirect, para luego extraer los metadatos de los artículos con la herramienta “Web Importer”. Como se puede observar en la Figura 19, se ingresó el siguiente string de búsqueda TITLE((emotion or emotions or mood or moods or affect or affects) and (interaction or share or "self-report")) arrojando 152 resultados.

Search results: 152 results found for pub-date > 2006 and pub-date < 2011 and TITLE((emotion or emotions or mood or moods or affect or affects) and (interaction or share or "self-report")).

Purchase | Export | Relevance ▾

- Étude exploratoire des relations entre conscience émotionnelle, partage social des émotions, états anxieux dépressifs Original Research Article
L'Encéphale, Volume 36, Supplement 2, June 2010, Pages D97-D104
A. Pasquier, J.-L. Pedinielli
 Abstract | Purchase PDF - \$35.95
- Mirrors in science: How mirror neurons changed cognitive neuroscience
Cortex, Volume 46, Issue 1, January 2010, Pages 139-143
Marcel Brass, Shirley-Ann Rüschemeyer
 Purchase PDF - \$35.95

Figura 19: Búsqueda en Science direct.

Luego se procedió a ocupar la herramienta “Web Importer” y en la Figura 20, se observa que solo se pueden guardar 99 artículos de los 152 que arrojó la base de datos Science Direct. Por los tanto, se podrán marcar 99 referencias y exportarlas a un archivo bibliográfico (RIS, BibTeX, XML).

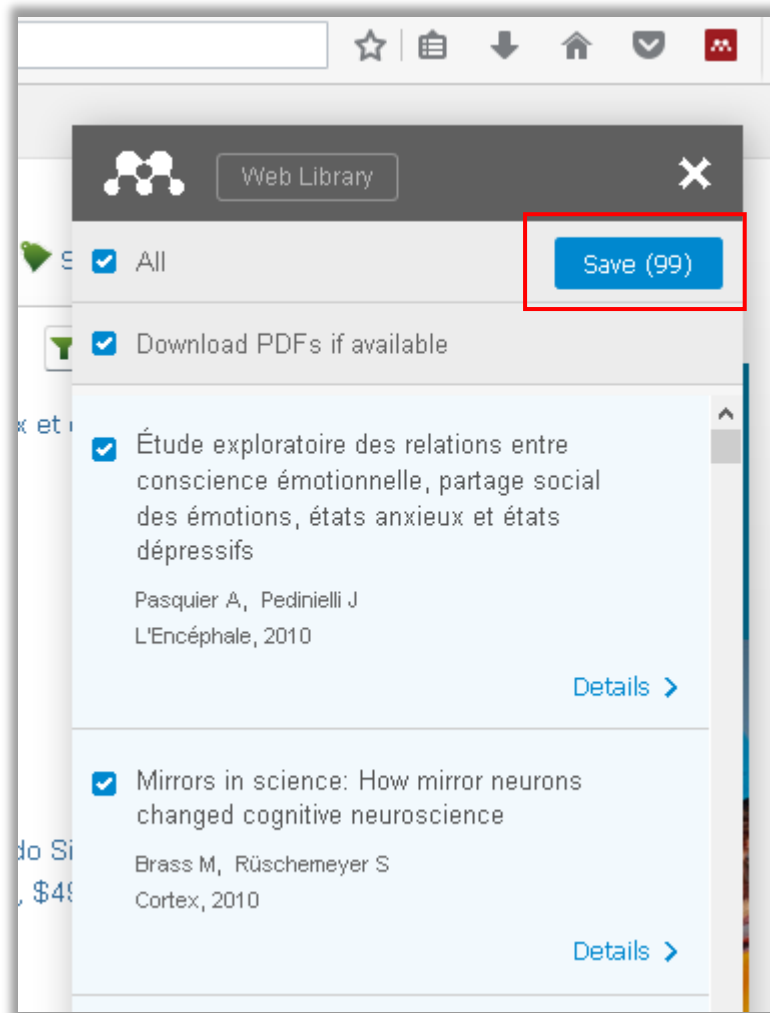


Figura 20: Uso de la herramienta "Web Importer".

3.3. *JabRef*

JabRef es un gestor de referencias bibliográficas de código abierto que permite importar, editar, organizar y exportar referencias en formato BibTeX, siendo útil para usuarios que ocupan LaTeX [22].

➤ *Búsqueda Web*

JabRef permite añadir referencias a través de enlaces a servicios de búsqueda en línea como IEEEExplore, Springer, Medline proporcionando realizar búsquedas directamente sobre estas bases de datos. Primero se deberá seleccionar la opción “Búsqueda Web” y se desplegará un cuadro con las opciones de las bases de datos como se muestra en la Figura 21.

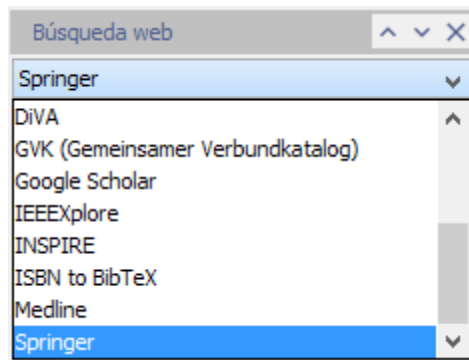


Figura 21: Opciones de bases de datos para realizar búsquedas en línea.

Para el ejemplo se escogieron las bases de datos IEEEExplore y Springer. Se realizó la búsqueda con el siguiente string: “systematic reviews”, donde fue ingresado en el motor de búsqueda de la herramienta JabRef,

Para IEEEExplore fueron encontrados 1703 entradas en donde solo se pudieron descargar 100 artículos de la búsqueda, como se ve en la Figura 22.

Ahora se verá el ejemplo para Springer con el mismo string de búsqueda y si observar en la Figura 23, se encontraron 390979 archivos en la búsqueda, en el que se pudieron recuperar todas las referencias para esta búsqueda, el problema que hubo es que al momento de descargarlas tomó demasiado tiempo en extraer estos metadatos.

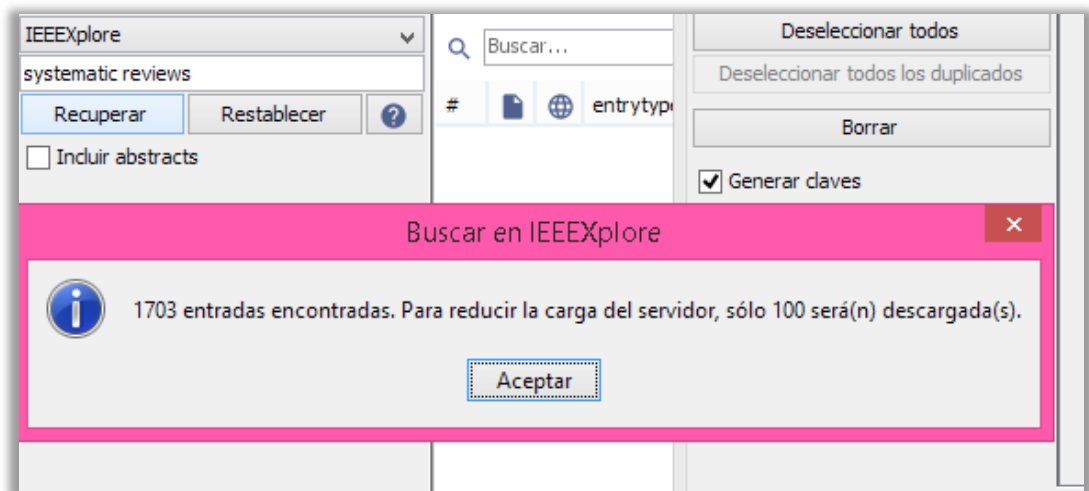


Figura 22: Búsqueda en la bases de datos IEEEExplore.

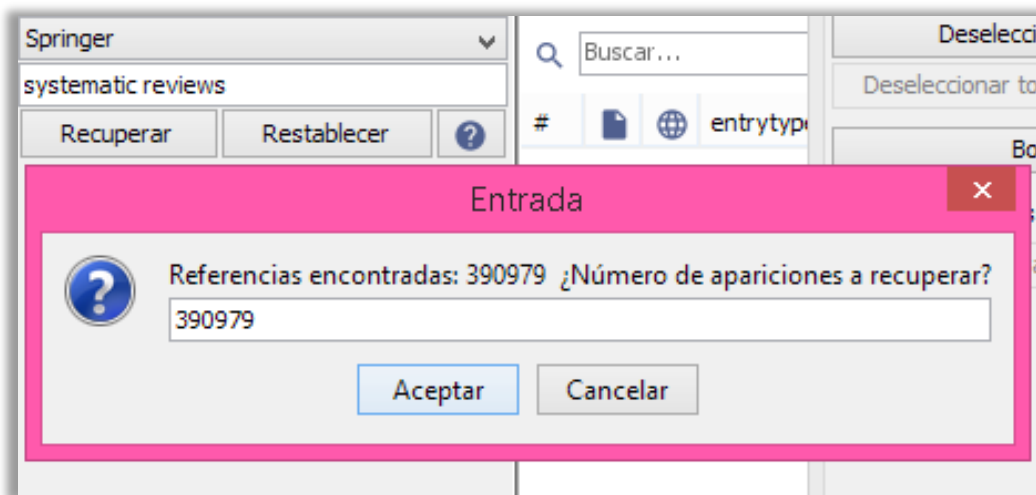


Figura 23: Búsqueda en la bases de datos Springer.

3.4. StArt

Como se ha mencionado anteriormente, realizar una revisión sistemática toma demasiado tiempo debido al rigor requerido de hacer las actividades de este proceso, es por esto que se desarrolló esta herramienta para soportar parte o la totalidad de las fases de las revisiones sistemáticas con el objetivo de minimizar el esfuerzo requerido por parte de los investigadores [5].

Primero se mostrará como rellenar los campos del protocolo, es importante llenar todos los campos para asegurar que este modelo sea replicable. Se empezará ingresando las palabras claves siendo de vital importante que sean relacionadas con la revisión, incluyendo sinónimos como se muestra en la Figura 24 (a).

Luego el investigador deberá elegir las fuentes de donde se quiere sacar la información, el usuario tendrá que seleccionar las opciones que se encuentran en una lista predeterminada como se muestra en la Figura 24 (b).

Después de esto se definirán los criterios de selección de estudio, en otras palabras, los criterios de inclusión y exclusión se deben definir aquí (ver Figura 24 (c)), al definir un criterio, es necesario clasificarlo como tipo de inclusión o exclusión, y a continuación añadirlo a la lista de criterios. Ahora se deberá definir los campos del formulario de extracción de información (Ver Figura 24 (d)) [23].

Keywords and Synonyms*

Keywords:

QDA
 analisis cualitativo
 codificacion

(a)

Source list*

Source:

ACM
 IEEE
 Scopus
 Springer

(b)

Study selection criteria (inclusion and exclusion)

Criterion:

(I) presenta analisis cualitativo
 (I) presenta pautas para conducir un analisis
 (E) Introduccion de analisis

(c)

Data Extraction Form Fields*

Text
 Pick One List
 Pick Many List

Name

grupo de busqueda={estudio de evaluacion,propuesta de estudio,relato de experiencia}
 cual tecnica fue utilizada?

(d)

Figura 24: Ejemplo de llenado del protocolo en la fase de planificación.

Después de completar los campos del protocolo, el usuario deberá ir a la fase de ejecución. En el proceso de identificación de la investigación, el usuario tiene que:

1. Crear una cadena de búsqueda para cada fuente elegida en el protocolo. Por ejemplo Scopus, IEEEExplore.
2. Guardar el resultado en un archivo BibTeX.
3. Copiar la cadena de búsqueda aplicada en el motor de búsqueda y luego se deberá pegar en el campo correspondiente StArt.

Para importar los resultados de la cadena de búsqueda en StArt, es necesario crear una sesión de búsqueda para cada base de datos. Para el ejemplo se escogió ACM Digital Library, y se ingresó la cadena de búsqueda para esa base de datos como se muestra en la Figura 25.

Insert search session informations

Insert the search session information

String

```
((acmdlTitle:(emotion) OR acmdlTitle:(emotions) OR acmdlTitle:(mood) OR acmdlTitle:(moods) OR acmdlTitle:(affect) OR acmdlTitle:(affects)) AND (acmdlTitle:(interaction) OR acmdlTitle:(share) OR acmdlTitle:(\"self-report\")))
```

Source

ACM

The field "String" should be copied from the search machine.

◀ Back Next ▶ Finish Cancel

Figura 25: Ejemplo ingreso string de búsqueda

El investigador tendrá que extraer manualmente las referencias bibliográficas de la base de datos elegida, para que luego pueda importarlas a la herramienta como se muestra en la Figura 26.

Durante la fase de selección de los estudios, es necesario seguir la estrategia definida en el protocolo. Comúnmente, los criterios de inclusión y exclusión se aplican teniendo en cuenta el título, el abstract y las palabras claves de cada estudio. En esta fase se pueden ver todos los estudios recuperados manualmente por el investigador que fueron importados a la herramienta StaRt, como se muestra en la Figura 26.

The screenshot displays the 'General information' section of the StaRt interface. It includes a search string field with the following text: `{{(acmdlTitle:(emotion) OR acmdlTitle:(emotions) OR acmdlTitle:(mood) OR acmdlTitle:(moods) OR acmdlTitle:(affect) OR acmdlTitle:(affcmdlTitle:(self-report)))}}`. Below this, the search machine is set to 'ACM', the number of papers is '0', and the date of the search is blank. An 'Observations' field is also present. At the bottom, there is an 'Import Reference File' section with four buttons: 'BIBTEX', 'MEDLINE', 'RIS', and 'Cochrane'. The 'BIBTEX' button is highlighted with a red border.

Figura 26: Importar resultados obtenidos de la búsqueda en ACM

En la Figura 27 muestra el menú de la fase de Selección. Se puede observar cuántos estudios fueron clasificados como aceptados, rechazados, duplicados o aún sin clasificar. Si se selecciona la opción de los documentos aún sin clasificar se mostrarán todos los estudios de la búsqueda. Al hacer doble click en uno de los estudios se abrirá un panel con pestañas relacionado con un estudio en específico como se muestra en la Figura 28.

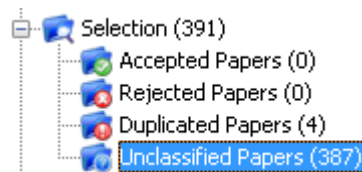


Figura 27: Menú fase selección.

En el panel de la Figura 28, en la primera y segunda pestaña, se pueden revisar algunos datos del estudio, el investigador aplica los criterios adecuados y además puede modificar el estado (aceptado, rechazado, aun sin clasificar, duplicados) y la prioridad de lectura del estudio. En la cuarta pestaña, se puede ver una cuadrícula con una similitud entre el estudio que está leyendo con los otros .

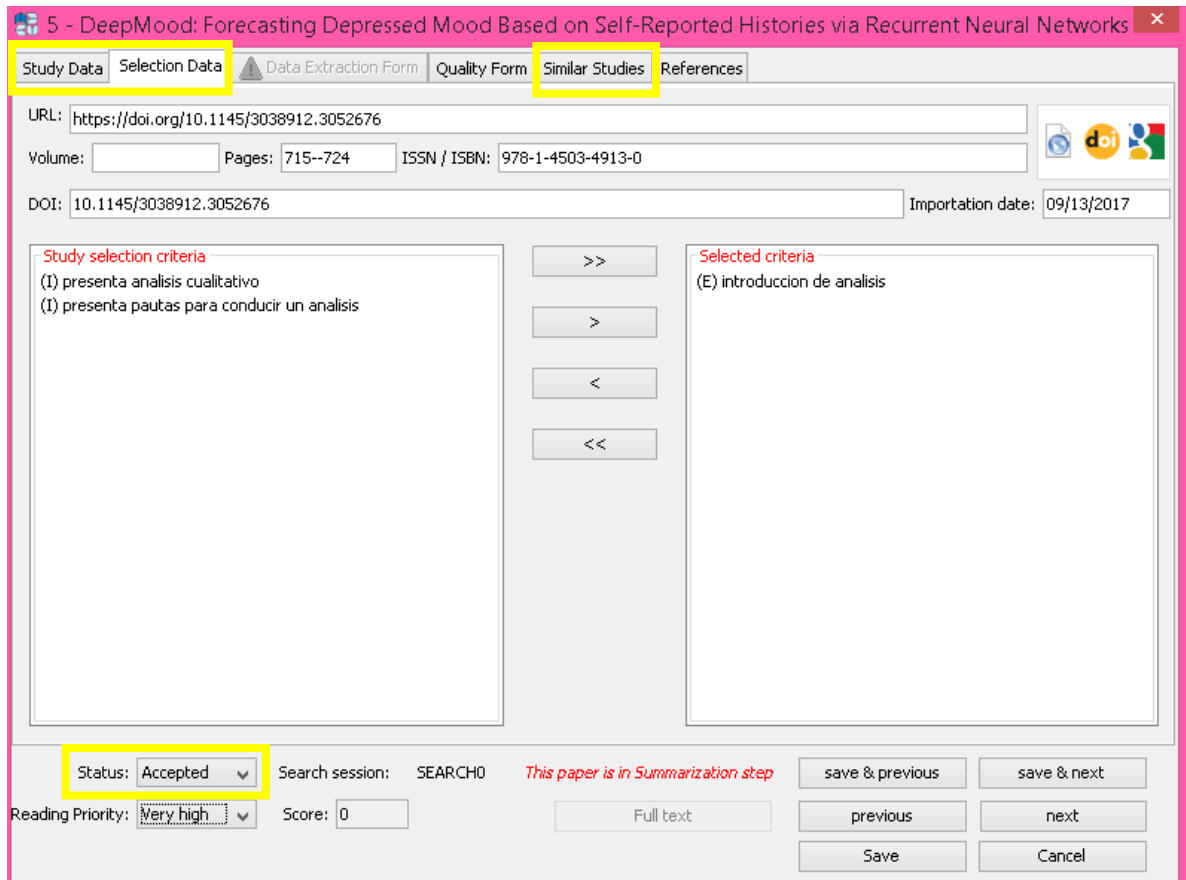


Figura 28: Panel de un estudio en específico.

En la etapa de extracción es necesario seguir la estrategia definida en el protocolo. Aquí los criterios de inclusión y exclusión se aplican nuevamente, teniendo en cuenta el documento completo.

Luego de leer el documento completo se puede establecer el estudio como rechazado si no es relevante. Sin embargo, si el estudio es relevante se debe llenar el formulario de extracción de información. Al final de la etapa de extracción se tiene un conjunto de estudios que fueron aceptados en ambas etapas de selección y extracción. Estos estudios restantes deben ser considerados en la fase de resumen.

3.4. Sobre las aplicaciones encontradas.

Referente a las cuatro aplicaciones analizadas anteriormente se puede decir que tienen algún grado de similitud con la herramienta computacional a desarrollar en este proyecto.

- La base de datos Epistemonikos permite extraer al usuario los metadatos de los artículos en un formato llamado RIS que es distinto al formato de la herramienta computacional siendo este el formato BibTeX. El problema de esta extracción es que solo deja extraer 100 datos de la búsqueda; esto quiere decir que si se desea realizar una búsqueda más amplia no se podrá obtener todas las referencias de los artículos.
- Mendeley y JabRef permite extraer las referencias bibliográficas en formato BibTeX al igual que la herramienta computacional a desarrollar, el problema es de la inconsistencia que existe entre la búsqueda realizada y la extracción de las referencias. Sin embargo, no solucionaría el problema que tienen los investigadores en estar ingresando manualmente a cada una de las bases de datos.
- La herramienta StArt no puede recuperar los estudios automáticamente de la cadena de búsqueda ingresada en el motor, por lo que el investigador tendrá que ingresar a la base de datos elegida en el protocolo, exportar los metadatos de los artículos, y luego importarlos a la herramienta StArt manualmente; esto implica que el proceso extracción de referencias bibliográficas no es automatizado. La ventaja de esta herramienta es que está la opción de eliminar los estudios duplicados.

Capítulo 4

Desarrollo de la herramienta Computacional.

4.1 Primer Incremento de la Herramienta

Para este proyecto se desarrollaron dos incrementos de la herramienta. El primer incremento se basó en crear las conexiones con las distintas bases de datos en línea. Estas conexiones se lograron con diferentes métodos, dependiendo la elección de un método sobre otro del comportamiento de las bases de datos en las que se quiere trabajar.

Como se mencionó anteriormente en la sección 2.4, para poder extraer información de las bases de datos se deberá procesar código HTML desde una URL, se realiza un requerimiento a dicha dirección usando el protocolo HTTP. Es por esto que se mostrarán todas las URL de las bases de datos en las cuales la herramienta computacional se conectará para poder trabajar sobre ellas, para seguidamente mostrar el contenido de las direcciones de las bases de datos y observar en qué parte se debe ingresar para poder extraer la información que se requiere, para así conocer cuál debe ser el resultado esperado de la ejecución de la herramienta.

Paralelamente se mostrará la forma de ingresar un string de búsqueda en cada base de datos científica. Esto se tuvo que analizar porque los motores de búsquedas de éstas funcionan de forma diferente. El motor de búsqueda que se implementó en la herramienta está basado en los criterios utilizados por las bases de datos científicas, y los campos por los que se van a buscar en las páginas son: a) Título, b) Palabra Clave, c) Abstract.

Se ocupará la siguiente cadena de búsqueda para ir mostrando con ejemplos el ingreso a las bases de datos.

(emotion or mood) and (interaction or share or “self-report”)

4.1.1. Búsquedas en las bases de datos y Resultados.

Lo primero que se debió hacer fue realizar búsquedas en las distintas bases de datos, con el objetivo de saber si el string de búsqueda generado era el correcto, para ello se hicieron pruebas necesarias, para determinar si se acepta o se rechaza las cadenas de búsquedas.

Con el profesor guía se acordó lo siguiente en la fase de prueba:

- Se toma una muestra aleatoria de 50 resultados a partir de una gran cantidad de datos arrojados y se somete a pruebas para verificar si estos contienen las palabras claves de la cadena de búsqueda
- Si el porcentaje de la cantidad de resultados correctos de la muestra es igual o superior a 80%, el string de búsqueda generado es aceptado.
- Si el porcentaje de la cantidad de resultados correctos de la muestra no pasa la prueba, se tendrá que modificar el string de búsqueda hasta que sea aceptado.

❖ *ACM Digital Library*

La siguiente tabla muestra un ejemplo de las cadenas de búsquedas por campos (Titulo, Abstract, Keyword) que se ingresaron en la base de datos ACM para realizar las búsquedas.

Tabla 3: Formato de cadenas de búsquedas por campo en la base de datos ACM

Campos	Ejemplo del Formato de una string de búsqueda
Título	((acmdlTitle:(emotion) OR acmdlTitle:(mood)) AND (acmdlTitle:(interaction) OR acmdlTitle:(share) OR acmdlTitle:(“self-report”))
Abstract	((recordAbstract:(emotion) OR recordAbstract:(mood)) AND (recordAbstract:(interaction) OR recordAbstract:(share) OR recordAbstract:(“self-report”))
Keyword	((keywords.author.keyword:(emotion) OR keywords.author.keyword:(mood)) AND (keywords.author.keyword:(interaction) OR keywords.author.keyword:(share) OR keywords.author.keyword:(“self-report”))
Título o Abstract	(((acmdlTitle:(emotion) OR acmdlTitle:(mood)) AND (acmdlTitle:(interaction) OR acmdlTitle:(share) OR acmdlTitle:(“self-report”))) OR (((recordAbstract:(emotion) OR recordAbstract:(mood)) AND (recordAbstract:(interaction) OR recordAbstract:(share) OR recordAbstract:(“self-report”)))

Título o Keyword	<p>((acmdlTitle:(emotion) OR acmdlTitle:(mood)) AND (acmdlTitle:(interaction) OR acmdlTitle:(share) OR acmdlTitle:"self-report"))</p> <p>OR</p> <p>((keywords.author.keyword:(emotion) OR keywords.author.keyword:(mood)) AND (keywords.author.keyword:(interaction) OR keywords.author.keyword:(share) OR keywords.author.keyword:"self-report"))</p>
Abstract o Keyword	<p>((recordAbstract:(emotion) OR recordAbstract:(mood)) AND (recordAbstract:(interaction) OR recordAbstract:(share) OR recordAbstract:"self-report"))</p> <p>OR</p> <p>((keywords.author.keyword:(emotion) OR keywords.author.keyword:(mood)) AND (keywords.author.keyword:(interaction) OR keywords.author.keyword:(share) OR keywords.author.keyword:"self-report"))</p>

<p>Título o Abstract o Keyword</p>	<p>((acmdlTitle:(emotion) OR acmdlTitle:(mood)) AND (acmdlTitle:(interaction) OR acmdlTitle:(share) OR acmdlTitle:"self-report"))</p> <p>OR</p> <p>((recordAbstract:(emotion) OR recordAbstract:(mood)) AND (recordAbstract:(interaction) OR recordAbstract:(share) OR recordAbstract:"self-report"))</p> <p>OR</p> <p>((keywords.author.keyword:(emotion) OR keywords.author.keyword:(mood)) AND (keywords.author.keyword:(interaction) OR keywords.author.keyword:(share) OR keywords.author.keyword:"self-report"))</p>
--	---

Tabla 4: Porcentaje de resultados correctos arrojados por ACM.

Campos	% de resultados correctos de la muestra
Título	100%
Abstract	100%
Keyword	100%
Título o Abstract	100%
Título o Keyword	100%
Abstract o Keyword	100%
Título o Abstract o keyword	100%

❖ *Taylor and Francis.*

Tabla 5: Formato de cadenas de búsquedas por campos en Taylor and Francis.

Campos	Ejemplo del Formato de una string de búsqueda
Título	((title:emotion OR title:mood) AND (title:interaction OR title:share OR title: "self-report"))
Abstract	((abstract:emotion OR abstract:mood) AND (abstract:interaction OR abstract:share OR abstract:"self-report"))
Keyword	((keyword:emotion OR keyword:mood) AND (keyword:interaction OR keyword:share OR keyword:"self-report"))
Título o Abstract	(((title:emotion OR title:mood) AND (title:interaction OR title:share OR title:"self-report")) OR ((abstract:emotion OR abstract:mood) AND (abstract:interaction OR abstract:share OR abstract:"self-report"))
Título o Keyword	(((title:emotion OR title:mood) AND (title:interaction OR title:share OR title:"self-report")) OR ((keyword:emotion OR keyword:mood) AND (keyword:interaction OR keyword:share OR keyword:"self-report"))

Abstract o Keyword	<p>((abstract:emotion OR abstract:mood) AND (abstract:interaction OR abstract:share OR abstract:"self-report"))</p> <p>OR</p> <p>((keyword:emotion OR keyword:mood) AND (keyword:interaction OR keyword:share OR keyword:"self-report"))</p>
Título o Abstract o Keyword	<p>((title:emotion OR title:mood) AND (title:interaction OR title:share OR title:"self-report"))</p> <p>OR</p> <p>((abstract:emotion OR abstract:mood) AND (abstract:interaction OR abstract:share OR abstract:"self-report"))</p> <p>OR</p> <p>((keyword:emotion OR keyword:mood) AND (keyword:interaction OR keyword:share OR keyword:"self-report"))</p>

Tabla 6: Porcentaje de resultados correctos arrojador por Taylor and Francis.

Campos	% de la cantidad de resultados correctos
Título	100%
Abstract	92%
Keyword	100%
Título o Abstract	84%
Título o Keyword	100%
Abstract o Keyword	94%
Título o Abstract o keyword	96%

❖ *Springer.*

Tabla 7: Formato de cadena de búsqueda por campo título en Springer.

Campos	Ejemplo del Formato de un string de búsqueda
Título	(title=(emotion*) or title=(mood*)) and (title=(interaction*) or title=(share*) or title=("self-report"*))

Tabla 8: Porcentaje de Resultados correctos arrojador por Springer.

Campos	% de la cantidad de resultados correctos
Título	100%

❖ *Emerald.*

Para esta base de datos solo se ingresa el string de búsqueda sin ningún formato (Ver Figura 29), se debe tener cuidado que al ingresar la cadena los operadores booleanos deben ser con mayúscula.

La Figura 29 muestra una búsqueda, en donde se ingresa la cadena de búsqueda y se eligen una de las opciones, para el caso del proyecto (Título, Abstract, Keywords).

(emotion OR mood) AND (interaction OR share OR "self-report")

Content Item title

- Anywhere
- Author
- Abstract
- Publication title
- Sp Issue/Vol title
- Content Item title**
- Keywords
- ISSN/ISBN/EISBN
- Volume
- Issue
- Page
- Caption

Figura 29: Búsqueda por Campos en Emerald.

Tabla 9: Porcentaje de resultados correctos arrojados por Emerald.

Campos	% de la cantidad de resultados correctos
Título	100%
Abstract	96%
Keyword	100%
Título o Abstract	96%
Título o Keyword	100%
Abstract o Keyword	96%
Título o Abstract o keyword	98%

❖ *Wiley & Sons.*

Al igual que la base de datos Emerald, se ingresa el string de búsqueda sin formato solo los operadores booleanos deben estar en mayúscula, y elegir una de las opciones que entrega Wiley& Sons (Título, Abstract y Keywords), como se ve en la Figura 30.

SEARCH FOR

(emotion OR mood) AND (interaction OR sha

AND

AND

Add another row >

SPECIFY DATE RANGE

All Dates

In the last

Between

Please enter 4-digit years.

All Fields

- All Fields
- Publication Titles
- Article Titles** **earch**
- Author
- FullText
- Abstract
- Author Affiliation
- Keywords
- Funding Agency
- ISBN
- ISSN **earch**

Figura 30: Búsqueda por campo en Wiley & Sons

Tabla 10: Porcentaje de resultados correctos arrojados por Wiley & Sons

Campos	% de la cantidad de resultados correctos
Título	100%
Abstract	100%
Keyword	100%
Título o Abstract	98%
Título o Keyword	100%
Abstract o Keyword	96%
Título o Abstract o keyword	98%

❖ *IEEEExplore.*

Tabla 11: Formato de cadenas de búsquedas por campo en IEEEExplore.

Campos	Ejemplo del Formato de una string de búsqueda
Título	((("Document Title":emotion OR "Document Title":mood) AND ("Document Title":interaction OR "Document Title":share OR "Document Title":self report))
Abstract	(("Abstract":emotion OR "Abstract":mood) AND ("Abstract":interaction OR "Abstract":share OR "Abstract":self-report))
Keyword	(("Author Keywords":emotion OR "Author Keywords":mood) AND ("Author Keywords":interaction OR "Author Keywords":share OR "Author Keywords":self-report))
Título o Abstract	((("Document Title":emotion OR "Document Title":mood) AND ("Document Title":interaction OR "Document Title":share OR "Document Title":self report)) OR (("Abstract":emotion OR "Abstract":mood) AND ("Abstract":interaction OR "Abstract":share OR "Abstract":self-report))))
Título o Keyword	((("Document Title":emotion OR "Document Title":mood)

	<p>AND ("Document Title":interaction OR "Document Title":share OR "Document Title":self report"))</p> <p>OR</p> <p>(("Author Keywords":emotion OR "Author Keywords":mood</p> <p>AND ("Author Keywords":interaction OR "Author Keywords":share OR "Author Keywords":self-report"))</p>
Abstract o Keyword	<p>((("Abstract":emotion OR "Abstract":mood)</p> <p>AND ("Abstract":interaction OR "Abstract":share OR "Abstract":self-report"))</p> <p>OR</p> <p>(("Author Keywords":emotion OR "Author Keywords":mood</p> <p>AND ("Author Keywords":interaction OR "Author Keywords":share OR "Author Keywords":self-report"))</p>
Título o Abstract o Keyword	<p>((("Document Title":emotion OR "Document Title":mood)</p> <p>AND ("Document Title":interaction OR "Document Title":share OR "Document Title":self report"))</p> <p>OR</p> <p>(("Abstract":emotion OR "Abstract":mood)</p>

	<p>AND ("Abstract":interaction OR "Abstract":share OR "Abstract":self-report))</p> <p>OR</p> <p>(("Author Keywords":emotion OR "Author Keywords":mood</p> <p>AND ("Author Keywords":interaction OR "Author Keywords":share OR "Author Keywords":self-report)))</p>
--	---

Tabla 12: Porcentaje de resultados correctos arrojados por IEEEExplore.

Campos	% de la cantidad de resultados correctos
Título	100%
Abstract	100%
Keyword	100%
Título o Abstract	100%
Título o Keyword	100%
Abstract o Keyword	100%
Título o Abstract o keyword	100%

❖ *Science Direct.*

La Figura 31 muestra una búsqueda, en donde se ingresa la cadena de búsqueda con los operadores booleanos en mayúscula y se eligen una de las opciones, para el caso del proyecto (Título, Abstract, Keywords).

The screenshot displays the Science Direct search interface. At the top, under the heading "Search for", there is a search input field containing the query "(emotion OR mood) AND (interaction OR share OR "self-repoi". Below the input field is a dropdown menu currently set to "AND". To the right of the search field is a dropdown menu for selecting search fields, with "All Fields" selected and a list of options including "All Fields", "Abstract, Title, Keywords", "Authors", "Specific Author", "Source Title", "Title" (highlighted in blue), "Keywords", "Abstract", "References", "ISSN", and "ISBN". Below the search field is a section titled "Refine your search" with checkboxes for "Journals" (checked), "Books" (checked), and "Open Access articles only" (unchecked). At the bottom, there is a list of subjects: "Chemistry", "Computer Science" (highlighted in blue), "Decision Sciences", and "Earth and Planetary Sciences".

Figura 31: Búsqueda por campos en Science Direct.

Tabla 13: Porcentaje de resultados correctos arrojados por Science Direct.

Campos	% de la cantidad de resultados correctos
Título	100%
Abstract	96%
Keyword	100%
Título o Abstract	100%
Título o Keyword	100%
Abstract o Keyword	100%
Título o Abstract o keyword	100%

❖ *Scopus.*

La Figura 32 muestra una búsqueda, en donde se ingresa la cadena de búsqueda con los operadores booleanos en mayúscula y se eligen una de las opciones, para el caso del proyecto (Título, Abstract, Keywords).

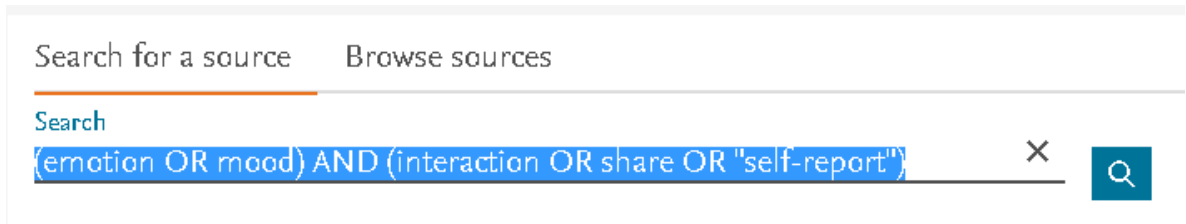


Figura 32: Búsqueda por campos en Scopus

Tabla 14: Porcentaje de resultados correctos arrojados por Scopus.

Campos	% de la cantidad de resultados correctos
Título	100%
Abstract	100%
Keyword	100%
Título o Abstract	100%
Título o Keyword	100%
Abstract o Keyword	100%
Título o Abstract o keyword	100%

Las cadenas de búsquedas que fueron ingresados en las bases de datos fueron aceptadas, ya que el porcentaje de la cantidad de resultados correctos fue mayor a 80% en todos los formatos de las bases de datos analizadas, por lo tanto se ocuparán éstos formatos para desarrollar la herramienta computacional.

4.1.2 Análisis de las bases de datos.

A la hora de extraer los metadatos de los artículos, lo primero que se tendrá que realizar es determinar qué información se necesitará, de dónde se va a obtener, y cómo se va a procesar. Para este caso es de vital importancia establecer la estructura de los sitios web a los que se quiere extraer la información, ya que los resultados de la herramienta dependerán del contenido de las bases de datos. En esta sección se mostrará con ejemplos cuál debe ser el resultado esperado de la ejecución de la herramienta, y además mostrar el código fuente de las bases de datos. En esta sección solo se mostrará para la base de datos ACM; las demás bases de datos se encuentran en el Apéndice B.

❖ ACM

Para mostrar el contenido de esta base de datos, se ocupará el formato del título de la Tabla 2., se van a resaltar los elementos necesarios que servirán para el desarrollo de la herramienta.

Para realizar las búsquedas, se debió acceder a “búsquedas avanzadas”. Esta búsqueda facilita un formulario en el que se puede buscar por una serie de campos determinados (Título, Abstract, Palabra Clave) y filtrar por fecha de la publicación, en donde se ingresa el string de búsqueda que se desea con su determinado formato (Tabla 3 Formato Título) en el formulario como se muestra en la Figura 33.

Edit Query Query syntax is generated automatically; editing below will override this, to revert back, [Reset Query](#)

((acmdlTitle:(emotion) OR acmdlTitle:(mood)) |

AND (acmdlTitle:(interaction) OR acmdlTitle:(share) OR acmdlTitle:("self-report"))

View Full Query Syntax

```
"query": { AND (acmdlTitle:(interaction) OR acmdlTitle:(share) OR acmdlTitle:("self-report")) }  
"filter": {owners.owner=HOSTED}
```

Figura 33: Búsqueda avanzada de ACM.

Una vez que se haya ingresado la cadena de búsqueda, la base de datos se redirigirá a una “página de resultados” (Ver Figura 34); en la parte superior de la página se indica la URL Figura 34 (1) como se puede observar está compuesta de parámetros, los cuales serán descritos a continuación.

- Subrayado amarillo es la URL solicitada.
- Subrayado rojo es la cadena de búsqueda que se ingresó para el ejemplo.
- Subrayado verde es un parámetro que va cambiando a medida que cambia el número de página; como la cantidad de resultados por página son siempre 20, el parámetro está vacío porque se encuentra en la primera página, pero si se avanza a la segunda el parámetro cambiará a $\$start=21$ y así sucesivamente hasta recorrer todas las páginas.

En la parte superior izquierda Figura 34 (2) de la página se indica el número de registros recuperados que pertenecen a publicaciones de ACM, en Figura 34 (3) la cantidad de páginas, y en Figura 34 (4) son los registros que arroja la búsqueda y servirán para acceder a la información más detallada del documento como el DOI; este parámetro es el que servirá para acceder a los metadatos de los registros.

ACM

1 [http://dl.acm.org/results.cfm?query=\(\(acmdlTitle%3A\(emotion\) OR acmdlTitle%3A\(mood\) \) AND \(acmdlTitle%3A\(interaction\) OR acmdlTitle%3A\(share\) OR acmdlTitle%3A\("self-report"\)\)\)&start=...](http://dl.acm.org/results.cfm?query=((acmdlTitle%3A(emotion) OR acmdlTitle%3A(mood)) AND (acmdlTitle%3A(interaction) OR acmdlTitle%3A(share) OR acmdlTitle%3A()

2 381 results found

Export Results: [bibtex](#) | [endnote](#) | [acmref](#) | [csv](#)

Result 1 – 20 of 381

3 Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) >>

Sort by: [relevance](#)

1 [What Can Self-Reports and Acoustic Data Analyses on Emotions Tell Us?](#) 4

[Samaneh Soleimani, Eme Lai-Chong Law](#)

June 2017 DIS '17: Proceedings of the 2017 Conference on Designing Interactive Systems

Publisher: ACM

Bibliometrics: Citation Count: 0
Downloads (6 Weeks): 44, Downloads (12 Months): 44, Downloads (Overall): 44

Full text available: [PDF](#)

There are two approaches to measure people's experiences: memory-based and moment-based. Whereas the memory-based approaches are susceptible to the peak-end effect, the moment-based approaches appear to better reflect the experience of the present. In this paper, we propose that emotional assessment of think aloud verbalisations is a moment-based methodology for ...

Keywords: user experience, acoustic analysis, emotion, evaluation

[\[result highlights\]](#)

2 [Automatic recognition of self-reported and perceived emotion: does joint modeling help?](#)

[Biqiao Zhang, Georg Essl, Emily Mower Provost](#)

October 2016 ICMI 2016: Proceedings of the 18th ACM International Conference on Multimodal Interaction

Publisher: ACM

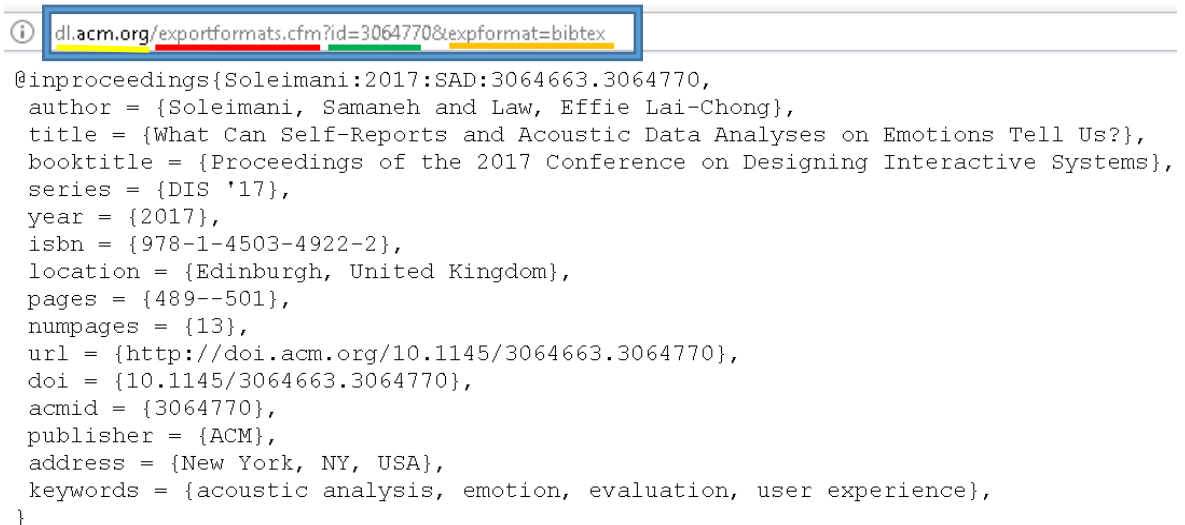
Bibliometrics: Citation Count: 1
Downloads (6 Weeks): 3, Downloads (12 Months): 105, Downloads (Overall): 105

Figura 34: Página de resultados obtenidos en la búsqueda de ACM.

Luego de seleccionar uno de los artículos de la búsqueda, se redireccionará a otra página en donde se encuentra una opción para descargar las referencias bibliográficas en formato BibTeX; se selecciona esa opción y para poder obtener esta URL que se muestra en la Figura 35, se deberá acceder a las peticiones HTTP tipo POST y observar los parámetros que entrega la página. Con esos parámetros se construye la URL para que entregue la referencia bibliográfica. Los parámetros son los siguientes:

- Subrayado Amarillo es la URL solicitada.
- Subrayado en rojo es un parámetro que indica exportar la referencia; ese parámetro es siempre el mismo, no cambia.
- Subrayado en verde es el parámetro del DOI, este cambia a media que se selecciona los distintos artículos de la consulta.
- Subrayado naranja es un parámetro que indica que está en formato BibTeX, no cambia.

En la Figura 35 se pueden observar las referencias bibliográficas en formato BibTeX del registro seleccionado.



```
@inproceedings{Soleimani:2017:SAD:3064663.3064770,  
  author = {Soleimani, Samaneh and Law, Effie Lai-Chong},  
  title = {What Can Self-Reports and Acoustic Data Analyses on Emotions Tell Us?},  
  booktitle = {Proceedings of the 2017 Conference on Designing Interactive Systems},  
  series = {DIS '17},  
  year = {2017},  
  isbn = {978-1-4503-4922-2},  
  location = {Edinburgh, United Kingdom},  
  pages = {489--501},  
  numpages = {13},  
  url = {http://doi.acm.org/10.1145/3064663.3064770},  
  doi = {10.1145/3064663.3064770},  
  acmid = {3064770},  
  publisher = {ACM},  
  address = {New York, NY, USA},  
  keywords = {acoustic analysis, emotion, evaluation, user experience},  
}
```

Figura 35: Ejemplo de referencia bibliográfica de ACM.

4.1.3. Estructura de las bases de datos

En esta sección se presentará el contenido del sitio web ACM. Los demás contenidos de las otras bases de datos se encuentran en el Apéndice A de este informe.

La estructura de las bases de datos es un aspecto que hay que tener en cuenta al realizar web scraping. Se necesitará saber la accesibilidad de los datos de origen. Aquí se consideran dos aspectos que hay que tener en contemplación al momento de hacer la extracción de los datos:

- Según la estructura de presentación de los datos: Si todos los datos a extraer están en una única página o si están en varias páginas.
- Según la manera en que la página los presenta: Los datos se pueden mostrar solicitando una página web específica para cada dato o conjunto de datos, es decir, se puede obtener el enlace de los datos, o por el contrario, desde la misma página web se van cargando dinámicamente y presentando los datos usando Javascript y/o técnicas de AJAX.

Como se dijo anteriormente, se mostrará el contenido HTML de lo que se debe extraer para poder sacar lo que se desea que son las referencias bibliográficas de los resultados. A continuación se mostrará paso a paso los datos de las páginas que se necesitaran para la extracción de los metadatos.

1. Se requiere extraer la cantidad de resultados que se obtienen al realizar la búsqueda. Este dato se encuentra en una sola página como se muestra en la Figura 34 (2). En la estructura de esta página existe una clase contenedor con “id=searchtots”; dentro de esta clase se localiza la etiqueta “strong” que es donde se encuentra la cantidad de resultados; ahí se tendrá que llegar para obtener este dato, como se muestra en la Figura 36.

```
▼ <div id="searchtots">  
  <strong>433</strong>  
  results found  
</div>
```

Figura 36: Extracto de código (cantidad de resultados) de ACM

2. Se requiere conseguir los DOI de los resultados encontrados, para eso se inspecciona el código se observa una clase contenedor con “<div id=’results’>”, dentro de este contenedor se puede encontrar la clase “<div class=’details’>”, como se muestra en la Figura 37.

```
▼ <div id="results"> 1  
  <div class="pagerange">Result 1 - 20 of 433</div>  
  ▶ <div class="pagelogic">... </div>  
  <br clear="all">  
  ▶ <div id="resultmenu">... </div>  
  <br clear="all">  
  ▶ <div class="numbering">... </div>  
  ▶ <div class="details">... </div> 2
```

Figura 37: Extracto de código para obtener todos los ítems de ACM.

3. Dentro de cada etiqueta “details” se podrá encontrar la clase “title”. Esta clase tiene un único hijo que es la etiqueta “a”. A la herramienta lo que le interesa es obtener el “href” para poder conseguir el id (DOI).

4.1.4. Procesamiento

Una vez obtenido todo el HTML de las páginas con la herramienta curl de PHP, solo se debe extraer la información que se necesita, eliminando todo lo restante. Para esto se ocupó “simple HTML DOM Parser” que es una librería para parsear el DOM, y así obtener contenidos específicos como por ejemplo, en la Figura 39 es un extracto de código de la herramienta computacional desarrollada para extraer el DOI de la base de datos ACM si observan en

- En Figura 39 (1) la herramienta desarrollada encuentra la etiqueta `<div id =”results”>` de la Figura 37 (1).
- En Figura 39 (2) encuentra `<div class = “details”>` de la Figura 37 (2).
- En Figura 39 (3) encuentra `<div class= “title”>-> href` de la Figura 38 (1), como se obtiene todo el enlace (Figura 38 (2)), y se necesita obtener solo el id, se deberá iterar sobre el código aplicando sucesivas veces la función explode de PHP como se muestra en la Figura 39 (4), esta sirve para separar la información hasta obtener únicamente la información que interesa que en este caso es el DOI Figura 38 (3).

```
<div class="details"> 1
  <div class="title"> 3
    <a href="citation.cfm?id=3064776&CFID=1000632074&CFTOKEN=43801171" 2
      ... </a>
  </div>
```

Figura 38: Extracto de código para obtener el doi de ACM.

```

private function getPagina($pagina) {
    $this->html = new simple_html_dom();
    $this->html->load($this->getHtmlResultados
        ($this->url_base .
        $this->url_query .
        $this->query .
        $this->desde_anio.
        $this->start .
        ($this->resultados_por_pagina
        * $pagina)));
    $resultado = $this->html->find('#results', 0);
    $detalles = $resultado->find('div.details');
    $contador = 1;
    foreach ($detalles as $detalle) {
        $href item = $detalle->find('div.title a', 0)->href;
        /*dividimos el string solo para obtener
        el DOI*/
        $id = explode('=', explode('&', $id)[0])[1];
    }
}

```

Figura 39: Extracto de código del archivo acm.php

4.2. Segundo Incremento de la Herramienta

Para iniciar el proceso de construcción del segundo incremento de la herramienta computacional, se determinó qué cambios se debían realizar. Se procedió a agregar:

- Búsqueda por fecha.
- Construcción de la interfaz gráfica.
- Registro de referencias bibliográficas en un archivo.

4.2.1. Búsqueda por fecha

❖ Fecha ACM

El parámetro que se le debe agregar a la URL es &dte, si se agrega ese parámetro lo que hace la base de datos ACM es refinar la búsqueda que realice el usuario. En la Figura 40 se realizó una búsqueda con una fecha del 2013 esto quiere decir que busca los registros que están durante y después de la fecha ingresada.

```
https://dl.acm.org/results.cfm?query=((acmdlTitle%3A(emotion) OR  
acmdlTitle%3A(mood) ) AND (acmdlTitle%3A(interaction) OR acmdlTitle%3A(share) OR  
acmdlTitle%3A("self-report"))&dte=2013
```

Figura 40: URL + filtro fecha ACM.

Las fechas de las otras bases de datos la pueden encontrar en el Anexo A de este informe.

4.2.2. Construcción de interfaz gráfica

En la construcción de la interfaz gráfica, primero se implementó un motor de búsqueda permitiendo la entrada de la cadena formadas por los investigadores, como se muestra en la Figura 41.

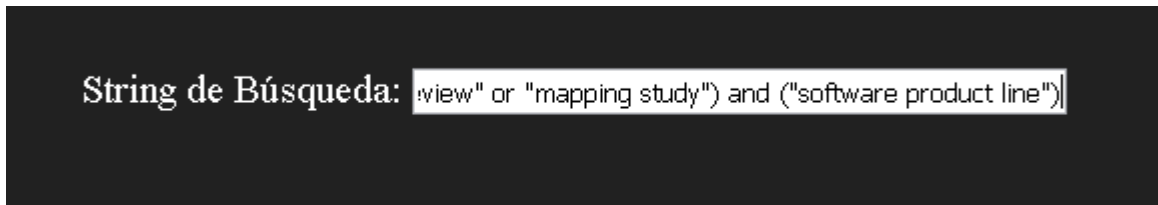


Figura 41: Ingreso cadena de búsqueda.

Además, la herramienta cuenta con un módulo de búsqueda de estudios el cual permite la elección de las bases de datos científicas. Este módulo incrustado en la interfaz permite la selección de las fuentes nombradas en la delimitación del problema, como se muestra en la Figura 42.

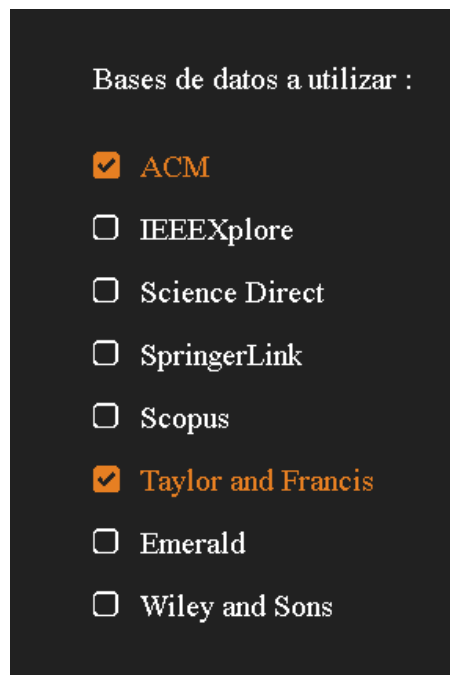
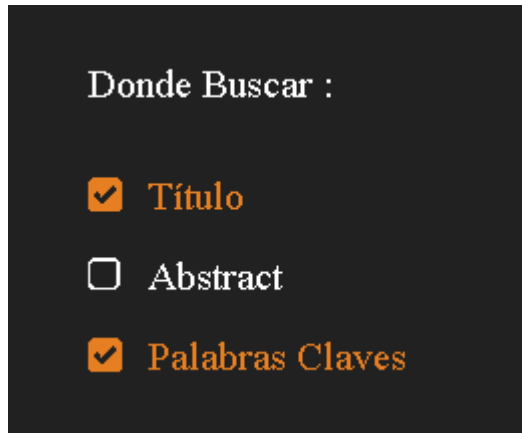


Figura 42: Seleccionar bases de datos.

En la Figura 43 existe un módulo para refinar la búsqueda, permitiendo buscar la cadena de búsqueda ingresada en el título, abstract y palabras claves (a), pudiéndose también refinar por un rango de fechas (b).



Donde Buscar :

- Título
- Abstract
- Palabras Claves

(a)



Escoger Rango de Fecha:

Desde mes ▼ Año ▼

Hasta mes ▼ Año ▼

(b)

Figura 43: Filtros de la Herramienta Computacional.

Una vez que se ingresa la cadena de búsqueda, se seleccionan las fuentes, los filtros, y se ingresa una fecha (opcional), se presiona el botón “Buscar”, y la herramienta al realizar la búsqueda se conecta con las bases de datos seleccionadas por el usuario, recopilando las referencias bibliográficas de los artículos dependiendo de lo que se eligió al realizar la búsqueda.

4.2.3. Guardar metadatos en un archivo

Se guardan las referencias bibliográficas extraídas por la herramienta en un archivo. Lo que se debe hacer es crear un archivo y luego escribir en éste la información rescatada de las bases de datos. Con fopen se abre el archivo pasando como parámetro el nombre que va a tener siendo este “archivo.txt”, y como segundo parámetro “a” indica que se está abriendo el fichero para solo escritura. Con fwrite se escribe dentro del archivo la información recopilada por el software con la variable \$this->html_segunda.

```
/*Guardar metadatos en un archivo*/  
$file = fopen("archivo.txt", "a");  
fwrite($file, $pre . PHP_EOL);  
fclose($file);
```

Figura 44: Guardar en un archivo.

Capítulo 5

Pruebas del Sistema

Este capítulo mostrará los resultados obtenidos después de desarrollar el proyecto. Se podrán ver diferentes pruebas que se han llevado a cabo para comprobar la consistencia de los datos extraídos y el tiempo que se demora la aplicación construida en extraer dichos datos.

Para las pruebas se ocupará el string de búsqueda: (emotion or mood) and (interaction or share or “self-report”), con un filtro fecha del 2016 hasta el 2017 para que la muestra sea más pequeña.

Para comprobar el funcionamiento del sistema, lo que se debió hacer fue ejecutar las bases de datos y los filtros por separado, Por ejemplo:

- ❖ Seleccionar una **base de datos** y seleccionar filtro **Título**.
- ❖ Seleccionar una **base de datos** y seleccionar filtro **Abstract**.
- ❖ Seleccionar una **base de datos** y seleccionar filtro **keyword**.
- ❖ Seleccionar una **base de datos** y seleccionar filtro **Título y Abstract**.
- ❖ Seleccionar una **base de datos** y seleccionar filtro **Título y keyword**.
- ❖ Seleccionar una **base de datos** y seleccionar filtro **Abstract y Keyword**.
- ❖ Seleccionar una **base de datos** y seleccionar filtro **Título, Abstract y Keyword**.

En el momento que se ejecutó el sistema para cada una de las opciones del ejemplo, paralelamente se realizó la búsqueda en la base de datos elegida, para ir comprobando inmediatamente si los resultados extraídos son correctos con respecto a los resultados de la base de datos.

5.1. Resultados entre la cantidad de referencias bibliográficas de las bases de datos v/s la herramienta computacional para una búsqueda en el título.

En la Tabla 15 se muestra una comparación entre la cantidad de registros que arroja cada una de las bases de datos para el filtro **Título** y la cantidad de registro que arroja la aplicación, midiendo el tiempo (minutos) que se demora en extraer dichos registros.

Como pueden observar existe consistencia entre los registros extraídos de las bases de datos y los registros que se extrajeron de la aplicación.

Tabla 15: Pruebas del sistema de una búsqueda en el título.

Fuente	Cantidad de registros que arroja base de datos	Tiempo de extracción de la Herramienta computacional (minutos)	Cantidad de registros que arroja herramienta computacional
ACM	36	2	36
Taylor and Francis	38	1	38
Wiley & Sons	29	2	29
Emerald	3	0.3	3
Springer	69	1	69

5.2. *Resultados entre la cantidad de referencias bibliográficas de las bases de datos v/s la herramienta computacional para una búsqueda en el Abstract.*

En la Tabla 16 se muestra una comparación entre la cantidad de registros que arroja cada una de las bases de datos para el filtro **Abstract** y la cantidad de registro que arroja la aplicación, midiendo el tiempo (minutos) que se demora en extraer dichos registros.

Como pueden observar existe consistencia entre los registros extraídos de las bases de datos y los registros que se extrajeron de la aplicación, la base de datos ACM tuvo una inconsistencia de registros.

Tabla 16: Pruebas del sistema de una búsqueda en el abstract

Fuentes	Cantidad de registros que arroja base de datos	Tiempo de extracción de la Herramienta computacional (minutos))	Cantidad de registros que arroja herramienta computacional
ACM	165	10	162
Taylor and Francis	253	6	253
Wiley & Sons	118	5	118
Emerald	50	1	50

5.3. *Resultados entre la cantidad de referencias bibliográficas de las bases de datos v/s la herramienta computacional para una búsqueda en el Keyword.*

En la Tabla 17 se muestra una comparación entre la cantidad de registros que arroja cada una de las bases de datos para el filtro **Keyword** y la cantidad de registro que arroja la aplicación, midiendo el tiempo (minutos) que se demora en extraer dichos registros.

Como pueden observar existe consistencia entre los registros extraídos de las bases de datos y los registros que se extrajeron de la aplicación.

Tabla 17: Pruebas del sistema de una búsqueda en el keyword.

Fuentes	Cantidad de registros que arroja base de datos	Tiempo de extracción de la Herramienta computacional (minutos)	Cantidad de registros que arroja herramienta computacional
ACM	30	2	30
Taylor and Francis	23	0.6	23
Wiley & Sons	30	2	30
Emerald	20	8	20

5.4. *Resultados entre la cantidad de referencias bibliográficas de las bases de datos v/s la herramienta computacional para una búsqueda en el título y abstract.*

En la Tabla 18 se muestra una comparación entre la cantidad de registros que arroja cada una de las bases de datos para el filtro (**Título y Abstract**) y la cantidad de registro que arroja la aplicación, midiendo el tiempo (minutos) que se demora en extraer dichos registros.

Como pueden observar existe consistencia entre los registros extraídos de las bases de datos y los registros que se extrajeron de la aplicación, la base de datos ACM tuvo una inconsistencia de registros.

Tabla 18: Pruebas del sistema de una búsqueda en el título y abstract.

Fuentes	Cantidad de registros que arroja base de datos	Tiempo de extracción de la Herramienta computacional (minutos)	Cantidad de registros que arroja herramienta computacional desarrollada
ACM	181	12	174
Taylor and Francis	253	6	253
Wiley & Sons	118	3	118
Emerald	50	1	50

5.5. Resultados entre la cantidad de referencias bibliográficas de las bases de datos v/s la herramienta computacional para una búsqueda en el título y keyword.

En la Tabla 19 se muestra una comparación entre la cantidad de registros que arroja cada una de las bases de datos para el filtro (**Título y Keyword**) y la cantidad de registro que arroja la aplicación, midiendo el tiempo (minutos) que se demora en extraer dichos registros.

Como pueden observar existe consistencia entre los registros extraídos de las bases de datos y los registros que se extrajeron de la aplicación, la base de datos ACM tuvo una inconsistencia de registros.

Tabla 19: Pruebas del sistema para el título y keyword.

Fuentes	Cantidad de registros que arroja base de datos	Tiempo de extracción de la Herramienta computacional (minutos)	Cantidad de registros que arroja herramienta computacional
ACM	60	4	56
Taylor and Francis	43	1	43
Wiley & Sons	37	1	37
Emerald	10	0.3	10

5.6. Resultados entre la cantidad de referencias bibliográficas de las bases de datos v/s la herramienta computacional para una búsqueda en el abstract y keyword.

En la Tabla 20 se muestra una comparación entre la cantidad de registros que arroja cada una de las bases de datos para el filtro (**Abstract y Keyword**) y la cantidad de registro que arroja la aplicación, midiendo el tiempo (minutos) que se demora en extraer dichos registros.

Como pueden observar existe consistencia entre los registros extraídos de las bases de datos y los registros que se extrajeron de la aplicación, la base de datos ACM tuvo una inconsistencia de registros.

Tabla 20: Pruebas del sistema para el abstract y keyword.

Fuentes	Cantidad de registros que arroja base de datos	Tiempo de extracción de la Herramienta computacional (minutos)	Cantidad de registros que arroja herramienta computacional
ACM	174	10	170
Taylor and Francis	253	6	253
Wiley & Sons	119	3	119
Emerald	143	3	143

5.7. Resultados entre la cantidad de referencias bibliográficas de las bases de datos v/s la herramienta computacional para una búsqueda en el título, abstract y keyword.

En la Tabla 21 se muestra una comparación entre la cantidad de registros que arroja cada una de las bases de datos para el filtro (**Título, Abstract y Keyword**) y la cantidad de registro que arroja la aplicación, midiendo el tiempo (minutos) que se demora en extraer dichos registros.

Como pueden observar existe consistencia entre los registros extraídos de las bases de datos y los registros que se extrajeron de la aplicación, la base de datos ACM tuvo una inconsistencia de registros.

Tabla 21: Pruebas del sistema para el título, abstract y keyword.

Fuentes	Cantidad de registros que arroja base de datos	Tiempo de extracción de la Herramienta computacional (minutos)	Cantidad de registros que arroja herramienta computacional
ACM	189	13	174
Taylor and Francis	253	6	253
Wiley & Sons	119	3	119
Emerald	144	3	144

Si se observan las tablas desde 5.1 a 5.7, la herramienta computacional entregó resultados satisfactorios.

La base de datos ACM se comportó de forma distinta a las demás. Esta BD generalmente bloqueaba la IP, implicando no poder extraer los datos. Esto se debe a que ésta detectaba que se trataba de un rastreador web, es por esto que se ocupó la función sleep para que el indexador web pudiera retrasar su ejecución comportándose como si fuera un humano, implicando que el tiempo de extracción para esta base de datos sea mayor que el resto de las otras extracciones realizadas por la herramienta computacional desarrollada, pero sigue obteniendo los datos en un menor tiempo comparado si se realiza manualmente. Además, existe un pequeño margen de error en la consistencia de los datos. Si se mira en la Tabla 20, la base de datos arrojó 165 resultados y la herramienta construida arrojó 162 resultados. Este problema no se pudo detectar, se revisó exhaustivamente el código HTML de ACM, para poder encontrar si se trataba de un código mal estructurado pero todas las etiquetas del HTML estaban con el mismo nombre, y además es que solo en la primera y segunda página existe el margen de error, cabe mencionar que la consistencia de los datos extraídos por la herramienta construida es mayor al 90%.

Para Springer, Taylor and Francis, Emerald y Wiley & Sons los resultados encontrados fueron todos satisfactorios; el tiempo de ejecución es mínimo en comparación a una extracción manual de las referencias en las bases de datos.

Capítulo 6

Conclusión

Por último, se van a mostrar las conclusiones finales que se extraen tras la realización del proyecto.

6.1. Principales dificultades

Las dificultades encontradas durante el desarrollo de la herramienta se describen en los siguientes puntos.

- **Completo desconocimiento de la técnica de web scraping:** Cuando se comenzó el proyecto no se tenía ninguna idea de lo que significaba esta técnica, ni siquiera se imaginaba lo que ésta podía llegar hacer. Durante la etapa inicial de investigación se logró comprender la técnica, lo que permitió enfocar correctamente el proyecto hacia sus objetivos.
- **Continuos cambios de la herramienta:** Que el proceso sea automático, no implica que no haya que estar pendiente. Las mayores dificultades que se encontraron son el cambio de la estructura HTML de las páginas de donde se extrae la información, y el bloqueo de la IP del servidor en algunos sitios webs.

En el caso de que haya algún cambio en el marcado de página, por muy pequeño que sea, con seguridad se tendrá que modificar o rehacer el procedimiento de procesamiento de los datos. En el momento que la herramienta computacional no encuentre la estructura que espera, no será capaz de obtener el resultado final, esto implica que si no se actualiza la herramienta no tiene ningún sentido ocuparla; esto conlleva, a que la persona que ocupe la herramienta siempre tendrá que estar realizando mantención para que funcione correctamente; en cualquier caso, es mejor

que tener que hacer el proceso de extraer las referencias bibliográficas manualmente ya que al cabo de un tiempo esta tarea se hará tediosa y repetitiva.

En el caso del bloqueo de la IP, normalmente esto ocurre por dos razones: el sitio o sitios webs de donde se obtiene la información mediante la técnica de web scraping se ha dado cuenta de ello, y bloquea para evitar realizar la extracción; o bien simplemente porque han detectado un número demasiado elevado de peticiones desde una misma dirección IP, y eso los sitios webs lo consideran un comportamiento sospechoso. La primera solución fue mover el bot a un nuevo servidor, con una dirección IP distinta, pero poco tiempo después se repitió el bloqueo; luego se encontró una solución duradera utilizando proxies. Un proxy, simplificando, es simplemente un servidor que hace de intermediario entre las peticiones de un cliente al servidor de destino. Esto, fundamentalmente, permite que se enmascare la dirección IP original y ésta no pueda ser detectada por el servidor de destino, evitando su bloqueo.

6.2. Problemas durante el proceso de extracción

Como todas las técnicas, el web scraping tiene sus limitaciones y existen algunas tecnologías que limitan o hacen extremadamente complicada su utilización.

- Ajax y Javascript: Causa problemas para los desarrolladores web, ya que permite a los nuevos contenidos ser descargados y presentados en una web sin la necesidad de una página. Esto rompe la relación uno a uno entre URL y contenido, ya que el contenido presentado depende en cierta parte de lo que el usuario hace después de cargar una página lo causa en general no se tenga acceso a ese contenido.

- Problemas con Scopus: Al principio la herramienta desarrollada lograba conectarse con la base de datos y extraía adecuadamente todos los resultados. Al tiempo cuando se volvieron a realizar pruebas la base de datos había cambiado y solo se conseguía conectar con la página de resultados, es decir se podía seguir extrayendo apropiadamente el DOI, pero al momento de conectarse para conseguir los metadatos se originaba un error de que no podía descargar el contenido de la base de datos que se estaba haciendo un requerimiento. El problema de esto se debió a la codificación de los caracteres que existía en la URL para sacar las referencias bibliográficas. Se intentó solucionar con la función `urlencode` que se usa para codificar una cadena para ser usada como la parte de una consulta de una URL pero no sirvió de nada. Lo otro que se intentó fue ir reemplazando cada carácter especial por su código ASCII con la función `str_replace` pero tampoco sirvió. Esto complicó no poder realizar la extracción de las referencias bibliográficas.
- Problemas IEEEExplore: Si se construye la URL con los parámetros de la Figura 52 (b) y luego se ingresa en un navegador, la página arroja “el método de solicitud no es compatible”.

La herramienta computacional no pudo acceder a esta base de datos ya que los parámetros de la URL están en JSON. El problema es que cuando la página que contiene la información está generada dinámicamente con Javascript, este método de web scraping no servirá de mucho debido a que cuando se hace la solicitud a la página, la información aún no está en HTML y en ningún momento se está ejecutando el código Javascript.

6.3. Objetivos alcanzados

El primer objetivo específico “Estudiar la forma en que se realizan las revisiones sistemáticas y mapeos sistemáticos”, se consiguió aprender a cabalidad cada proceso logrando el objetivo.

En el segundo objetivo específico “Revisar Bases de datos científicas”, se analizó en profundidad el contenido HTML de cada base de datos, ya que sin etapa no se podría desarrollar la herramienta.

En el tercer objetivo específico “Estudiar conceptos para el desarrollo”, se estudiaron todos los conceptos relacionados a la herramienta computacional desarrollada, para lograr entender mejor el funcionamiento y con esto alcanzar el objetivo general del proyecto.

El uso de la herramienta agiliza la fase de “Selección de los estudios” de una revisión sistemática, hasta este momento con las pruebas que se han realizado, las búsquedas con la herramienta entregan resultados consistentes y acordes con respecto a las búsquedas manuales, llegando a la conclusión que la herramienta computacional si permite a los investigadores extraer las referencias bibliográficas de las búsquedas que se realicen y guardarlos en un archivo; logrando el último objetivo que es desarrollar la herramienta computacional, esperando colaborar con este elemento en la investigación para facilitar el trabajo de los investigadores.

Como trabajo futuro implica probar la herramienta en más casos de estudio para constatar su efectividad y tener un rango más amplio de resultados del uso de ésta.

Bibliografía

- [1] B. Kitchenham, “Procedures for Performing Systematic Reviews,” 2004.
- [2] B.V. Elsevier, “Science Direct,” 2017. [Online]. Available: <http://www.sciencedirect.com/>.
- [3] R. S. Pressman, *Ingeniería del software. Un enfoque práctico.*, Séptima. McGraw-Hill, 2010.
- [4] A. W. C. Wohlin, P. Runeson, M. Host, M. Ohlsson, B. Regnell, *Experimentation in Software Engineering*, Springer. Boston, 2000.
- [5] S. Fabbri, E. Hernandez, and A. Di Thommazo, “Using Information Visualization and Text Mining to Facilitate the Conduction of Systematic Literature Reviews,” vol. 1, pp. 243–256, 2013.
- [6] D. Young, “Using Bib TEX,” pp. 1–25, 2002.
- [7] J. A. López, “Guía casi completa de BibTeX,” pp. 1–78, 2006.
- [8] J. M. M. Chaus, “Escuela Técnica Superior de Ingeniería WSM : Metodología de Web Scraping para Android y ejemplificación mediante la aplicación UPMDroid,” 2000.
- [9] A. Haque and S. Singh, “Anti-Scraping Application Development Anti-Scraping Application Development,” no. August, 2015.
- [10] J. Adell, “La internet como TELARAÑA: El World- Wide Web.,” pp. 25–32, 2002.
- [11] C. Mateu, *Desarrollo de aplicaciones web*, Primera. 2004.
- [12] C. Hernández, “Aplicación de Técnicas de Web Scraping al Boletín Oficial de Castilla y León.,” Universidad de Valladolid, 2014.

- [13] A. Aulbach, A. Zmievski, and L. Boshell, “Manual de PHP Stig Sæther Bakken,” 2002.
- [14] C. V. Mariño, “Programación en PHP5. Nivel Básico,” 2008.
- [15] “PHP: cURL - Manual,” 2001. [Online]. Available: <http://www.php.net/curl>.
- [16] The World Wide Web Consortium, “Introduction to HTML,” 1999. [Online]. Available: https://www.w3schools.com/html/html_intro.asp.
- [17] R. York, *Beginning CSS: Cascading Style Sheets for Web Design*. 2006.
- [18] The World Wide Web Consortium., “Introduction to CSS,” 1999. [Online]. Available: https://www.w3schools.com/css/css_intro.asp.
- [19] “Epistemonikos,” 2012. [Online]. Available: <http://www.epistemonikos.org/>.
- [20] P. Universidad, “Epistemonikos : la forma más rápida y fiable de encontrar evidencia Epistemonikos : la forma más rápida y fiable de encontrar evidencia,” pp. 2005–2007, 2012.
- [21] M. C. R. Otero, “Guía de uso de mendeley,” 2012.
- [22] L. Carlos, T. Ninahuanca, and C. Torres, “Una breve introducción a JabRef,” pp. 1–10, 2014.
- [23] “StArt Tool - Introduction,” 2013. [Online]. Available: <https://www.youtube.com/watch?v=zCTKl1TBmxU>.

Apéndice A

Estructura HTML de las Páginas Web.

Como ya se ha dicho anteriormente la herramienta se basa en la metodología del web scraping, de manera que para obtener la información relevante para el investigador, se accede a las distintas páginas webs y mediante ésta técnica obtiene los datos solicitados.

A.1. Taylor and Francis

Lo primero para el análisis de la base de datos, es acceder a ésta para visualizar dónde se debe ingresar el string de búsqueda, solicitándole a la página la URL de la consulta.

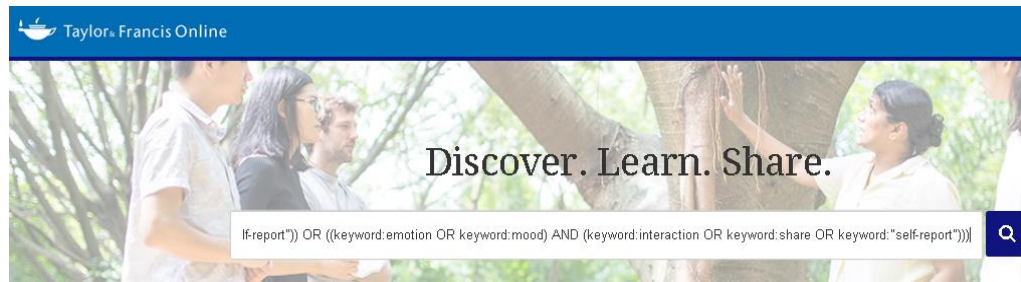
La URL es la siguiente: <http://tandfonline.com/>

Si se accede al link desde el navegador se trasladará a la página de la Figura 45 (a). Posterior a esto se ingresó el string de búsqueda (Tabla 5 “título y keyword”) en el motor de la página, entregando la información que se muestra en la Figura 45 (b), quedando la URL solicitada para esa consulta de la siguiente manera:

[http://tandfonline.com/action/doSearch?AllField=\(\(\(title:emotion+OR+title:mood+\)+AND+\(title:interaction+OR+title:share+OR+title:"self-report"\)\)+OR+\(\(keyword:emotion+OR+keyword:mood\)+AND+\(keyword:interaction+OR+keyword:share+OR+keyword:"self-report"\)\)\)](http://tandfonline.com/action/doSearch?AllField=(((title:emotion+OR+title:mood+)+AND+(title:interaction+OR+title:share+OR+title:)

Si se quiere realizar búsquedas con el filtro fecha, se deberán agregar al final de la URL los siguientes parámetros:

- AfterMonth
- AfterYear
- BeforeMonth
- BeforeYear



(a)

E...	Mé	A...	Do	C... ▲	T...	T...	T...
● 204	POST	<input type="checkbox"/> csi...	<input checked="" type="checkbox"/> csi...	js beac...	gif	—	0 B
● 200	GET	<input type="checkbox"/> do...	<input checked="" type="checkbox"/> ta...	docum...	html	56,86 KB	545,81 ...

Cabeceras	Cookies	Parámetros	Respu
URL solicitada: http://tandfonline.com/action/doSearch?AllField=			
Cabeceras	Cookies	Parámetros	
▼ Filtrar los parámetros de la petición ▼ Cadena consultada AllField: (((title:emotion+OR+title:mood...e+OR+keyword:"self-report")))			

(b)

Figura 45: Página Taylor and Francis.

En la Figura 46 se puede observar la página de resultados de la petición solicitada de la Figura 45.

En la Figura 46 se tiene lo siguiente, en (1) se observan la cantidad de resultados, y en (2) el artículo, se va a requerir saber la estructura HTML de esos dos elementos para poder desarrollar la herramienta.



Figura 46: Página de resultados de Taylor and Francis.

En la Figura 47 se mostrará el código HTML de lo que se va a extraer.

```

▼ <div id="searchResultContainer"
  class="standardSearchWidgetPadding searchResultContainer">
  ev
  ▶ <div class="searchResultDropZoneContainer"> ... </div>
  ▶ <h1 class="search-keyword"> ... </h1>
  ▼ <ul class="num-results">
    ▼ <li class="search-results">
      <strong>1-10</strong>
      of
      <strong>184</strong>
      results
    </li>
  </ul>

```

(a)

```

▼ <ol class="search-results">
  <!-- rendering_time:47-->
  <!-- search_time:7653-->
  <!-- total_search_time:7700-->
  <!-- modified:1513604824000-->
  ▶ <li> ... </li>
  <!-- modified:1513604824000-->
  ▶ <li> ... </li>
  <!-- modified:1513604824000-->
  ▶ <li> ... </li>
  <!-- modified:1513604824000-->
  ▶ <li> ... </li>

```

(b)

```

▼ <li>
  ▼ <article class="searchResultItem" data-title="Alba Emoting and emotional
    melody: surfing the emotional wave in Cachagua, Chile">
    ▼ <div class="art_title"> ev
      <div class="article-type">Article</div> ev
      ▼ <span class="hlFld-Title"> ev
        ▶ <a class="ref nowrap" href="/doi/full/10.1080/19443927.2010.504998">

```

(c)

Figura 47: Extractos de códigos de la Pagina Taylor and Francis.

De la Figura 47 se obtiene lo siguiente:

- (a) Extracto de código de la cantidad de resultados.
- (b) Extracto de código de la lista de resultados.
- (c) Extracto de código de la URL donde se encuentra el DOI.

Para descargar las referencias bibliográficas en formato BibTeX se deberá acceder a las peticiones HTTP que son de tipo POST y observar los parámetros que entrega la página. Con esos parámetros se construye la URL para que entregue la referencia bibliográfica. Los parámetros son los siguientes, como se ve en la Figura 48.

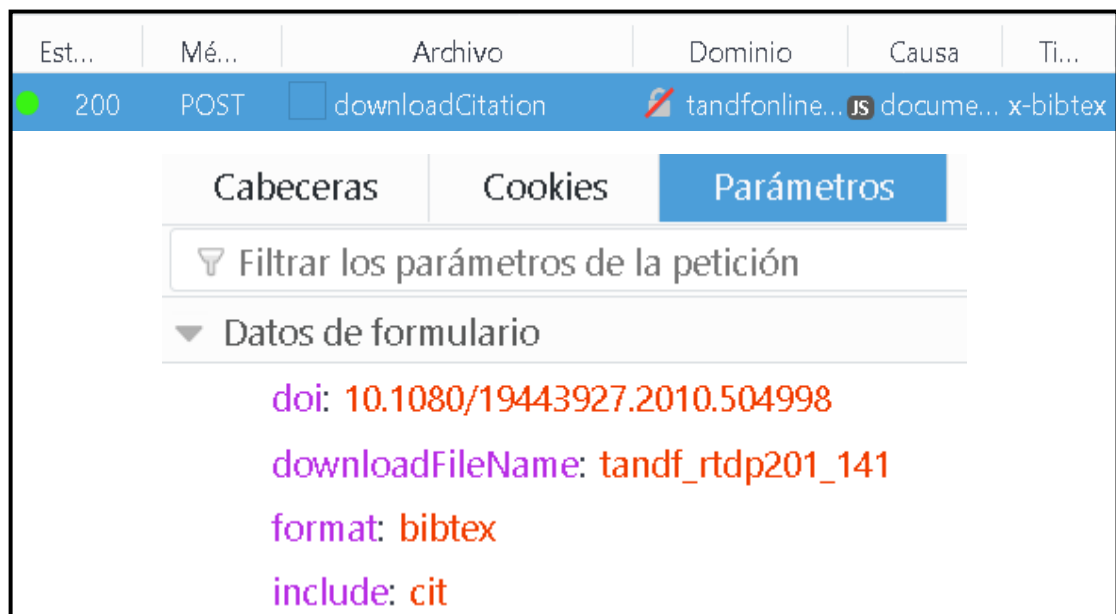


Figura 48: Peticiones HTTP solicitar referencias bibliográficas Taylor and Francis

La URL queda de la siguiente forma:

<http://www.tandfonline.com/action/downloadCitation?doi=10.3402/ejpt.v5.24104&downloadFileName=10.3402/ejpt.v5.24104&format=bibtex&include=cit>

- Subrayado Amarillo es la URL solicitada de la página Taylor and Francis.
- Subrayado en Morado es un parámetro para indicar el nombre del archivo.
- Subrayado en Verde es el parámetro del DOI; este cambia a medida que se selecciona los distintos artículos de la consulta.
- Subrayado en Celeste son parámetros para indicar que se eligió el formato BibTeX y que solo incluye la cita bibliográfica.

B.2. Emerald.

Para el análisis de la base de datos, se accederá a ésta para visualizar dónde se debe ingresar el string de búsqueda, solicitándole a la página la URL de la consulta.

La URL es la siguiente: <http://emeraldinsight.com/search/advanced>.

Si se accede al link desde el navegador, éste llevará a la página de la Figura 49 (a), posterior a esto se ingresó el string de búsqueda como en la Figura 29 en el motor de la página, entregando la siguiente información como se muestra en la Figura 49 (b), quedando la URL solicitada para esa consulta de la siguiente manera:

[http://emeraldinsight.com/action/doSearch?text1=\(emotion+OR+mood\)+AND+\(interaction+OR+share+OR+"self-report"&field1=Title&AfterMonth=2&AfterYear=2015&BeforeMonth=2&BeforeYear=2016](http://emeraldinsight.com/action/doSearch?text1=(emotion+OR+mood)+AND+(interaction+OR+share+OR+)

- Subrayado en amarillo es la URL solicitada.
- Subrayado en verde la cadena de búsqueda que se ingresa para la búsqueda.
- Subrayado celeste es el filtro en donde se va a buscar.
- Subrayado morado es el rango de fechas que se eligió.

Advanced Search

Search for

(emotion OR mood) AND (interaction

Content Item title

Publication date:

- All dates
- Last month
- Last 6 months
- Last year
- Specific date range

From: February 2015

To: February 2016

Search

(a)

Cabeceras	Cookies	Parámetros	
URL solicitada: http://emeraldinsight.com/action/doSearch?			
Cabeceras	Cookies	Parámetros	Respuesta
▼ Filtrar los parámetros de la petición			
▼ Cadena consultada			
text1: (emotion+OR+mood)+AND+(interac...on+OR+share+OR+"self-report")			
field1: Title			
AfterMonth: 2			
AfterYear: 2015			
BeforeMonth: 2			
BeforeYear: 2016			

(b)

Figura 49: Página Emerald.

A continuación se mostrará el código HTML de lo que se va a extraer

```

▼ <div id="searchResultContainer" class="standardSe
  <!--adding search result tabs, if we'collection
  ▶ <div class="search-tabs search-tabs-widget clea
  ▼ <div id="allTabsContainer">
    ▶ <div class="clear">...</div>
    <!--adding result controls-->
    ▼ <div class="searchResultPagination">
      ▼ <div class="stacked paginationStatus">
        <span class="top"></span>
        <span class="pagingVerb">Results:</span>
        1-3 of 3

```

(a)

```

▼ <ol id="searchResultItems" class="searchResultItems ">
  ▶ <li class="search-result-row clear">...</li>
  <!--modified:1509486877000-->
  ▶ <li class="search-result-row clear">...</li>
  <!--modified:1509486877000-->
  ▶ <li class="search-result-row clear">...</li>
  <!--modified:1509486877000-->
  ▶ <li class="search-result-row clear">...</li>

```

(b)

```

▼ <li class="search-result-row clear">
  ▶ <div class="search-result-col1">...</div>
  ▼ <div class="search-result-col2">
    ▼ <div class="articleEntry">
      ▶ <div class="pull-left">...</div> ev
      ▼ <div class="content-container"> ev
        ▼ <span class="art_title">
          ▼ <span class="hlFld-Title">
            ▶ <a class="ref nowrap" href="/doi/full/10.1108
              /IR-08-2015-0164">...</a> ev
          </span>
        </span>
      </span>

```

(c)

Figura 50: Extractos de códigos de página Emerald.

De la Figura 50 se obtiene lo siguiente:

- (a): Extracto de código de la cantidad de resultados.
- (b): Extracto de código de la lista de resultados.
- (c): Extracto de código del DOI.

A continuación, se explicará la forma de descargar las referencias bibliográficas y su respectiva URL.

Luego de realizar la búsqueda en la base de datos se accede a los resultados encontrados. Posterior a eso se tendrá que seleccionar la opción “Download Citation”, en donde se redirecciona a otra página, permitiendo elegir el formato; una vez que se elige el formato, se selecciona el botón “Download article citation data”, para descargar las referencias bibliográficas deberán acceder a las peticiones HTTP que son de tipo POST y observar los parámetros que entrega la página como se muestra en la Figura 51.

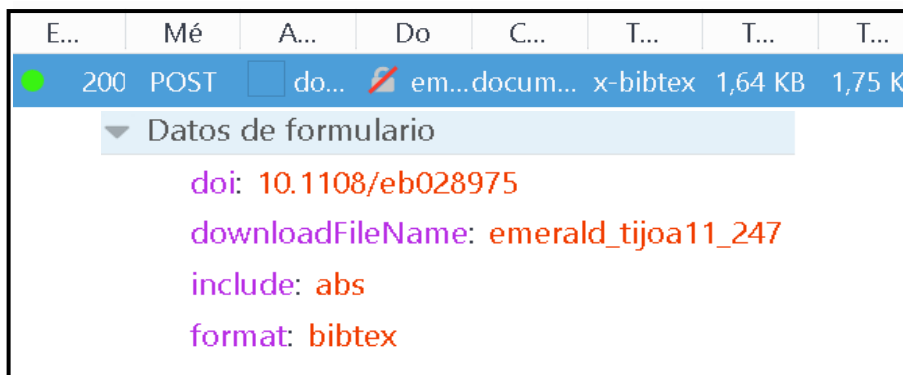


Figura 51: Peticiones HTTP solicitar referencias bibliográficas página Emerald

Con estos parámetros se podrá obtener la URL para que entregue las referencias bibliográficas

http://emeraldinsight.com/action/downloadCitation?doi=10.1108/eb028975&downloadFileName=emerald_tjpa11_247&format=bibtex

- Subrayado celeste es la URL solicitada.
- Subrayado amarillo es el doi del artículo seleccionado.
- Subrayado en verde es el nombre del archivo, puede ser cualquier nombre.
- Subrayado morado es el formato que se escogió.

En la URL no se incluyó el parámetro &include=abs, ya que solo se necesita las citas bibliográficas, y si se incluye ese parámetro entregará el abstract y las citas.

B.3. Wiley & Sons.

Se accederá a la página de “Búsqueda avanzada” para visualizar dónde se debe ingresar el string de búsqueda, solicitando a la página la URL de la consulta.

La URL es la siguiente: <http://onlinelibrary.wiley.com/advanced/search>.

Si se accede al link desde el navegador, éste llevará a la página de la Figura 52 (a). Posterior a esto se ingresó el string de búsqueda como se muestra en la Figura 30 en el motor de la página, entregando los siguientes parámetros como se muestra en la Figura 52 (b), quedando la URL solicitada para esa consulta de la siguiente manera:

[http://onlinelibrary.wiley.com/advanced/search/results/reentry?searchRowCriteria\[0\].queryString=\(emotion+or+mood\)+and+\(interaction+or+share+or+%22self-report%22\)&searchRowCriteria\[0\].fieldName=keyword&dateRange=between&startYear=2013&endYear=2015](http://onlinelibrary.wiley.com/advanced/search/results/reentry?searchRowCriteria[0].queryString=(emotion+or+mood)+and+(interaction+or+share+or+%22self-report%22)&searchRowCriteria[0].fieldName=keyword&dateRange=between&startYear=2013&endYear=2015)

- Subrayado Amarillo: URL solicitada.
- Subrayado celeste: string de búsqueda ingresado.
- Subrayado morado: Filtro campo.
- Subrayado verde: Rango de fechas.

Advanced Search | [Search Tips >](#)

SEARCH FOR

(emotion OR mood) AND (interaction OR share) Keywords

AND All Fields

AND All Fields

[Add another row >](#) [Search](#)

SPECIFY DATE RANGE

All Dates

In the last 6 Months

Between 2013 2015

Please enter 4-digit years.

[Search](#)

(a)

Esta...	Mét...	Archivo	Dominio	Causa	T...	Trans...
302	POST	search	onlinelibrar...	document	html	52,07 KB
200	GET	results?start=1	onlinelibrar...	document	html	52,07 KB
200	GET	dtagent_ICAp_60005002...	onlinelibrar...	script	js	cacheado
200	GET	importAll.css	onlinelibrar...	stylesheet	css	cacheado

Cabeceras Cookies **Parámetros** Respuesta Tiempos

Filtrar los parámetros de la petición

▼ Datos de formulario

searchRowCriteria[0].queryString: (emotion+OR+mood)+AND+(interac...on+OR+shar

searchRowCriteria[0].fieldName: keyword

dateRange: between

startYear: 2013

endYear: 2015

(b)

Figura 52: Página Wiley & Sons.

Si se quiere realizar una búsqueda con más términos relacionándolos con los operadores lógicos, la URL queda de la siguiente manera:

[http://onlinelibrary.wiley.com/advanced/search/results/reentry?searchRowCriteria\[0\].queryString=\(emotion+or+mood\)+and+\(interaction+or+share+or+%22self-report%22\)&searchRowCriteria\[0\].fieldName=keyword&searchRowCriteria\[0\].booleanConnector=or&searchRowCriteria\[1\].queryString=\(emotion+or+mood\)+and+\(interaction+or+share+or+%22self-report%22\)&searchRowCriteria\[1\].fieldName=abstract](http://onlinelibrary.wiley.com/advanced/search/results/reentry?searchRowCriteria[0].queryString=(emotion+or+mood)+and+(interaction+or+share+or+%22self-report%22)&searchRowCriteria[0].fieldName=keyword&searchRowCriteria[0].booleanConnector=or&searchRowCriteria[1].queryString=(emotion+or+mood)+and+(interaction+or+share+or+%22self-report%22)&searchRowCriteria[1].fieldName=abstract)

- Subrayado en amarillo: es la variable para el operador booleano.
- Subrayado Verde: es la variable del campo en donde se quiere buscar, en este caso como se seleccionaron dos campos solo va cambiando lo que esta subrayado en verde.

searchRowCriteria[0].fieldName=keyword

searchRowCriteria[1].fieldName=abstract

```
▼ <div id="searchResults" xmlns="http://www.w3.org/1999/xhtml">
  ▶ <div class="sectionHeading"> ... </div>
  ▼ <div id="resultInfo">
    ▼ <p id="searchedForText">
      There are
      <em>37</em>
      results for:
      ▶ <em> ... </em>
    </p>
```

(a)

```
▼ <div id="searchResultsList">
  ▶ <div class="paginationFilter"> ... </div>
  ▼ <ol class="articles">
    ▶ <li> ... </li>
    ▶ <li> ... </li>
    ▶ <li> ... </li>
    ▶ <li> ... </li>
    ▶ <li> ... </li>
    ▶ <li> ... </li>
```

(b)

```
▼ <li>
  ▼ <div class="access">
    <input id="option1" name="doi" value="10.1111/sode.12126"
    >
  </div>
```

(c)

Figura 53: Extractos de códigos de la página Wiley & Sons.

De la Figura 53 se obtiene lo siguiente:

- (a): Extracto de código de la cantidad de resultados.
- (b): Extracto de código de la lista de resultados.
- (c): Extracto de código del DOI.

A continuación, se explicará la forma de descargar las referencias bibliográficas y su respectiva URL.

Luego de realizar la búsqueda en la base de datos, se accede a los resultados encontrados, posterior a eso se tendrá que seleccionar la opción “View/sabe citation”, en donde se despliega una ventana modal, donde podrán seleccionar el formato. Posterior a esto deberán acceder a las peticiones HTTP como se muestra en la Figura 54.

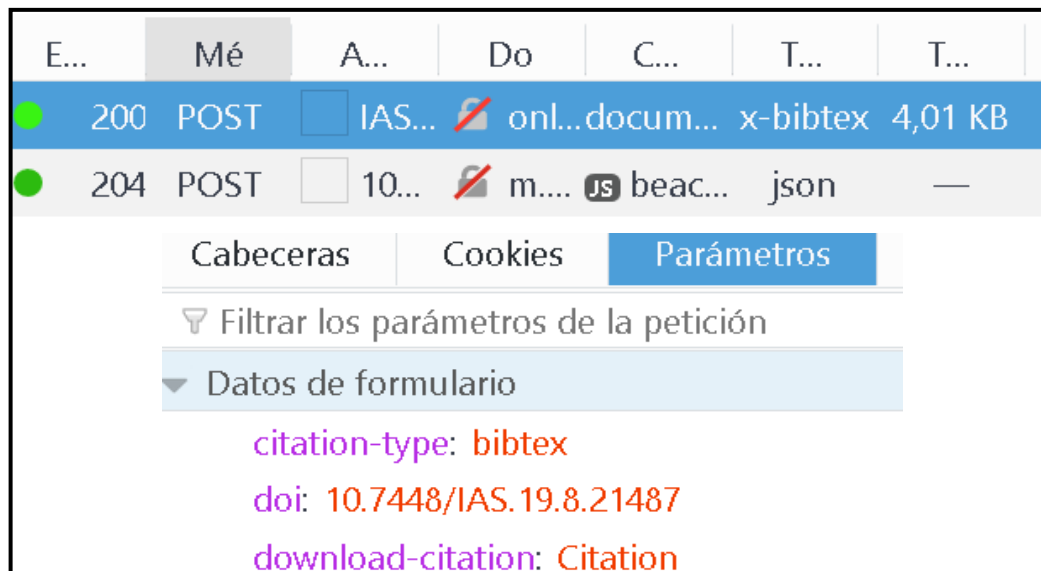


Figura 54: Peticiones HTTP solicitar referencias bibliográficas Wiley & Sons.

Una vez obtenidos los parámetros se podrá construir la URL quedando de la siguiente manera:

onlinelibrary.wiley.com/enhanced/getCitation/doi/10.7448/IAS.19.8.21487?citation-type=bibtex&doi=10.7448/IAS.19.8.21487&download-citation=Citation

Con la URL entregada no se visualizan las referencias bibliográficas, solo se genera un archivo en donde se encuentra la información que se necesita.

B.4. SpringerLink.

Para analizar la base de datos se accederá a ésta, para mostrar dónde se debe ingresar el string de búsqueda y cómo se debe hacer la consulta, y así obtener la URL solicitada.

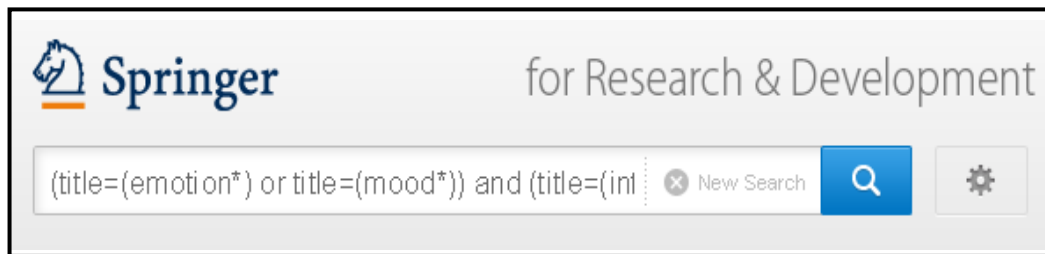
La URL es la siguiente: <http://link.springer.com/>.

Se accederá al link desde el navegador que llevará a la página de la Figura 55 (a). Posterior a esto se ingresó el string de búsqueda (Tabla 7 “título”) en el motor de la página, entregando la información de los parámetros como se muestra en la Figura 55 (b), quedando la URL solicitada para esa consulta de la siguiente manera:

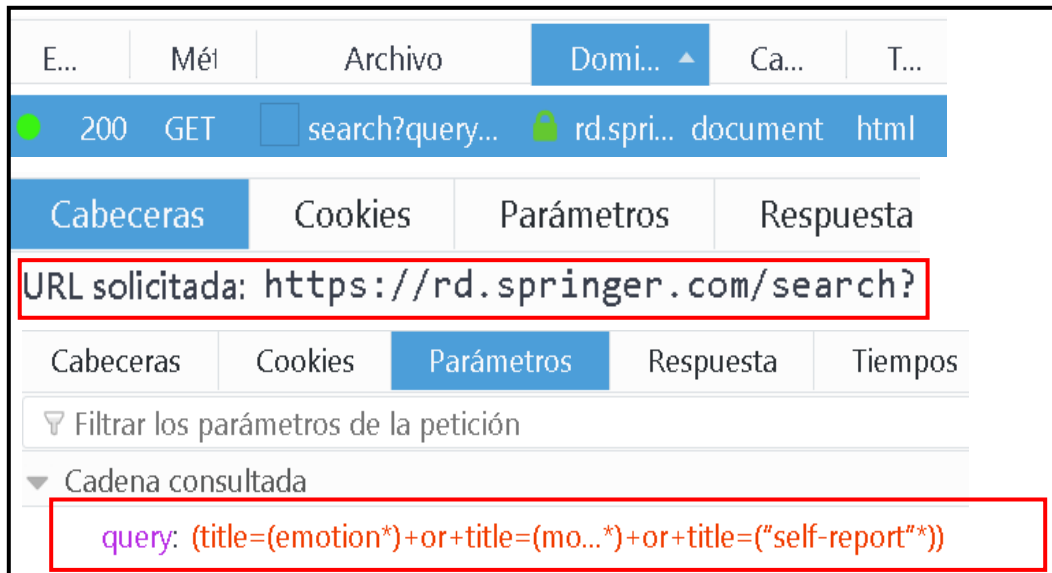
[https://rd.springer.com/search?query=\(title=\(emotion*\)+or+title=\(mood*\)\)+and+\(title=\(interaction*\)+or+title=\(share*\)+or+title=\(“self-report”*\)\)](https://rd.springer.com/search?query=(title=(emotion*)+or+title=(mood*))+and+(title=(interaction*)+or+title=(share*)+or+title=(“self-report”*)))

Los siguientes parámetros por si se quiere realizar el filtro fecha, solo se debe agregar a la URL concatenándolas con “&”:

- date-facet-year=between
- facet-start-year
- facet-end-year



(a)



(b)

Figura 55: Página de SpringerLink.

A continuación, se mostrará la estructura HTML de la página SpringerLink.

```
▼ <div id="kb-nav--main" class="col-main" role="main">
  ▼ <div class="header">
    ▼ <h1 class="number-of-search-results-and-search-terms">
      <strong>62</strong>
      Result(s) for
      ▶ <strong>...</strong>
```

(a)

```
▼ <ol id="results-list" class="content-item-list">
  ▶ <li class="no-access">...</li>
  ▶ <li>...</li>
  ▶ <li class="no-access">...</li>
  ▶ <li class="no-access">...</li>
  ▶ <li class="no-access">...</li>
  ▶ <li class="no-access">...</li>
  ▶ <li class="no-access">...</li>
  ▶ <li class="no-access">...</li>
```

(b)

```
▼ <li>
  ::before
  ▶ <div class="lozenges">...</div>
  <p class="content-type">Article</p>
  ▼ <h2>
    ▶ <a class="title" href="/article/10.1007/s13218-015-0394-8">...</a>
  </h2>
```

(c)

Figura 56: Extractos de códigos de la página SpringerLink.

De la Figura 56 se obtiene lo siguiente:

- (a): Extracto de código de la cantidad de resultados.
- (b): Extracto de código de la lista de resultados.
- (c): Extracto de código del DOI.

Luego de realizar la búsqueda en la base de datos, se accede a los resultados encontrados, posterior a eso se tendrá que seleccionar la opción “Cite paper”, mostrando el formato de la extracción de las referencias bibliográficas. Para conocer la URL se accederán a los parámetros como se muestra en la Figura 57.

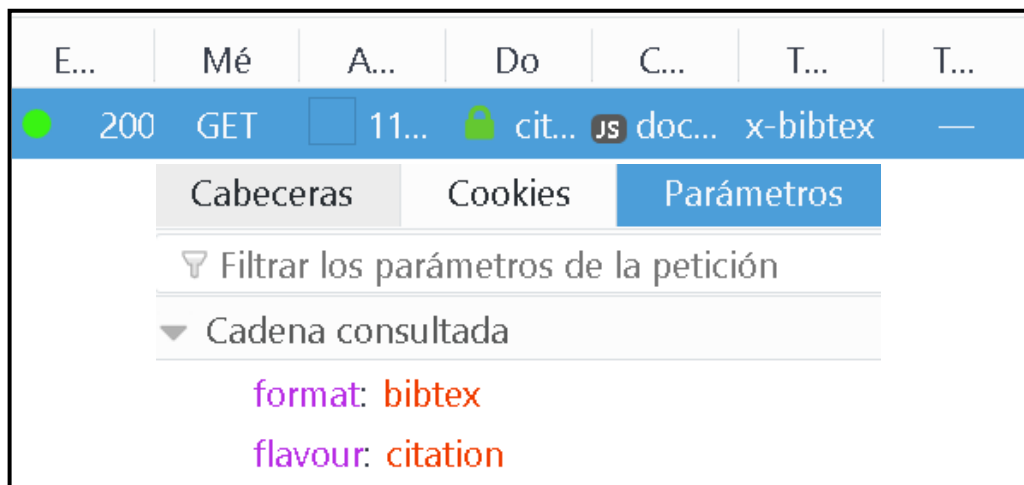


Figura 57: Peticiones HTTP solicitar referencias bibliográficas SpringerLink.

La URL queda de la siguiente manera en base a los parámetros de la URL solicitada

https://citationneeded.springer.com/v2/references/10.1007/11881070_40?format=bibtex&flavour=citation

- Subrayado amarillo es la URL solicitada. Lo único que cambia es el DOI del resultado seleccionado.
- Subrayado Verde es el formato elegido con la cita bibliográfica.

B.5. IEEEXplore.

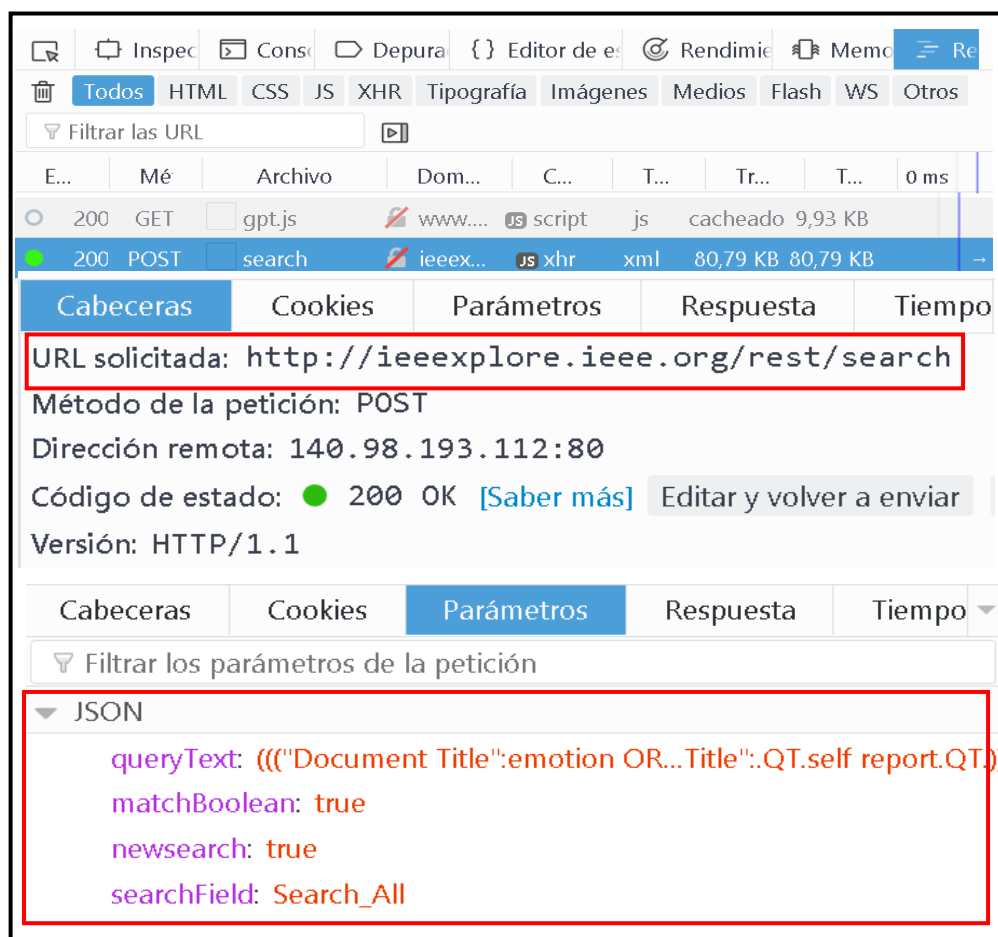
Para acceder a la página IEEEXplore”Búsqueda avanzada”, se debe ir a la siguiente URL: <http://ieeexplore.ieee.org/search/advsearch.jsp?expression-builder>

Se accederá al link desde el navegador que llevará a la página de la Figura 58 (a). Posterior a esto se ingresa el string de búsqueda (Tabla 13 “título”) en el motor de la página, entregando la información de los parámetros como se muestra en la Figura 58 (b).

Al realizar esta búsqueda se puede observar que el método de petición que se realizó al servidor es de tipo POST. El sitio realiza una llamada a una URL en donde los parámetros de la URL están en JSON.



(a)



(b)

Figura 58: Página IEEEXplore.

A continuación se mostrará la estructura HTML de la página.

```

▼ <div class="pure-u-1-1 Dashboard-header ng-scope"
  <!--ngIf: !vm.totalRecords-->
  <!--ngIf: vm.totalRecords === 1-->
  <!--ngIf: vm.totalRecords > 1-->
  <span class="ng-binding ng-scope" ng-if="vm.tota
  Displaying results 1-25 of 96</span>

```

(a)

```

▼ <div ng-transclude="">
  <!--ngRepeat: record in vm.records track by record.articleNumber-->
  ▶ <div class="List-results-items ng-scope" ng-repeat="record in vm.records
  by record.articleNumber" style="">...</div>
  <!--end ngRepeat: record in vm.records track by record.articleNumber-->
  ▶ <div class="List-results-items ng-scope" ng-repeat="record in vm.records
  by record.articleNumber">...</div>
  <!--end ngRepeat: record in vm.records track by record.articleNumber-->
  ▶ <div class="List-results-items ng-scope" ng-repeat="record in vm.records
  by record.articleNumber">...</div>
  <!--end ngRepeat: record in vm.records track by record.articleNumber-->
  ▶ <div class="List-results-items ng-scope" ng-repeat="record in vm.records
  by record.articleNumber">...</div>
  <!--end ngRepeat: record in vm.records track by record.articleNumber-->
  ▶ <div class="List-results-items ng-scope" ng-repeat="record in vm.records
  by record.articleNumber">...</div>

```

(b)

```

▼ <div class="List-results-items ng-scope" ng-repeat="record in vm.records
by record.articleNumber" style="">
  ▼ <xpl-result class="ng-isolate-scope" record="::record"
  records="::vm.data.records" user="::vm.user" selected-
  items="vm.selectedItems" bulk-selected-items="vm.bulkSelectedItems">
  ▼ <div class="pure-g">
    ▶ <div class="pure-u-1-24">...</div>
    ▼ <div class="pure-u-22-24" xpl-displayer="">
      ▼ <h2>
        <!--ngIf: ::record.ephemera-->
        <!--ngIf: ::(!record.ephemera)-->
        ▶ <a class="ng-binding ng-scope" ng-if="::(!record.ephemera)"
        href="/document/7376694/" xpl-highlight="" ng-bind-
        html="::record.title.replace('Redline','<font color=\red
        \>Redline</font>')" mathjax-bind="">...</a> ev

```

(c)

Figura 59: Extractos de códigos de la página IEEEExplore.

De la Figura 59 se obtiene lo siguiente:

(a): Extracto de código de la cantidad de resultados.

(b): Extracto de código de la lista de resultados.

(c): Extracto de código del DOI.

Para la obtención de las referencias bibliográficas de los resultados encontrados, se deberá seleccionar la opción “Download Citations”, se despliega una ventana modal en el que se muestra los formatos para elegir. Una vez que se selecciona el botón “Download”, se verá la petición de la URL solicitada y sus parámetros como se muestra en la siguiente Figura 60.

E...	Mé	A...	Do	C...	T...	T...
○	200	GET	<input type="checkbox"/>	styl... sta... JS style...	css	cachea... 4
●	200	GET	<input type="checkbox"/>	cm... da... JS img	gif	43 B
●	200	POST	<input type="checkbox"/>	do... iee... JS xhr	html	1,17 KB
●	200	GET	<input type="checkbox"/>	cm... da... JS img	gif	43 B

Cabeceras Cookies **Parámetros**

▼ Filtrar los parámetros de la petición

▼ Datos de formulario

citations-format: citation-only

download-format: download-bibtex

recordIds: 7376694

Figura 60: Peticiones HTTP solicitar referencias bibliográficas IEEEExplore

La URL queda de la siguiente manera:

<http://ieeexplore.ieee.org/xpl/downloadCitations?citations-format=citation-only&download-format=download-bibtex&recordIds=7325170>

- Subrayado en verde es la URL solicitada.
- Subrayado celeste son los parámetros que indican el tipo de formato y que incluye solo las citas.
- Subrayado amarillo es el DOI del artículo seleccionado.

B.6. Scopus.

Se accederá a la página de Scopus para visualizar dónde se debe ingresar la consulta, solicitándole a la página la URL de dicha consulta.

Por defecto, esta búsqueda proporciona una sola casilla Figura 61 (1) y la posibilidad de elegir en que campo realizar la consulta. Si se quiere combinar varios campos, se puede añadir tantos criterios de búsqueda como sean necesarios a través del botón añadir campo Figura 61 (2).

Previamente, se puede limitar la búsqueda Figura 61 (3) por rango de fechas tipo de documento o área científica como se muestra en la siguiente Figura 57.

Al realizar esta búsqueda, la página se redirecciona a otra página con la siguiente URL.

[https://www.scopus.com/results/results.uri?sot=b&sdt=b&xs=TITLE-ABS-KEY\("Cyberbullying"\)&cluster=scopu](https://www.scopus.com/results/results.uri?sot=b&sdt=b&xs=TITLE-ABS-KEY()

- Subrayado Amarillo es la URL principal que no cambia.
- Subrayado en verde es el parámetro de los campos, va cambiando dependiendo en que campo se quiere buscar y lo que está dentro de los paréntesis es la cadena de búsqueda.
- Subrayado Morado es el parámetro para la fecha.

The screenshot displays the Scopus search interface with the following elements:

- Navigation tabs: Documents (highlighted), Authors, Affiliations, Advanced.
- Search bar: Contains the text "Cyberbullying" (marked with a red '1') and a search icon.
- Search criteria: "Article title, Abstract, Keywords" (marked with a red '2') and a plus sign icon.
- Example text: "E.g., 'heart attack' AND stress".
- Limit section: "Limit" (marked with a red '3') with a dropdown arrow.
- Date range section: "Date range (inclusive)" with radio buttons for "Published" (selected) and "Added to Scopus in the last". The "Published" option has "All years" and "Present" (both with dropdown arrows) as date ranges. The "Added to Scopus in the last" option has "7 days" (with a dropdown arrow) as a date range.
- Document type section: "Document type" with "ALL" (with a dropdown arrow) as the selected option.
- Buttons: "Reset form" and "Search Q" (with a magnifying glass icon).

Figura 61: Página de Scopus.

A continuación se mostrará la estructura HTML de la página scopus

```
▼ <table id="srchResultsList" class="table">
  ▶ <thead> ... </thead>
  ▼ <tbody>
    ▶ <tr id="resultDataRow0" class="searchArea"> ... </tr>
    ▶ <tr id="resultLinkRow0" class="" data-sourcecode="p" data-citationcode="cp"> ... </tr>
    ▶ <tr id="previewabstractrow1" class="panel-collapse collapse displayNone"> ... </tr> ev
    ▶ <tr id="resultDataRow1" class="searchArea"> ... </tr>
    ▶ <tr id="resultLinkRow1" class="" data-sourcecode="j" data-citationcode="ar"> ... </tr>
    ▶ <tr id="previewabstractrow2" class="panel-collapse collapse displayNone"> ... </tr> ev
    ▶ <tr id="resultDataRow2" class="searchArea"> ... </tr>
    ▶ <tr id="resultLinkRow2" class="" data-sourcecode="j" data-citationcode="ar"> ... </tr>
    ▶ <tr id="previewabstractrow3" class="panel-collapse collapse displayNone"> ... </tr> ev
    ▶ <tr id="resultDataRow3" class="searchArea"> ... </tr>
    ▶ <tr id="resultLinkRow3" class="" data-sourcecode="b" data-citationcode="ch"> ... </tr>
    ▶ <tr id="previewabstractrow4" class="panel-collapse collapse displayNone"> ... </tr> ev
  </tbody>
```

(a)

```
▼ <tr id="resultDataRow0" class="searchArea">
  ▼ <th scope="row" rowspan="2">
    ▼ <div class="checkbox">
      <input id="eid_2-s2.0-84991086724" name="selectedEIDs" onclick=
      s2.0-84991086724" type="checkbox"> ev
```

(b)

Figura 62: Extracto de código de la página Scopus.

De la Figura 62 se obtiene lo siguiente:

(a): Extracto de código de la lista de resultados.

(b): Extracto de código del DOI.

B.7. Science Direct.

Lo primero para el análisis de la base de datos, se accederá a ésta para visualizar donde se debe ingresar el string de búsqueda, solicitándole a la página la URL de la consulta.

La URL es la siguiente: <http://www.sciencedirect.com/science/search>

Si se accede al link desde el navegador se carga la página de la Figura 63 (a). Posterior a esto se ingresa el string de búsqueda como se muestra en la Figura 31 en el motor de la página, entregando la siguiente información como se muestra en la Figura 63 (b), quedando la URL solicitada para esa consulta de la siguiente manera:

[http://www.sciencedirect.com/science?_ob=MiamiSearchURL&_method=submitForm&_acct=C000228598&_temp=all_search.tpl&md5=bc6a8104fc3e2467a87cd5c88f8d365&test_alid=&SearchText=\(emotion+OR+mood\)+AND+\(interaction+OR+share+OR+`self-report`\)&keywordOpt=TITLE&addTerm=0&addSearchText=&addkeywordOpt=ALL&source=srcJrl&source=srcBk&srcSel=1&DateOpt=2&fromDate=2007&toDate=Present&RegularSearch=Search](http://www.sciencedirect.com/science?_ob=MiamiSearchURL&_method=submitForm&_acct=C000228598&_temp=all_search.tpl&md5=bc6a8104fc3e2467a87cd5c88f8d365&test_alid=&SearchText=(emotion+OR+mood)+AND+(interaction+OR+share+OR+`self-report`)&keywordOpt=TITLE&addTerm=0&addSearchText=&addkeywordOpt=ALL&source=srcJrl&source=srcBk&srcSel=1&DateOpt=2&fromDate=2007&toDate=Present&RegularSearch=Search)

- Subrayado morado es un parámetro que se genera cada vez que se hace una consulta.
- Subrayado Verde es la cadena de búsqueda ingresada.
- Subrayado amarillo es el filtro campo en este caso para el título.
- Subrayado celeste es el rango de fechas.

Search for

(emotion OR mood) AND (interaction OR share OR "self-report") in Title

AND

in All Fields

Refine your search

Journals Open Access articles only

Books

- All Sciences -

- Agricultural and Biological Sciences
- Arts and Humanities
- Biochemistry, Genetics and Molecular Biology

Hold down the Ctrl key (or Apple Key) to select multiple entries.

All Years 2007 to: Present

Search

(a)

Cabeceras	Cookies	Parámetros	Cabeceras	Cookies	Parámetros
▼ Filtrar los parámetros de la petición ▼ Cadena consultada _ob: MiamiSearchURL _method: submitForm _acct: C000228598 _temp: all_search.tpl md5: bcb6a8104fc3e2467a87cd5c88f8d365 test_alid:			? Filtrar los parámetros de la petición SearchText: (emotion+OR+mood)+AND+(inte keywordOpt: TITLE addTerm: 0 addSearchText: addkeywordOpt: ALL source: [...] 0: srcJrl 1: srcBk srcSel: 1 DateOpt: 2 fromDate: 2007 toDate: Present RegularSearch: Search		

(b)

Figura 63: Página Science direct

```
▼ <div id="bodyResults" class="cbodyAllResults">
  ▼ <div class="searchPg" aria-label="Science Direct" role="Main">
    ▼ <div class="amtResults refineAvail adAvail skyScraper">
      ▼ <div class="iconLinks rssSearchBlk">
        ▼ <div style="float:left;">
          ▼ <h1 class="queryText">
            <strong>Search results: 264</strong>
```

(a)

```
▼ <ol class="articleList results">
  ▶ <li class="detail">...</li>
  <div id="resultsView_1" class="resultsView"></div>
  ▶ <li class="detail">...</li>
  <div id="resultsView_2" class="resultsView"></div>
  ▶ <li class="detail">...</li>
  <div id="resultsView_3" class="resultsView"></div>
  ▶ <li class="detail">...</li>
  <div id="resultsView_4" class="resultsView"></div>
  ▶ <li class="detail">...</li>
```

(b)

```
▼ <h2>
  ▶ <a id="title_S030645731500103X" class="cLink artTitle
  S_C_artTitle " href="/science/article/pii/S030645731500103X"
  data-docsubtype="fla" querystr="?_rdoc=1&fmt=high&
  _origin=search&docanchor=&_ct=264&_zone=rslt_list_item&
  md5=384249ad889fcb60c497fc0cc5e8e6a3">...</a>
</h2>
```

(c)

Figura 64: Extracto de código de Science Direct.

De la Figura 64 se obtiene lo siguiente:

(a): Extracto de código de la cantidad de resultados.

(b): Extracto de código de la lista de resultados.

(c): Extracto de código del DOI.

Luego de realizar la búsqueda en la base de datos, se accede a los resultados encontrados. Posterior a eso se tendrá que seleccionar la opción “Export”, mostrando el formato de la extracción de las referencias bibliográficas. Para conocer la URL se accederán a los parámetros como se muestra en la Figura 65.

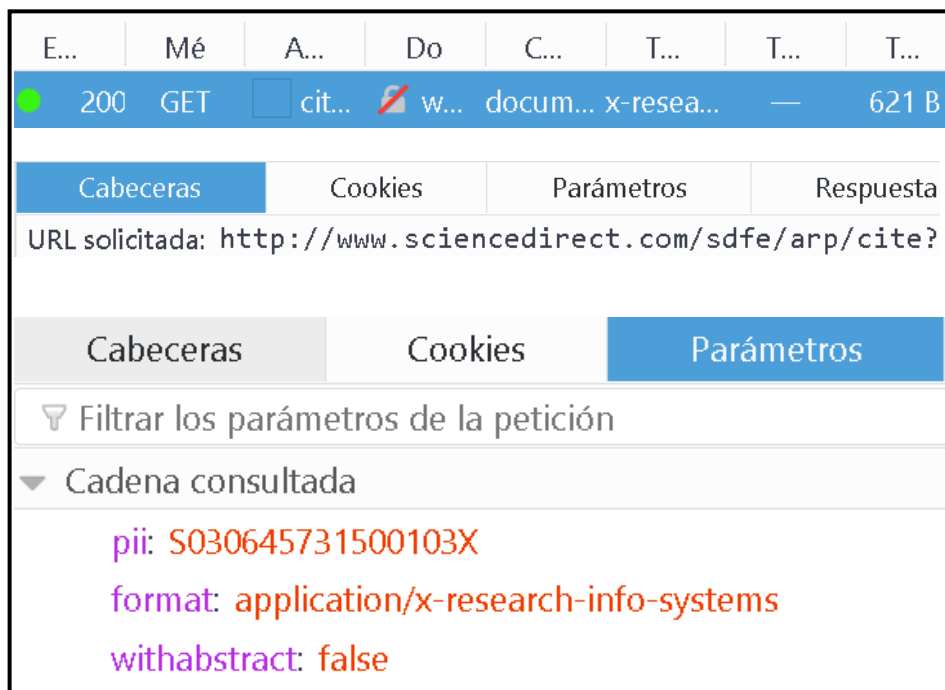


Figura 65: Peticiones HTTP solicitar referencias bibliográficas Science Direct.

Quedando la URL de la siguiente manera.

www.sciencedirect.com/sdfe/arp/cite?pii=S030645731500103X&format=application/x-research-info-systems&withabstract=false

- Subrayado amarillo es la url solicitada.
- Subrayado verde es el DOI del artículo.
- Subrayado verde parámetro del BibTeX.
- Subrayado morado indica que no incluye el abstract en las citas.

Apéndice B

Especificación de requisitos

B.1. Descripción general

La especificación de requisitos del sistema “HERRAMIENTA COMPUTACIONAL DE APOYO A LA ETAPA DE BÚSQUEDA DE ESTUDIOS PRIMARIOS EN LAS BASES DE DATOS CIENTÍFICAS”, pretende describir de manera general el comportamiento del sistema anteriormente indicado. Es importante mencionar que la aplicación tendrá interacción directa con las bases de datos, esto quiere decir que la información extraída proviene de las fuentes definidas por el usuario.

B.2. Actores de la Herramienta

Antes de definir los requisitos que debe satisfacer la herramienta, es necesario determinar qué actores se ven involucrados con el sistema y de qué forma se relacionan con él, para así tener mayor claridad a la hora de definir y priorizar requisitos. Los actores que se ven involucrados en este sistema son:

- Usuario: Este se relaciona con el sistema a través de la definición de las bases de datos con las que se trabajó y la información que se debe extraer.
- Bases de datos: Aquel sitio web que es parseado por la herramienta computacional para la recuperación de los datos que desea el usuario.
- Usuario final: Es quien tendrá interacción con la herramienta computacional, y además la visualización de la información recuperada por el sistema.

B.3. Requisitos del usuario

Estos requisitos buscan caracterizar la interacción entre el usuario y la herramienta computacional, y además, la manera en que la herramienta recibe los datos y de qué forma la información extraída es presentada al usuario. Los principales requisitos que caen dentro de esta categoría son:

- La información extraída de las bases de datos debe ser presentada de manera visual para que permita que el usuario final sea capaz de interpretarla de manera simple e intuitiva.
- El usuario debe ser capaz de definir las fuentes sobre las cuales se realizará la extracción de los datos y la disponibilidad de ellas.
- Para el usuario debe ser transparente el trabajo de la herramienta, si desea puede interactuar con ella a medida que se va desarrollando.

B.4. Requisitos del sistema

- La herramienta computacional deberá mostrar diferentes datos recopilados de varias páginas web, utilizando para ellos técnicas de web scraping.
- La herramienta deberá mostrar la recopilación de datos en un archivo creado dinámicamente en tiempo de ejecución.
- Cada bloque de datos debe ser independiente del resto; esto implica que si un bloque de alguna página web no está disponible, por cualquier motivo externo a la herramienta computacional, el resto de los bloques deberán seguir funcionando correctamente.

B.5. Requisitos no funcionales

- **Tiempo de procesamiento:** El sistema debe procesar la información que este recibe dentro de un periodo de tiempo menor a una extracción de registros manuales, el periodo de tiempo que se demora la herramienta computacional está cuantificado en las secciones 5.1 a 5.7.
- **Orientación a objetos:** El código que sustenta la plataforma debe ser programado utilizando orientación a objetos con el fin de permitir la extensibilidad del sistema una vez que la versión inicial de este haya sido desarrollada.
- **Conexión a Internet:** Para que el usuario pueda ocupar el software debe estar conectar a internet.

R1 Consultar por artículos en ACM: Esta función permite el ingreso del string de búsqueda, que introduzca dicho string en el motor de búsqueda de la herramienta siendo ingresado en la base de datos ACM Digital Library y que recopile los metadatos de la búsqueda.

Nombre	R1.- Consultar por artículos en ACM.
Entradas	bases_datos, búsqueda_por_campos, rango_fechas, string_búsqueda.
Proceso	<p>R1.1: Para el campo bases_datos se deberá seleccionar el campo con el valor ACM.</p> <p>R1.2: Para el campo búsqueda_por_campos se deberá seleccionar al menos uno de los siguientes valores existentes en la herramienta, éstos son los siguientes: título, abstract y keyword.</p> <p>R1.3: Para el campo rango_fechas, el usuario deberá seleccionar los 4 campos de fechas existente en la herramienta.</p> <p style="padding-left: 40px;">R1.3.1: Para el primer campo desde_mes el usuario deberá seleccionar un mes de inicio para realizar la búsqueda, los valores van desde enero hasta diciembre.</p> <p style="padding-left: 40px;">R1.3.2: Para el segundo campo desde_año el usuario deberá seleccionar un año de inicio para realizar la búsqueda, los valores van desde 1999 hasta 2017.</p> <p style="padding-left: 40px;">R1.3.3: Para el primer campo hasta_mes el usuario deberá seleccionar un mes de fin para realizar la búsqueda, los valores van desde enero hasta diciembre</p> <p style="padding-left: 40px;">R1.3.4: Para el primer campo hasta_año el usuario deberá seleccionar un año de fin para realizar la búsqueda, los valores van desde 1999 hasta 2017.</p> <p>R1.4: Para el campo string_búsqueda será una cadena de tamaño máximo 300.</p>

	<p>R1.4.1: El usuario deberá ingresar el <code>string_busqueda</code> en minúscula.</p> <p>R1.4.2: Si <code>string_busqueda</code> es una frase se deberá ingresar entre comillas. R1.4.3: Si <code>string_busqueda</code> contiene el operador lógico OR, las frases o palabras tendrán que ir entre paréntesis. Por ejemplo: (“systematic review” or “mapping study” or survey).</p> <p>R1.5: <code>string_busqueda</code> puede contener tanto números como letras, no están permitidos los caracteres especiales, solo “” * & ?.</p> <p>R1.6: Para realizar la búsqueda los requisitos R1.1, R1.2 y R1.4 son obligatorios y el requisito R1.3 es opcional.</p>
Salidas	<p>Listado de metadatos en ACM. La información de los metadatos es dependiendo del tipo de registro</p> <ul style="list-style-type: none"> ➤ Si el tipo de registro es Artículo los metadatos son los siguientes: <i>author, title, journal, year</i> y DOI. ➤ Si el tipo de registro es Incollection los metadatos son los siguientes: <i>author o editor, title, Publisher, year</i> y DOI. ➤ Si el tipo de registro es Artículo los metadatos son los siguientes: <i>author, title, booktitle, Publisher, year</i> y DOI. ➤ Si el tipo de registro es Inproceedings los metadatos son los siguientes: <i>author, title, booktitle, year</i> y DOI.

R2 Consultar por artículos en Wiley & Sons: Esta función permite el ingreso del string de búsqueda, que introduzca dicho string en el motor de búsqueda de la herramienta siendo ingresado en la base de datos Wiley & Sons y que recopile los metadatos de la búsqueda.

Nombre	R2.- Consultar por artículos en Wiley & Sons.
Entradas	bases_datos, búsqueda_por_campos, rango_fechas, string_búsqueda.
Proceso	<p>R2.1: Para el campo bases_datos se deberá seleccionar el campo con el valor Wiley & Sons.</p> <p>R2.2: Para el campo búsqueda_por_campos se deberá seleccionar al menos uno de los siguientes valores existentes en la herramienta, éstos son los siguientes: título, abstract y keyword.</p> <p>R2.3: Para el campo rango_fechas, el usuario deberá seleccionar los 4 campos de fechas existente en la herramienta.</p> <p style="padding-left: 40px;">R2.3.1: Para el primer campo desde_mes el usuario deberá seleccionar un mes de inicio para realizar la búsqueda, los valores van desde enero hasta diciembre.</p> <p style="padding-left: 40px;">R2.3.2: Para el segundo campo desde_año el usuario deberá seleccionar un año de inicio para realizar la búsqueda, los valores van desde 1999 hasta 2017.</p> <p style="padding-left: 40px;">R2.3.3: Para el primer campo hasta_mes el usuario deberá seleccionar un mes de fin para realizar la búsqueda, los valores van desde enero hasta diciembre</p> <p style="padding-left: 40px;">R2.3.4: Para el primer campo hasta_año el usuario deberá seleccionar un año de fin para realizar la búsqueda, los valores van desde 1999 hasta 2017.</p> <p>R2.4: Para el campo string_búsqueda será una cadena de tamaño máximo 300.</p>

	<p>R2.4.1: El usuario deberá ingresar el <code>string_busqueda</code> en minúscula.</p> <p>R2.4.2: Si <code>string_busqueda</code> es una frase se deberá ingresar entre comillas.</p> <p>R2.4.3: Si <code>string_busqueda</code> contiene el operador lógico OR, las frases o palabras tendrán que ir entre paréntesis. Por ejemplo: (“systematic review” or “mapping study” or survey).</p> <p>R2.5: <code>string_busqueda</code> puede contener tanto números como letras, no están permitidos los caracteres especiales, solo “” * & ?.</p> <p>R2.6: Para realizar la búsqueda los requisitos R2.1, R2.2 y R2.4 son obligatorios y el requisito R2.3 es opcional.</p>
Salidas	<p>Listado de metadatos en Wiley & Sons. La información de los metadatos es dependiendo del tipo de registro</p> <ul style="list-style-type: none"> ➤ Si el tipo de registro es Artículo los metadatos son los siguientes: <i>author, title, journal, year</i> y <i>DOI</i>. ➤ Si el tipo de registro es Incollection los metadatos son los siguientes: <i>author o editor, title, Publisher, year</i> y <i>DOI</i>. ➤ Si el tipo de registro es Artículo los metadatos son los siguientes: <i>author, title, booktitle, Publisher, year</i> y <i>DOI</i>. ➤ Si el tipo de registro es Inproceedings los metadatos son los siguientes: <i>author, title, booktitle, year</i> y <i>DOI</i>.

R3 Consultar por artículos en Springer: Esta función permite el ingreso del string de búsqueda, que introduzca dicho string en el motor de búsqueda de la herramienta siendo ingresado en la base de datos Springer y que recopile los metadatos de la búsqueda.

Nombre	R3.- Consultar por artículos en Springer.
Entradas	bases_datos, búsqueda_por_campos, rango_fechas, string_búsqueda.
Proceso	<p>R3.1: Para el campo bases_datos se deberá seleccionar el campo con el valor Springer.</p> <p>R3.2: Para el campo búsqueda_por_campos se deberá seleccionar al menos uno de los siguientes valores existentes en la herramienta, éstos son los siguientes: título, abstract y keyword.</p> <p>R3.3: Para el campo rango_fechas, el usuario deberá seleccionar los 4 campos de fechas existente en la herramienta.</p> <p style="padding-left: 40px;">R3.3.1: Para el primer campo desde_mes el usuario deberá seleccionar un mes de inicio para realizar la búsqueda, los valores van desde enero hasta diciembre.</p> <p style="padding-left: 40px;">R3.3.2: Para el segundo campo desde_año el usuario deberá seleccionar un año de inicio para realizar la búsqueda, los valores van desde 1999 hasta 2017.</p> <p style="padding-left: 40px;">R3.3.3: Para el primer campo hasta_mes el usuario deberá seleccionar un mes de fin para realizar la búsqueda, los valores van desde enero hasta diciembre</p> <p style="padding-left: 40px;">R3.3.4: Para el primer campo hasta_año el usuario deberá seleccionar un año de fin para realizar la búsqueda, los valores van desde 1999 hasta 2017.</p> <p>R3.4: Para el campo string_búsqueda será una cadena de tamaño máximo 300.</p>

	<p>R3.4.1: El usuario deberá ingresar el <code>string_busqueda</code> en minúscula.</p> <p>R3.4.2: Si <code>string_busqueda</code> es una frase se deberá ingresar entre comillas.</p> <p>R3.4.3: Si <code>string_busqueda</code> contiene el operador lógico OR, las frases o palabras tendrán que ir entre paréntesis. Por ejemplo: (“systematic review” or “mapping study” or survey).</p> <p>R3.5: <code>string_busqueda</code> puede contener tanto números como letras, no están permitidos los caracteres especiales, solo “” * & ?.</p> <p>R3.6: Para realizar la búsqueda los requisitos R3.1, R3.2 y R3.4 son obligatorios y el requisito R3.3 es opcional.</p>
Salidas	<p>Listado de metadatos en Springer de los artículos. La información de los metadatos es dependiendo del tipo de registro</p> <ul style="list-style-type: none"> ➤ Si el tipo de registro es Artículo los metadatos son los siguientes: <i>author, title, journal, year</i> y <i>DOI</i>. ➤ Si el tipo de registro es Incollection los metadatos son los siguientes: <i>author o editor, title, Publisher, year</i> y <i>DOI</i>. ➤ Si el tipo de registro es Artículo los metadatos son los siguientes: <i>author, title, booktitle, Publisher, year</i> y <i>DOI</i>. ➤ Si el tipo de registro es Inproceedings los metadatos son los siguientes: <i>author, title, booktitle, year</i> y <i>DOI</i>.

R4 Consultar por artículos en Emerald: Esta función permite el ingreso del string de búsqueda, que introduzca dicho string en el motor de búsqueda de la herramienta siendo ingresado en la base de datos Emerald y que recopile los metadatos de la búsqueda.

Nombre	R4.- Consultar por artículos en Emerald.
Entradas	bases_datos, búsqueda_por_campos, rango_fechas, string_búsqueda.
Proceso	<p>R4.1: Para el campo bases_datos se deberá seleccionar el campo con el valor Emerald.</p> <p>R4.2: Para el campo búsqueda_por_campos se deberá seleccionar al menos uno de los siguientes valores existentes en la herramienta, éstos son los siguientes: título, abstract y keyword.</p> <p>R4.3: Para el campo rango_fechas, el usuario deberá seleccionar los 4 campos de fechas existente en la herramienta.</p> <p>R4.3.1: Para el primer campo desde_mes el usuario deberá seleccionar un mes de inicio para realizar la búsqueda, los valores van desde enero hasta diciembre.</p> <p>R4.3.2: Para el segundo campo desde_año el usuario deberá seleccionar un año de inicio para realizar la búsqueda, los valores van desde 1999 hasta 2017.</p> <p>R4.3.3: Para el primer campo hasta_mes el usuario deberá seleccionar un mes de fin para realizar la búsqueda, los valores van desde enero hasta diciembre</p> <p>R4.3.4: Para el primer campo hasta_año el usuario deberá seleccionar un año de fin para realizar la búsqueda, los valores van desde 1999 hasta 2017.</p> <p>R4.4: Para el campo string_búsqueda será una cadena de tamaño máximo 300.</p> <p>R4.4.1: El usuario deberá ingresar el string_búsqueda en minúscula.</p>

	<p>R4.4.2: Si <code>string_busqueda</code> es una frase se deberá ingresar entre comillas.</p> <p>R4.4.3: Si <code>string_busqueda</code> contiene el operador lógico OR, las frases o palabras tendrán que ir entre paréntesis. Por ejemplo: (“systematic review” or “mapping study” or survey).</p> <p>R4.5: <code>string_busqueda</code> puede contener tanto números como letras, no están permitidos los caracteres especiales, solo “” * & ?.</p> <p>R4.6: Para realizar la búsqueda los requisitos R4.1, R4.2 y R4.4 son obligatorios y el requisito R4.3 es opcional.</p>
Salidas	<p>Listado de metadatos en Emerald. La información de los metadatos es dependiendo del tipo de registro</p> <ul style="list-style-type: none"> ➤ Si el tipo de registro es Artículo los metadatos son los siguientes: <i>author, title, journal, year</i> y <i>DOI</i>. ➤ Si el tipo de registro es Incollection los metadatos son los siguientes: <i>author o editor, title, Publisher, year</i> y <i>DOI</i>. ➤ Si el tipo de registro es Artículo los metadatos son los siguientes: <i>author, title, booktitle, Publisher, year</i> y <i>DOI</i>. ➤ Si el tipo de registro es Inproceedings los metadatos son los siguientes: <i>author, title, booktitle, year</i> y <i>DOI</i>.

R5 Consultar por artículos en Taylor and Francis: Esta función permite el ingreso del string de búsqueda, que introduzca dicho string en el motor de búsqueda de la herramienta siendo ingresado en la base de datos Taylor and Francis y que recopile los metadatos de la búsqueda.

Nombre	R5.- Consultar por artículos en Taylor and Francis.
Entradas	bases_datos, búsqueda_por_campos, rango_fechas, string_búsqueda.
Proceso	<p>R5.1: Para el campo bases_datos se deberá seleccionar el campo con el valor Taylor and Francis.</p> <p>R5.2: Para el campo búsqueda_por_campos se deberá seleccionar al menos uno de los siguientes valores existentes en la herramienta, éstos son los siguientes: título, abstract y keyword.</p> <p>R5.3: Para el campo rango_fechas, el usuario deberá seleccionar los 4 campos de fechas existente en la herramienta.</p> <p style="padding-left: 40px;">R5.3.1: Para el primer campo desde_mes el usuario deberá seleccionar un mes de inicio para realizar la búsqueda, los valores van desde enero hasta diciembre.</p> <p style="padding-left: 40px;">R5.3.2: Para el segundo campo desde_año el usuario deberá seleccionar un año de inicio para realizar la búsqueda, los valores van desde 1999 hasta 2017.</p> <p style="padding-left: 40px;">R5.3.3: Para el primer campo hasta_mes el usuario deberá seleccionar un mes de fin para realizar la búsqueda, los valores van desde enero hasta diciembre.</p> <p style="padding-left: 40px;">R5.3.4: Para el primer campo hasta_año el usuario deberá seleccionar un año de fin para realizar la búsqueda, los valores van desde 1999 hasta 2017.</p> <p>R5.4: Para el campo string_búsqueda será una cadena de tamaño máximo 300.</p>

	<p>R5.4.1: El usuario deberá ingresar el <code>string_busqueda</code> en minúscula.</p> <p>R5.4.2: Si <code>string_busqueda</code> es una frase se deberá ingresar entre comillas.</p> <p>R5.4.3: Si <code>string_busqueda</code> contiene el operador lógico OR, las frases o palabras tendrán que ir entre paréntesis. Por ejemplo: (“systematic review” or “mapping study” or survey).</p> <p>R5.5: <code>string_busqueda</code> puede contener tanto números como letras, no están permitidos los caracteres especiales, solo “” * & ?.</p> <p>R5.6: Para realizar la búsqueda los requisitos R5.1, R5.2 y R5.4 son obligatorios y el requisito R5.3 es opcional.</p>
Salidas	<p>Listado de metadatos en Taylor and Francis. La información de los metadatos es dependiendo del tipo de registro.</p> <ul style="list-style-type: none"> ➤ Si el tipo de registro es Artículo los metadatos son los siguientes: <i>author, title, journal, year</i> y <i>DOI</i>. ➤ Si el tipo de registro es Incollection los metadatos son los siguientes: <i>author o editor, title, Publisher, year</i> y <i>DOI</i>. ➤ Si el tipo de registro es Artículo los metadatos son los siguientes: <i>author, title, booktitle, Publisher, year</i> y <i>DOI</i>. ➤ Si el tipo de registro es Inproceedings los metadatos son los siguientes: <i>author, title, booktitle, year</i> y <i>DOI</i>.

R6 Consultar por artículos en Scopus: Esta función permite el ingreso del string de búsqueda, que introduzca dicho string en el motor de búsqueda de la herramienta siendo ingresado en la base de datos Scopus y que recopile los metadatos de la búsqueda.

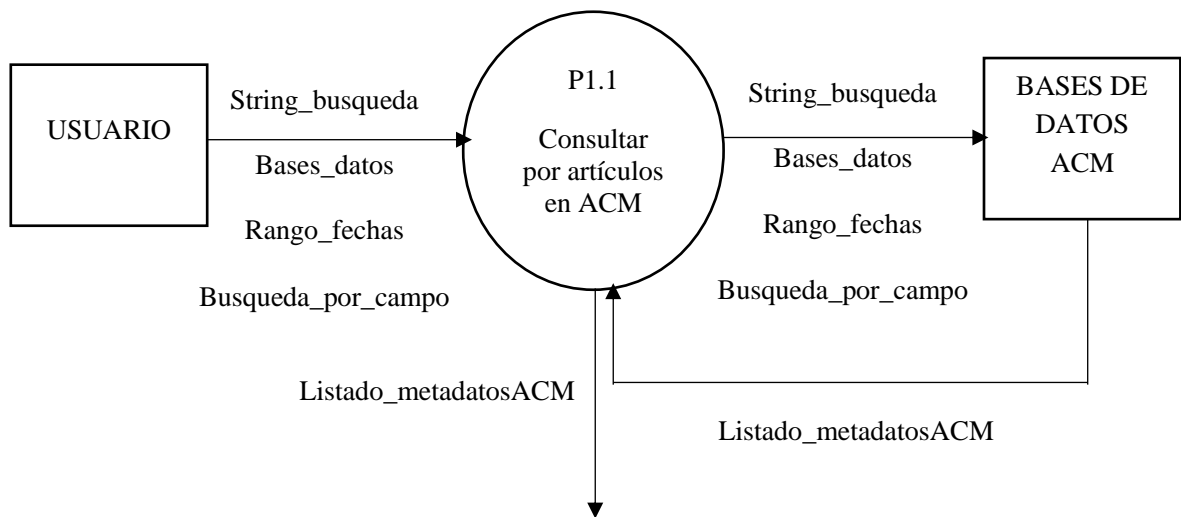
Nombre	R6.- Consultar por artículos en Scopus.
Entradas	bases_datos, búsqueda_por_campos, rango_fechas, string_búsqueda.
Proceso	<p>R6.1: Para el campo bases_datos se deberá seleccionar el campo con el valor Scopus.</p> <p>R6.2: Para el campo búsqueda_por_campos se deberá seleccionar al menos uno de los siguientes valores existentes en la herramienta, éstos son los siguientes: título, abstract y keyword.</p> <p>R6.3: Para el campo rango_fechas, el usuario deberá seleccionar los 4 campos de fechas existente en la herramienta.</p> <p style="padding-left: 40px;">R6.3.1: Para el primer campo desde_mes el usuario deberá seleccionar un mes de inicio para realizar la búsqueda, los valores van desde enero hasta diciembre.</p> <p style="padding-left: 40px;">R6.3.2: Para el segundo campo desde_año el usuario deberá seleccionar un año de inicio para realizar la búsqueda, los valores van desde 1999 hasta 2017.</p> <p style="padding-left: 40px;">R6.3.3: Para el primer campo hasta_mes el usuario deberá seleccionar un mes de fin para realizar la búsqueda, los valores van desde enero hasta diciembre.</p> <p style="padding-left: 40px;">R6.3.4: Para el primer campo hasta_año el usuario deberá seleccionar un año de fin para realizar la búsqueda, los valores van desde 1999 hasta 2017.</p> <p>R6.4: Para el campo string_búsqueda será una cadena de tamaño máximo 300.</p> <p style="padding-left: 40px;">R6.4.1: El usuario deberá ingresar el string_búsqueda en minúscula.</p>

	<p>R6.4.2: Si <code>string_busqueda</code> es una frase se deberá ingresar entre comillas.</p> <p>R6.4.3: Si <code>string_busqueda</code> contiene el operador lógico OR, las frases o palabras tendrán que ir entre paréntesis. Por ejemplo: (“systematic review” or “mapping study” or survey).</p> <p>R6.5: <code>string_busqueda</code> puede contener tanto números como letras, no están permitidos los caracteres especiales, solo “” * & ?.</p> <p>R6.6: Para realizar la búsqueda los requisitos R6.1, R6.2 y R6.4 son obligatorios y el requisito R6.3 es opcional.</p>
Salidas	<p>Listado de metadatos en Scopus. La información de los metadatos es dependiendo del tipo de registro.</p> <ul style="list-style-type: none"> ➤ Si el tipo de registro es Artículo los metadatos son los siguientes: <i>author, title, journal, year</i> y <i>DOI</i>. ➤ Si el tipo de registro es Incollection los metadatos son los siguientes: <i>author o editor, title, Publisher, year</i> y <i>DOI</i>. ➤ Si el tipo de registro es Artículo los metadatos son los siguientes: <i>author, title, booktitle, Publisher, year</i> y <i>DOI</i>. ➤ Si el tipo de registro es Inproceedings los metadatos son los siguientes: <i>author, title, booktitle, year</i> y <i>DOI</i>.

B.6. Diagrama de flujos de datos.

En esta sección de los anexos se presentan los diagramas de flujos de datos, los que permiten representar el cómo se distribuyen estos datos para realizar la búsqueda automatizada. Las funciones de los diagramas son las mismas que se representan en la especificación de requisitos.

B.6.1. Consultar por artículos en ACM.



Listado_metadatosACM

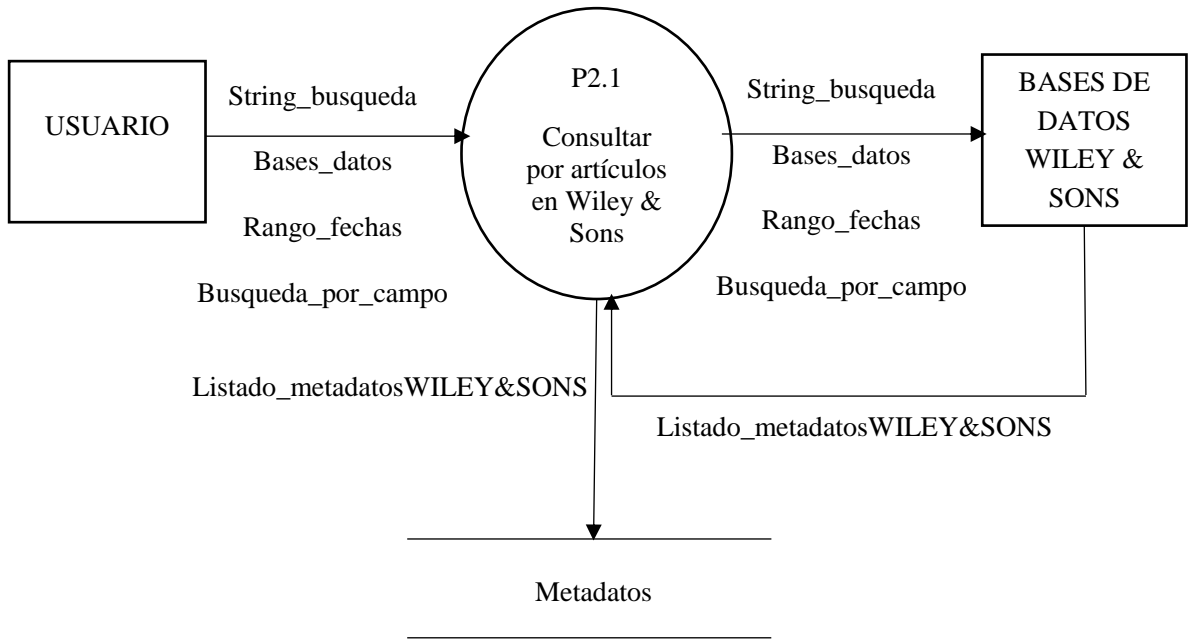
Autor
Booktitle
Título
Año
Journal
DOI
Publisher

Metadatos

Rango_Fechas

desde_mes
desde_año
hasta_mes
hasta_año

B.6.2. Consultar por artículos en Wiley & Sons.



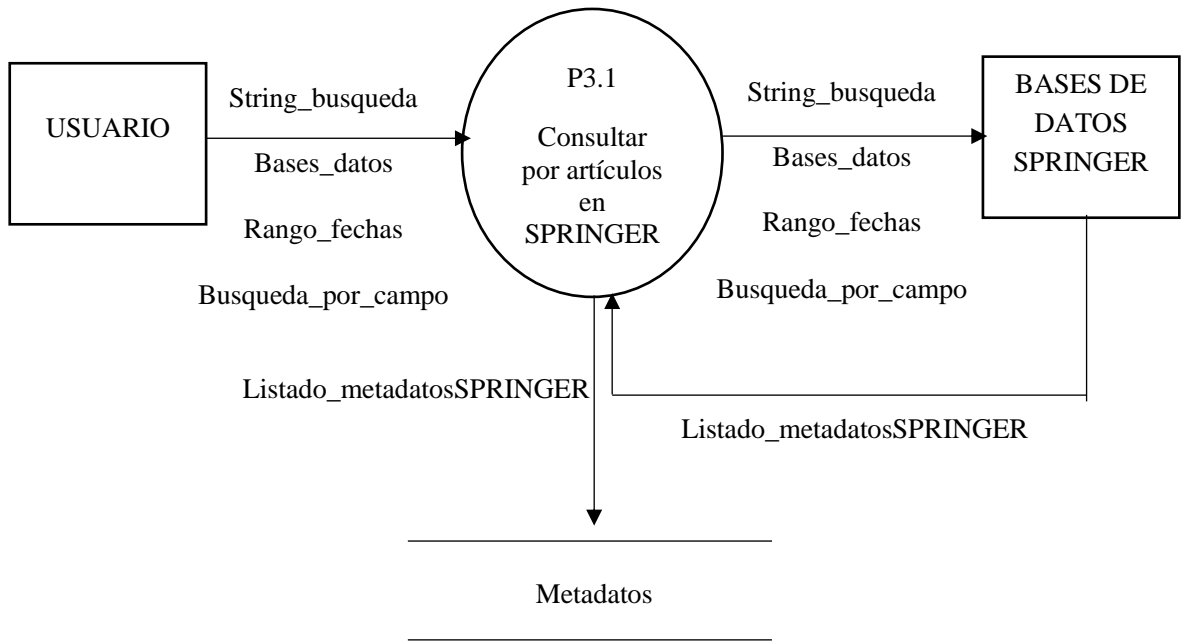
Listado_metadatosWiley&Sons

- Autor
- Booktitle
- Título
- Año
- Journal
- DOI
- Publisher

Rango_Fechas

- desde_mes
- desde_año
- hasta_mes
- hasta_año

B.6.3. Consultar por artículos en Springer.



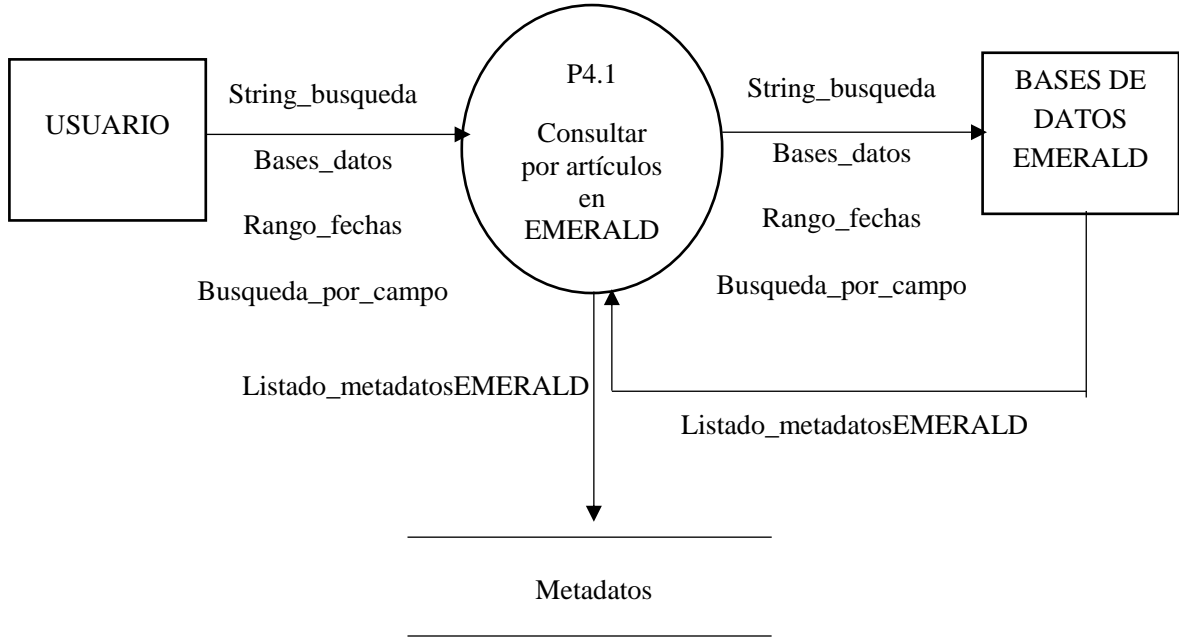
Listado_metadatosSpringer

- Autor
- Booktitle
- Título
- Año
- Journal
- DOI
- Publisher

Rango_Fechas

- desde_mes
- desde_año
- hasta_mes
- hasta_año

B.6.4. Consultar por artículos en Emerald.



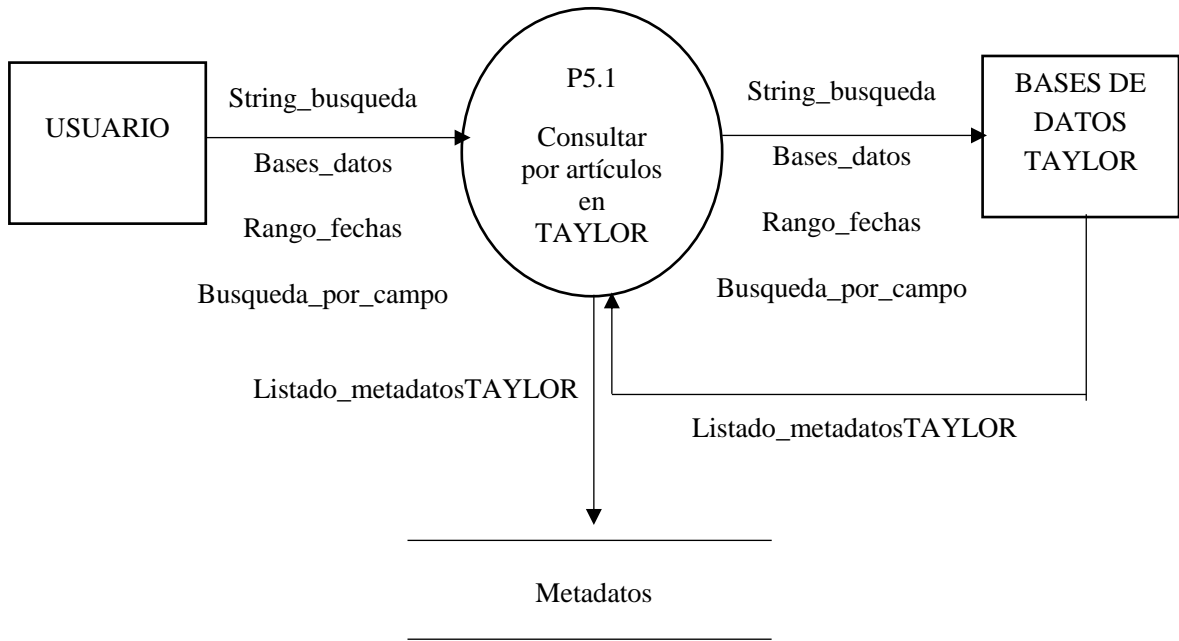
Listado_metadatosEmerald

Autor
 Booktitle
 Título
 Año
 Journal
 DOI
 Publisher

Rango_Fechas

desde_mes
 desde_año
 hasta_mes
 hasta_año

B.6.5. Consultar por artículos en Taylor and Francis.



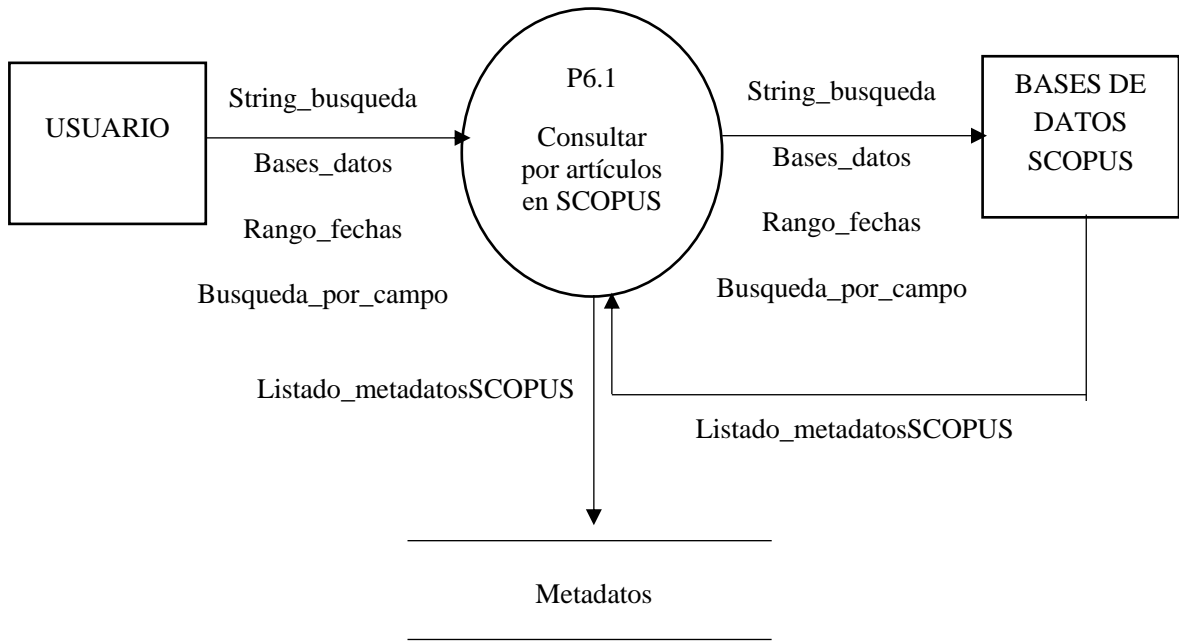
Listado_metadatosTaylor

- Autor
- Booktitle
- Título
- Año
- Journal
- DOI
- Publisher

Rango_Fechas

- desde_mes
- desde_año
- hasta_mes
- hasta_año

B.6.6. Consultar por artículos en Scopus.



Listado_metadatosScopus

- Autor
- Booktitle
- Título
- Año
- Journal
- DOI
- Publisher

Rango_Fechas

- desde_mes
- desde_año
- hasta_mes
- hasta_año

Apéndice C

Código Fuente del Segundo Incremento de la Herramienta

Se mostrará el código solo para el filtro título+ fecha.

C.1. Archivo: *server.php*

```
1  <?php
2
3  class server{
4
5      /* PARA QUE CREA QUE ES UN NAVEGADOR.*/
6      private $agent= 'Mozilla/5.0 (Windows; U;
7                      |Windows NT 6.1; en-US; rv:1.8.1.13)
8                      |Gecko/20080311 Firefox/2.0.0.13)';
9      protected $url_base;
10     protected $url_query;
11     protected $query;
12
13
14     protected function getHtmlResultados($url, $cookie = false){
15         /*Por defecto CURL no guarda las cookies, pero
16         es facil obligarle a que lo haga*/
17         /*Si el servidor al que te quieres conectar genera cookies
18         realizas las siguientes instrucciones, es importante advertir
19         que para que las cookies se guarden correctamente, el usuario
20         de apache tiene que tener acceso de lectura y escritura en el
21         directorio que especifiques*/
```

```

22 if($cookie) {
23     $ckfile = tempnam ("/tmp", 'cookiename');
24     $curl = curl_init();
25     curl_setopt($curl, CURLOPT_COOKIESESSION, true);
26     curl_setopt($curl, CURLOPT_COOKIEJAR, $ckfile);
27     curl_setopt($curl, CURLOPT_COOKIEFILE, $ckfile);
28     curl_setopt($curl, CURLOPT_CONNECTTIMEOUT, 30);
29     curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, false);
30     curl_setopt($curl, CURLOPT_VERBOSE, true);
31     curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);
32     /*El navegador que usamos, para que crea que es un navegador*/
33     curl_setopt($curl, CURLOPT_USERAGENT, $this->agent);
34     /*URL donde nos vamos a conectar para poder extraer la información*/
35     curl_setopt($curl, CURLOPT_URL, $url);
36     curl_setopt($curl, CURLOPT_FOLLOWLOCATION, true);
37     curl_setopt($curl, CURLOPT_HTTP_VERSION, CURL_HTTP_VERSION_1_1);
38     /*Guardar Página*/
39     $html = curl_exec($curl);
40
41     if(curl_errno($curl)){
42         die(curl_error($curl));
43     }
44     /*Cerrar la conexion*/
45     curl_close($curl);
46     return $html;
47 }else{
48     /*Inicializa el objeto curl con las opciones por defecto*/
49     $curl = curl_init();
50     curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, false);
51     curl_setopt($curl, CURLOPT_VERBOSE, true);
52     curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);
53     /*El navegador que usamos, para que crea que es un navegador*/
54     curl_setopt($curl, CURLOPT_USERAGENT, $this->agent);
55     /*URL donde nos vamos a conectar para poder extraer la información*/
56     curl_setopt($curl, CURLOPT_URL, $url);
57     curl_setopt($curl, CURLOPT_FOLLOWLOCATION, true);
58     curl_setopt($curl, CURLOPT_HTTP_VERSION, CURL_HTTP_VERSION_1_1);
59
60     /*Guardar Página*/
61     $html = curl_exec($curl);
62
63     if(curl_errno($curl)){
64         die(curl_error($curl));
65     }
66     /*Cerrar la conexion*/
67     curl_close($curl);
68     return $html;
69 }

```

```

70 protected function getHtmlBib($url, $cookie = false){
71     if($cookie){
72         $ckfile = tempnam ("/tmp", 'cookiename');
73         $curl = curl_init();
74         curl_setopt($curl, CURLOPT_COOKIESESSION, true);
75         curl_setopt($curl, CURLOPT_COOKIEJAR, $ckfile);
76         curl_setopt($curl, CURLOPT_COOKIEFILE, $ckfile);
77         curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, false);
78         curl_setopt($curl, CURLOPT_VERBOSE, true);
79         curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);
80         curl_setopt($curl, CURLOPT_USERAGENT, $this->agent);
81         curl_setopt($curl, CURLOPT_URL, $url);
82         curl_setopt($curl, CURLOPT_FOLLOWLOCATION, true);
83
84         $html = curl_exec($curl);
85
86         if(curl_errno($curl)){
87             die(curl_error($curl));
88         }
89         curl_close($curl);
90         return $html;
91     }else{
92         $curl = curl_init();
93         curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, false);
94         curl_setopt($curl, CURLOPT_VERBOSE, true);
95         curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);
96         curl_setopt($curl, CURLOPT_USERAGENT, $this->agent);
97         curl_setopt($curl, CURLOPT_URL, $url);
98         curl_setopt($curl, CURLOPT_FOLLOWLOCATION, true);
99
100        $html = curl_exec($curl);
101        if(curl_errno($curl)){
102            die(curl_error($curl));
103        }
104        curl_close($curl);
105        return $html;
106    }
107 }
108 }

```

C.2. Archivo: *acm.php*

```
1 <?php
2 /*Incluir el fichero server.php*/
3 include_once 'server.php';
4 /*script se puede ejecutar por siempre*/
5 set_time_limit(0);
6
7 class acm extends server {
8     /*Declaración de Variables*/
9     private $html = "";
10    private $resultados_por_pagina = 0;
11    private $start = "";
12    private $export = "";
13    private $count=0;
14
15    /*Metodo Constructor ACM */
16    public function __construct($query, $desde_anio='') {
17        /*Parámetros de la url*/
18        $this->url_base = 'http://dl.acm.org/';
19        $this->url_query = 'results.cfm?query=';
20        $this->query = $query;
21        $this->desde_anio = '&dte='.$desde_anio;
22        $this->start = '&start=';
23        $this->resultados_por_pagina = 20;
24        include_once "simple_html_dom.php";
25        /*se crea un nuevo objeto*/
26        $this->html = new simple_html_dom();
27
28    }
```

```

30 public function getPaginaResultadoTitulo($compuesto=false){
31     if($compuesto){
32
33         $this->query = $this->toSPRINGERStringTitulo();
34
35         /*Para reemplazar algunos terminos de
36         la cadena de busqueda */
37
38         $this->query = str_replace('and', '+AND', $this->query);
39         $this->query = str_replace('or', 'OR', $this->query);
40         $this->query = str_replace('+)', ')', $this->query);
41         $this->query = str_replace(' ', '+', $this->query);
42         $this->query = str_replace('_', '+', $this->query);
43         $this->query = str_replace('"', '%22', $this->query);
44
45     }
46
47     /*Indicarle al objeto cual es la
48     pagina a la cual queremos extraer la
49     informacion*/
50     $this->html->load($this->getHtmlResultados($this->url_base.
51     $this->url_query . $this->query.$this->desde_anio));
52     echo "<br>-----ACM-----". "<br/>";
53     // BUSCAR CANTIDAD RESULTADOS
54     $cantidad = $this->getCantidadResultados();
55     echo "cantidad resultados: {$cantidad}<br />";
56     // CANTIDAD ITERACIONES
57     $cantidad_paginas = $cantidad / $this->resultados_por_pagina;
58     echo "cantidad páginas: {$cantidad_paginas}<br />";
59     $cantidad_paginas = ceil($cantidad_paginas);
60     echo "cantidad páginas redondeado: {$cantidad_paginas}<br />";
61     /*El bucle continua hasta
62     que recorra todas las paginas
63     encontradas en la búsqueda
64     realizada, en cada iteración
65     se llama a la funcion getPagina*/
66     for($i = 0; $i < $cantidad_paginas; $i++){
67         $this->getPagina($i);
68
69     }
70 }
71
72 /*Esta funcion permite que la cadena
73 de busqueda ingresada por el usuario
74 se ajuste al formato de ACM digital library,
75 permite que a cada termino que se ingrese se
76 le agregue la palabra acmdlTitle*/
77 public function toSPRINGERStringTitulo(){

```

```

77      /*Asume que es unicamente por el titulo*/
78      /*Elimina espacios o saltos de lineas en un string*/
79      $string = trim($this->query);
80      /*Cortar string en base a un caracter*/
81      $string = explode(' ', $string);
82
83      $string_salida = ' ';
84      /*el bucle asigna a la variable
85      $termino el valor del elemento actual
86      que esta recorriendo en ese momento,
87      en la siquient iteracion devolverá
88      el siguiente valor, hasta recorrer
89      todo el string que se ingreso en
90      la busqueda*/
91      foreach ($string as $termino) {
92          $bandera_cierre_parentesis = false;
93          /*Si el elemento actual que se esta
94          recorriendo es distinto a "or" o
95          a "and" */
96          if(!($termino == "or" || $termino == "and")){
97              /*Si se encuentra '(' en el elemento
98              actual*/
99              if (strpos($termino, '(') !== false) {
100                  $string_salida .= '(';
101                  /*se reemplaza el elemento por nada*/
102                  $termino = str_replace('(', '', $termino);
103              }
104              if (strpos($termino, ')') !== false) {
105                  $bandera_cierre_parentesis = true;
106                  $termino = str_replace(')', '', $termino);
107              }
108              /*Se reemplaza el valor encontrado
109              por acmdlTitle:(termino)*/
110              $termino = str_replace($termino, "acmdlTitle:
111              ({$termino})+", $termino);
112              $string_salida .= $termino . ' ';
113              if($bandera_cierre_parentesis) $string_salida .= ')';
114          }else{
115              $string_salida .= $termino . ' ';
116          }
117      }
118
119
120      $string_salida = trim($string_salida);

```

```

511 private function getPagina($pagina){
512     $this->html = new simple_html_dom();
513     $this->html->load($this->getHtmlResultados
514     ($this->url_base. $this->url_query .
515     $this->query . $this->desde_anio.
516     $this->start .
517     ($this->resultados_por_pagina * $pagina));
518     /*Obtener div results*/
519     /*El id del menu. Único ítem*/
520     $resultado = $this->html->find('#results', 0);
521     /*La clase de cada link. Múltiples resultados*/
522     $detalles = $resultado->find('div.details');
523     echo "<h1>Detalles " . count($detalles) . "</h1>";
524     $contador = 1;
525     /*el bucle asigna a la variable
526     $detalle el valor del elemento actual
527     que esta recorriendo en ese momento,
528     en la siguiente iteracion devolverá
529     el siguiente valor, hasta recorrer
530     todos los resultados encontrados
531     de la busqueda*/
532     foreach ($detalles as $detalle) {
533         /*Se encuentra el hijo de la etiqueta
534         <div class="title">*/
535         $href_item = $detalle->find('div.title a', 0)->href;
536         echo "<br>".$href_item."<br>";
537         $this->getBib($href_item, $contador);
538         echo "<br><br>";
539         /*Se retrasa la ejecucion para que la
540         página ACM no bloquee al bot*/
541         sleep (rand (3,5));
542         $contador++;
543     }
544 }
545
546 private function getCantidadResultados(){
547     /*se crea una variable div_cantidad,
548     en donde se almacena la pagina HTML
549     que nos conectamos utilizaremos el
550     metodo find y le pasaremos la etiqueta
551     <div id="searchtots">*/
552     $div_cantidad = $this->html->find('#searchtots', 0);
553     /*se obtiene el primer hijo de la etiqueta #searchtots*/
554     $strong = $div_cantidad->first_child();

```

```

555     $cantidad = $strong->plaintext;
556     /*Eliminar puntos y comas*/
557     $cantidad = str_replace(',',' ',
558     str_replace('.', ' ', $cantidad));
559     return $cantidad;
560 }
561
562 private function getBib($id, $contador){
563     /*dividimos el string solo para
564     obtener el id*/
565     $id = explode('&', explode('&', $id)[0])[1];
566     /*Parámetros de la URL para obtener los metadatos
567     en formato BIBTEX*/
568     $this->export = "exportformats.cfm?id={$id}
569     &expformat=bibtex";
570     /*Crea un objeto DOM a partir de un string*/
571     $html = str_get_html($this->getHtmlBib
572     ($this->url_base . $this->export));
573     $pre = $html->find("#{$id}", 0)->innertext;
574     $cantidad = $this->getCantidadResultados();
575     echo "<br>resultado: {$contador}</br>";
576     echo $pre;
577     /*Guardar metadatos en un archivo*/
578     $file = fopen("archivo.txt", "a");
579     fwrite($file, $pre . PHP_EOL);
580     fclose($file);
581 }
582 }

```

C.3. Archivo: *taylorfrancis.php*

```
1 <?php
2 /*Incluir el fichero server.php*/
3 include_once 'server.php';
4 /*La función permite que el
5 script se puede ejecutar por siempre*/
6 set_time_limit(0);
7
8 class taylorfrancis extends server {
9     /*Declaración de Variables*/
10    private $html = "";
11    private $html_segunda = "";
12    private $resultados_por_pagina = 0;
13    private $start = "";
14    private $parametro_pagina = "";
15    private $export = "";
16    private $count=0;
17    private $desde_mes;
18    private $desde_anio;
19    private $hasta_mes;
20    private $hasta_anio;
21
22    /*Método Constructor Taylor & Francis*/
23    public function __construct($query, $desde_mes='',
24                               $desde_anio='', $hasta_mes='',
25                               $hasta_anio='') {
26        /*Parámetros de la URL*/
27        $this->url_base = 'http://www.tandfonline.com/';
28        $this->url_query = 'action/doSearch?AllField=';
29        $this->query = $query;
30        $this->desde_mes = '&AfterMonth=' . $desde_mes;
31        $this->desde_anio = '&AfterYear=' . $desde_anio;
32        $this->hasta_mes = '&BeforeMonth=' . $hasta_mes;
```

```

33     $this->hasta_anio = '&BeforeYear=' . $hasta_anio;
34     $this->start = '&startPage=';
35     $this->parametro_pagina = '&pageSize=';
36     $this->resultados_por_pagina = 50;
37     $this->export = 'action/downloadCitation?doi=';
38     /*Incluimos el archivo simple_html_dom.php*/
39     include_once "simple_html_dom.php";
40     /*Se crean dos nuevos objetos*/
41     $this->html = new simple_html_dom();
42     $this->html_segunda = new simple_html_dom();
43 }
44
45 public function getPaginaResultado_titulo($compuesto=false){
46     if($compuesto){
47
48         $this->query = $this->toSPRINGERStringTitulo();
49         /*Para reemplazar algunos terminos de
50         la cadena de búsqueda */
51         $this->query = str_replace('and', '+AND', $this->query);
52         $this->query = str_replace('or', 'OR', $this->query);
53         $this->query = str_replace('+)', ')', $this->query);
54         $this->query = str_replace(' ', '+', $this->query);
55         $this->query = str_replace('_', '+', $this->query);
56         $this->query = str_replace('"', '%22', $this->query);
57     }
58     /*Indicarle al objeto cual es la
59     pagina a la cual queremos extraer la
60     informacion*/
61     $this->html->load($this->getHtmlResultados
62     ($this->url_base . $this->url_query . $this->query.
63     $this->desde_mes.$this->desde_anio . $this->hasta_mes .
64     $this->hasta_anio, true));
65
66     $this->getCantidadResultados();
67     echo "<br>-----TAYLOR&FRANCIS-----".<br/>";
68     /*Buscar Cantidad de Resultados*/
69     $cantidad = $this->getCantidadResultados();
70     echo "cantidad resultados: {$cantidad}<br />";
71     /*Cantidad de Iteraciones*/
72     $cantidad_paginas = $cantidad / $this->resultados_por_pagina;
73     echo "cantidad páginas: {$cantidad_paginas}<br />";
74     $cantidad_paginas = ceil($cantidad_paginas);
75     echo "cantidad páginas redondeado: {$cantidad_paginas}<br />";
76

```

```

77      /*El bucle continua hasta
78      que recorra todas las paginas
79      encontradas en la búsqueda
80      realizada, en cada iteración
81      se llama a la funcion getPagina*/
82      for($i = 0; $i < $cantidad_paginas; $i++){
83          $this->getPagina($i);
84
85      }
86  }
87
88  public function toSPRINGERStringTitulo(){
89      /*Asume que es unicamente por el título*/
90      /*Elimina espacios o saltos de linea en un string*/
91      $string = trim($this->query);
92      /*Cortar string en base a un caracter*/
93      $string = explode(' ', $string);
94
95      $string_salida = ' ';
96      /*el bucle asigna a la variable
97      $termino el valor del elemento actual
98      que esta recorriendo en ese momento,
99      en la siguiente iteracion devolverá
100     el siguiente valor, hasta recorrer
101     todo el string que se ingreso en
102     la busqueda*/
103     foreach ($string as $termino) {
104         $bandera_cierre_parentesis = false;
105         /*Si el elemento actual que se esta
106         recorriendo es distinto a "or" o
107         a "and" */
108         if(!($termino == "or" || $termino == "and")){
109             /*Si se encuentra '(' en el elemento
110             actual*/
111             if (strpos($termino, '(') !== false) {
112                 $string_salida .= '(';
113                 /*se reemplaza el elemento por nada*/
114                 $termino = str_replace('(', '', $termino);
115             }
116             if (strpos($termino, ')') !== false) {
117                 $bandera_cierre_parentesis = true;
118                 $termino = str_replace(')', '', $termino);
119             }
120             /*Se reemplaza el valor encontrado

```

```

121         por title:(termino)*/
122         $termino = str_replace($termino,
123         "title:{$termino}+", $termino);
124         $string_salida .= $termino . ' ';
125         if($bandera_cierre_parenthesis) $string_salida .= ')';
126     } else{
127         $string_salida .= $termino . ' ';
128     }
129 }
130 }
131
132 $string_salida = trim($string_salida);
133 return "(". $string_salida . ")";
134 }
683 public function getCantidadResultados() {
684     /*se crea una variable div_cantidad,
685     en donde se almacena la pagina HTML
686     que nos conectamos utilizaremos el
687     metodo find y le pasaremos la etiqueta
688     <div id="searchResultContainer">*/
689     $div_cantidad = $this->html->find
690     ('div#searchResultContainer',0);
691
692     /*Dentro de la etiqueta
693     <div id="searchResultContainer">
694     encontramos la etiqueta
695     <div class="num-results">,
696     en donde la variable $div
697     le asignamos lo encontrado
698     en la etiqueta*/
699     $div = $div_cantidad->find('.num-results',0);
700
701     /*Dentro de la etiqueta
702     <div class="num-results">,
703     encontramos la etiqueta
704     <li class="search-results">,
705     en donde la variable $li
706     le asignamos lo encontrado
707     en la etiqueta*/
708     $li = $div->find('.search-results',0);

```

```

709
710     /*Dentro de la etiqueta
711     <li class="search-results">,
712     encontramos la etiqueta
713     <strong>,
714     en donde la variable $strong
715     le asignamos lo encontrado
716     en la etiqueta*/
717     $strong = $li->find('strong',1);
718     /*Lo convierte a texto*/
719     $cantidad = $strong->plaintext;
720
721     /*Eliminar Puntos y Comas*/
722     $cantidad = str_replace(',',' ', str_replace('.',',', $cantidad));
723     return $cantidad;
724 }
725
726 public function getPagina($pagina){
727
728     $this->html = new simple_html_dom();
729     /*se realiza la conexión con la página*/
730     $this->html->load($this->getHtmlResultados
731     ($this->url_base . $this->url_query . $this->query.
732     $this->desde_mes.$this->desde_anio .
733     $this->hasta_mes . $this->hasta_anio.
734     $this->start . $pagina .
735     $this->parametro_pagina .
736     $this->resultados por pagina,true));
737     /*Encontramos el formulario principal
738     con la etiqueta <form id= "frmSearchResults"*/
739     $formulario = $this->html->find('#frmSearchResults', 0);
740     /*Encontramos la etiqueta que contiene
741     a la lista de resultados*/
742     $ol = $formulario->find('.search-results', 0);
743     /*lista de resultados*/
744     $lis = $ol->find('li');
745     /*Se recorre cada uno de los resultados*/
746     foreach($lis as $li){
747         $article = $li->find('article.searchResultItem', 0);
748         $div = $article->first_child();
749         /*Enlace donde se encuentra el doi
750         de cada resultado*/
751         $href_item = $div->find('a',0)->href;

```

```

752     /*Reemplazamos algunos parametros del link
753     encontrado, para solo obtener el doi*/
754     $href_item = str_replace('/doi/full/', '', $href_item);
755     $href_item = str_replace('/doi/abs/', '', $href_item);
756     $href_item = str_replace('%28', '(', $href_item);
757     $href_item = str_replace('%29', ')', $href_item);
758     echo "<br>".$href_item."<br>";
759     /*Se codifica la cadena (doi)*/
760     $href_item = urlencode($href_item);
761     echo "<br>".$href_item."<br>";
762     $this->html_segunda = new simple_html_dom();
763     /*Conexión a la Página donde se encuentra
764     los metadatos de los resultados*/
765     $this->html_segunda->load($this->getHtmlResultados
766     ($this->url_base . $this->export . "{$href_item}"
767     . "&downloadFileName={$href_item}" .
768     "&include=cit"."&format=bibtex" , true));
769     echo $this->html_segunda . "<br>";
770     /*Guardar metadatos en un archivo*/
771     $file = fopen("archivo.txt", "a");
772     fwrite($file, $this->html_segunda . PHP_EOL);
773     fclose($file);
774
775     echo "<br><br>";
776 }
777 }
778 }

```

C.4. Archivo: *springer.php*

```
1 <?php
2 /*Incluir el fichero server.php*/
3 set_time_limit(0);
4 /*La función permite que el
5 script se puede ejecutar por siempre*/
6 include_once 'server.php';
7
8 class springer extends server {
9     /*Declaración de Variables*/
10    private $html = "";
11    private $resultados_por_pagina = 0;
12    private $start = "";
13    private $export = "";
14
15    /*Método Constructor SpringerLink*/
16    public function __construct($query, $desde_anio='',
17                                $hasta_anio='') {
18        $this->url_base = 'http://link.springer.com/';
19        $this->url_query = '?query=';
20        $this->query = $query;
21        $this->desde_anio = '&facet-start-year='.$desde_anio;
22        $this->start_anio= '&previous-start-year='.$desde_anio;
23        $this->hasta_anio = '&facet-end-year='.$hasta_anio;
24        $this->end_anio = '&previous-end-year='.$hasta_anio;
25        $this->start = 'search/page/';
26        $this->resultados_por_pagina = 20;
27        $this->export = 'citation-needed.services.springer.com/v2/references
28        $this->format_export = '?format=bibtex&flavour=citation';
29        /*Incluimos el archivo simple_html_dom.php*/
30        include_once "simple_html_dom.php";
31        /*Se crea un nuevo objeto*/
32        $this->html = new simple_html_dom();
33    }
34
35    public function getPaginaResultado ($compuesto=false) {
36        if($compuesto){
37            $this->query = $this->toSPRINGERString();
38            /*Para reemplazar algunos terminos de
39            la cadena de búsqueda */
40            $this->query = str_replace('=', '%3D', $this->query);
41            $this->query = str_replace(' ', '+', $this->query);
42            $this->query = str_replace('_', '+', $this->query);
43        }
```

```

44  /*Indicarle al objeto cual es la
45  pagina a la cual queremos extraer la
46  informacion*/
47  $this->html->load($this->getHtmlResultados
48  ($this->url_base . $this->start . '1' . $this->url_query
49  $this->query . $this->desde_anio . $this->start_anio .
50  $this->hasta_anio . $this->end_anio));
51  echo "<br>-----SPRINGER-----". "<br/>";
52  /*Buscar Cantidad de Resultados*/
53  $cantidad = $this->getCantidadResultados();
54  echo "<br>". "cantidad resultados: {$cantidad}<br />";
55  /*Cantidad de Iteraciones*/
56  $cantidad_paginas = $cantidad / $this->resultados_por_pagina
57  echo "cantidad páginas: {$cantidad_paginas}<br />";
58  $cantidad_paginas = ceil($cantidad_paginas);
59  echo "cantidad páginas redondeado: {$cantidad_paginas}<br />
60
61  /*El bucle continua hasta
62  que recorra todas las paginas
63  encontradas en la búsqueda
64  realizada, en cada iteración
65  se llama a la funcion getPagina*/
66  for($k = 0; $k < $cantidad_paginas; $k++){
67      $this->getPagina($k);
68  }
69  }
70  private function getCantidadResultados(){
71      /*Obtener el menú superior*/
72      $div_cantidad = $this->html->find('#kb-nav--main', 0);
73      /*Obtener la cabecera*/
74      $div = $div_cantidad->find('.header', 0);
75      /*Obtenemos la etiqueta h1 con su respectica clase*/
76      $h1 = $div->find
77      ('.number-of-search-results-and-search-terms', 0);
78      /*Obtenemos el primer hijo del h1
79      que es la cantidad de resultado
80      que arroja la búsqueda*/
81      $strong = $h1->first_child();
82      /*Lo convierte a texto*/
83      $cantidad = $strong->plaintext;
84      /*Eliminar Puntos y Comas*/
85      $cantidad = str_replace('.', '', str_replace('.', '', $cantidad));
86      return $cantidad;
87  }

```

```

88
89 □ public function toSPRINGERString(){
90     /*Asume que es únicamente por el título*/
91     /*Elimina espacios y saltod de linea en un string*/
92     $string = trim($this->query);
93     /*Cortar string en base a un caracter*/
94     $string = explode(' ', $string);
95     $string_salida = '';
96 □ /*el bucle asigna a la variable
97     $termino el valor del elemento actual
99     en la siguiente iteración devolverá
100    el siguiente valor, hasta recorrer
101    todo el string que se ingreso en
102    la búsqueda*/
103 □ foreach ($string as $termino) {
104
105     $bandera_cierre_parentesis = false;
106 □ /*Si el elemento actual que se esta
107     recorriendo es distinto a "or" o
108     a "and" */
109 □ if(!($termino == "or" || $termino == "and")){
110 □     /*Si se encuentra '(' en el elemento
111     actual*/
112 □     if (strpos($termino, '(') !== false) {
113         $string_salida .= '(';
114         /*se reemplaza el elemento por nada*/
115         $termino = str_replace('(', '', $termino);
116     }
117 □     if (strpos($termino, ')') !== false) {
118         $bandera_cierre_parentesis = true;
119         $termino = str_replace(')', '', $termino);
120     }
121 □     /*Se reemplaza el valor encontrado
122     por el formato de la página
123     title:(termino)*/
124     $termino = str_replace($termino,
125     "title=({$termino}*)", $termino);
126
127     $string_salida .= $termino . ' ';
128     if($bandera_cierre_parentesis)
129         $string_salida .= ') ';
130 }else{
131     $string_salida .= $termino . ' ';
132 }
133 }
134 $string_salida = trim($string_salida);
135 return $string_salida;
136
137 }

```

```

138
139 private function getPagina($pagina){
140     $this->html = new simple_html_dom();
141     $this->html->load($this->getHtmlResultados
142     ($this->url_base . $this->start . ($pagina+1) .
143     $this->url_query . $this->query . $this->desde_anio .
144     $this->start_anio . $this->hasta_anio . $this->end_anio))
145
146     /*Obtener el menu resultado con id
147     "results-list"*/
148     /* El id del menu. Único item*/
149     $resultado = $this->html->find('#results-list', 0);
150     /*Obtener la lista de resultados
151     del menu anterior. Múltiples Resultados*/
152     $lis = $resultado->find('li');
153     echo "<br /> Son " . count($lis) . " lis <br />";
154     /*Recorrer la lista de resultados*/
155     for($i = 0; $i < count($lis); $i++){
156         echo "<li> n° " . ($i+1) . "<br />";
157         /*Obtener la etiqueta
158         <div class="text">*/
159         $li = $lis[$i]->find('div.text', 0);
170     }else{
171         $href = $li->find('h2 > a', 0)->href;
172
173     }
174
175     /*Reemplazamos algunos parametros del link
176     encontrado, para solo obtener el doi*/
177     $href = str_replace('referenceworkentry/', '', $href);
178     $href = str_replace('chapter/', '', $href);
179     $href = str_replace('article/', '', $href);
180     $href = str_replace('book/', '', $href);
181     $href = str_replace('bookseries/', '', $href);
182     $href = str_replace('_', '%5F', $href);
183     $href = str_replace('protocol/', '', $href);
184     /*Esta función devuelve un string de la
185     estructura HTML de la Página, donde
186     se encuentra los metadatos de resultados
187     encontrados*/
188     $bibtex = file_get_contents("http://{ $this->export }
189     { $href } { $this->format_export }");
190     echo $bibtex;
191     /*Guardar metadatos en un archivo*/
192     $file = fopen("archivo.txt", "a");
193     fwrite($file, $bibtex . PHP_EOL);
194     fclose($file);
195
196     }
197 }

```

C.5. Archivo:emerald.php

```
1 <?php
2 /*Incluir el fichero server.php*/
3 include_once 'server.php';
4 /*La función permite que el
5 -script se puede ejecutar por siempre*/
6 set_time_limit(0);
7
8 class emerald extends server {
9     /*Declaración de Variables*/
10    private $html = "";
11    private $resultados_por_pagina = 0;
12    private $parametro_pagina = "";
13    private $start = "";
14    private $export = "";
15    private $html_segunda="";
16
17    /*Método Constructor Emerald*/
18    public function __construct($query, $desde_mes='',
19                                $desde_anio='', $hasta_mes='',
20                                $hasta_anio='') {
21
22        /*Parámetros de la URL*/
23        $this->url_base = 'http://www.emeraldinsight.com/
24                           action/doSearch?';
25        $this->url_base_export = 'http://www.emeraldinsight.com/
26                                   action/';
27        $this->url_query1 = 'text1=';
28        $this->desde_mes = '&AfterMonth='.$desde_mes;
29        $this->desde_anio = '&AfterYear='.$desde_anio;
30        $this->hasta_mes = '&BeforeMonth='.$hasta_mes;
31        $this->hasta_anio = '&BeforeYear='.$hasta_anio;
32        $this->url_titulo = '&field1=Title';
33        $this->url_abstractabs = '&field1=Abstract';
34        $this->url_keywordkey = '&field1=Keyword';
35        $this->url_conector = '&logicalOpe1=OR';
36        $this->url_conectoror = '&logicalOpe2=OR';
37        $this->url_query2 = '&text2=';
38        $this->url_abstract = '&field2=Abstract';
39        $this->url_query3 = '&text3=';
40        $this->url_keyword = '&field3=Keyword';
41        $this->url_keyword2 = '&field2=Keyword';
42        $this->start = '&startPage=';
43        $this->parametro_pagina = '&pageSize=';
44        $this->query = $query;
```

```

44     $this->resultados_por_pagina = 20;
45     $this->export = 'downloadCitation?doi=';
46     /*Incluimos el archivo simple_html_dom.php*/
47     include_once "simple_html_dom.php";
48     /*Se crean dos nuevos objetos*/
49     $this->html = new simple_html_dom();
50     $this->html_segunda = new simple_html_dom();
51 }
52
53 public function getPaginaResultadoTitulo($compuesto=false){
54     if($compuesto){
55
56         $this->query = $this->toSPRINGERString();
57         /*Para reemplazar algunos terminos de
58         la cadena de búsqueda */
59         $this->query = str_replace('and', '+AND', $this->query);
60         $this->query = str_replace('or', 'OR', $this->query);
61         $this->query = str_replace('+', ''), $this->query);
62         $this->query = str_replace(' ', '+', $this->query);
63         $this->query = str_replace('_', '+', $this->query);
64     }
65
66     /*Indicarle al objeto cual es la
67     pagina a la cual queremos extraer la
68     informacion*/
69     $this->html->load($this->getHtmlResultados
70         ($this->url_base .
71         $this->query .
72         $this->url_titulo .
73         $this->desde_mes .
74         $this->desde_anio .
75         $this->hasta_mes .
76         $this->hasta_anio , true));
77
78     echo "<br>-----EMERALD-----".<br/>";
79
80     /*Buscar Cantidad Resultados*/
81     $cantidad = $this->getCantidadResultados();
82     echo "cantidad resultados: {$cantidad}<br />";
83     /*Cantidad Páginas*/
84     $cantidad_paginas = $cantidad / $this->resultados_por_pagina;
85     echo "cantidad páginas: {$cantidad_paginas}<br />";
86     $cantidad_paginas = ceil($cantidad_paginas);
87     echo "cantidad páginas redondeado: {$cantidad_paginas}<br />";

```

```

132 $articleEntry = $contenedor->find('.articleEntry',0);
133 /*Obtener el segundo div de $articleEntry*/
134 $articleContainer = $articleEntry->find('div', 1);
135 /*Obtener la etiqueta span*/
136 $span = $articleContainer->find('span', 0);
137 /*obtener el primer hijo
138 de $span*/
139 $spanHref = $span->first_child();
140 /*se encuentra la etiqueta "a" apuntando
141 al enlace, indicando la URL, donde
142 se encuentra el doi*/
143 $href_item = $spanHref->find('a',0)->href;
144 echo "<br>".$href_item."<br>";
145 /*Reemplazamos algunos parametros del link
146 encontrado, para solo obtener el doi*/
147 $href_item = str_replace('/doi/pdfplus/', '', $href_item);
148 $href_item = str_replace('/doi/full/', '', $href_item);
149
150 /*Conexión a la Página donde se encuentra
151 los metadatos de los resultados*/
152 $this->html_segunda = new simple_html_dom();
153 $this->html_segunda->load($this->getHtmlResultados
154 ($this->url_base_export .
155 $this->export .
156 "{$href_item}" .
157 "&downloadFileName={$href_item}" .
158 "&include=abs" .
159 "&format=bibtex" ,true));
160 /*Guardar metadatos en un archivo*/
161 $file = fopen("archivo.txt", "a");
162 fwrite($file, $this->html_segunda . PHP_EOL);
163 fclose($file);
164 echo "<br><br>";
165 }
166 }
167

```

C.6. Archivo:wileysons.php

```
1 <?php
2 /*Incluir el fichero server.php*/
3 include_once 'server.php';
4 /*La función permite que el
5 script se puede ejecutar por siempre*/
6 set_time_limit(0);
7
8 class wileysons extends server {
9     /*Declaración de Variables*/
10    private $html = "";
11    private $html_segunda="";
12    private $resultados_por_pagina = 0;
13    private $start = "";
14    private $parametro_pagina = "";
15    private $export = "";
16    private $count=0;
17    private $bibtex ="";
18
19
20    /*Método Constructor Wiley & Sons*/
21    public function __construct($query, $desde_anio='',
22                                $hasta_anio='') {
23        /*Parámetros de la URL*/
24        $this->url_base = 'http://onlinelibrary.wiley.com/';
25        $this->url_base2 = 'http://onlinelibrary.wiley.com';
26        $this->url_query = 'advanced/search/results/reentry?';
27        $this->between = '&dateRange=between';
28        $this->fechas = '&dateRange=allDates';
29        $this->desde_anio = '&startYear='.$desde_anio;
30        $this->hasta_anio = '&endYear='.$hasta_anio;
31        $this->url_titulo = '&searchRowCriteria[0].fieldName=document-titl
32        $this->url_abstract = '&searchRowCriteria[0].fieldName=abstract';
33        $this->url_keyword = '&searchRowCriteria[0].fieldName=keyword';
34        $this->start = '&start=';
35        $this->parametro_pagina = '&pageSize=';
36        $this->resultados_por_pagina = 20;
37        $this->export = 'enhanced/getCitation/doi/';
38        $this->bibtext = '?citation-type';
39        /*Incluimos el archivo simple_html_dom.php*/
40        include_once "simple_html_dom.php";
41        /*Se crean dos nuevos objetos*/
42        $this->html = new simple_html_dom();
43        $this->html_segunda = new simple_html_dom();
44    }
```

```

45
46
47 public function getPaginaResultado_titulo(){
48
49     /*Para reemplazar los espacios
50     por "+" de la cadena de búsqueda */
51     $this->query = str_replace(' ', '+', $this->query);
52     /*Indicarle al objeto cual es la
53     pagina a la cual queremos extraer la
54     informacion*/
55     $this->html->load($this->getHtmlResultados
56     ($this->url_base .
57     $this->url_query .
58     $this->url_query_string .
59     $this->query .
60     $this->desde_anio .
61     $this->hasta_anio.
62     $this->between.
63     $this->url_titulo.
64     $this->start .'1' ));
65
66     echo "<br>-----WILEY&SONS-----".<br/>"
67
68     /*Buscar Cantidad Resultados*/
69     $cantidad = $this->getCantidadResultados();
70     $p = $this->getCantidadResultados();
71     $div_cantidad = $this->getCantidadResultados();
72
73     echo "cantidad resultados: {$cantidad}<br />";
74     /*Cantidad Páginas*/
75     $cantidad_paginas = $cantidad / $this->resultados_por_pagina;
76     echo "cantidad páginas: {$cantidad_paginas}<br />";
77     $cantidad_paginas = ceil($cantidad_paginas);
78     echo "cantidad páginas redondeado: {$cantidad_paginas}<br />";
79     /*El bucle continua hasta
80     que recorra todas las paginas
81     encontradas en la búsqueda
82     realizada, en cada iteración
83     se llama a la funcion getPagina*/
84     for($i = 0; $i < $cantidad_paginas; $i++){
85         $this->getPagina_titulo($i);
86     }
87 }
88

```

```

89 private function getPagina_titulo($pagina){
90     $this->html = new simple_html_dom();
91     /*se realiza la conexión con la página*/
92     $this->html->load($this->getHtmlResultados
93         ($this->url_base .
94         $this->url_query.
95         $this->url_query_string .
96         $this->query .
97         $this->desde_anio .
98         $this->hasta_anio.
99         $this->between.
100        $this->url_titulo.
101        $this->start .
102        (($this->resultados_por_pagina * $pagina)+1)));
103     /*Encontramos el formulario principal
104     con la etiqueta <form id= "frmSearchResults"*/
105     $formulario = $this->html->find('#searchResultsList', 0);
106     /*Obtener el menu donde se
107     encuentran los ítems*/
108     $sol = $formulario->find('.articles',0);
109     /*Obtener la lista
110     de resultados*/
111     $lis = $sol->find('li');
112     /*Recorrer cada uno de
113     los resultados encontrados*/
114     foreach($lis as $li){
115         /*Obtener la etiqueta
116         div con clase "access"*/
117         $article = $li->find('div.access', 0);
118         /*Obtener la etiqueta "input"
119         donde se encuentra el doi*/
120         $doi = $article->find('input', 0)->value;
121         $input = $doi;
122         /*Se realizan reemplazos*/
123         $input = str_replace('<', '%3C', $input);
124         $input = str_replace('>', '%3E', $input);

```

```

125     /*Se obtienen los metadatos,
126     se ocupa esta función porque
127     los datos no se pueden visualizar
128     a través de la URL, por consiguiente se
129     tuvo que abrir el archivo que
130     entrega la pagina para poder
131     extraer los metadatos*/
132     $bibtex = file_get_contents
133             ("{$this->url_base}
134             {$this->export}
135             {$input}
136             ?citation-type=bibtex
137             &doi={$input}
138             &download-citation=Citation"));
139     echo "<br><br>". $bibtex . "<br />-----";
140     /*Guardar metadatos en un archivo*/
141     $file = fopen("archivo.txt", "a");
142     fwrite($file, $bibtex . PHP_EOL);
143     fclose($file);
144     echo "<br><br>";
145
146 }
147
148
149 }
150
151
152 public function getCantidadResultados(){
153     /*Obtener el menu con id "searchResults"*/
154     $div_cantidad = $this->html->find('#searchResults',0){
155     /*Obtener hijo con id "resultInfo"*/
156     $info = $div_cantidad->find('#resultInfo',0);
157     $p = $info->find('#searchedForText',0){
158     /*Obtener la cantidad de resultados*/
159     $em = $p->first_child();
160     /*Lo convierte a texto*/
161     $cantidad = $em->plaintext;
162     /*Eliminar puntos y comas*/
163     $cantidad = str_replace('.', '', str_replace(',', '', $cantidad));
164     return $cantidad;
165
166 }
167
168 }

```

C.7. Archivo:scopus.php

```
1 <?php
2 /*La función permite que el
3 script se puede ejecutar por siempre*/
4 set_time_limit(0);
5 /*Incluir el fichero server.php*/
6 include_once 'server.php';
7
8 class scopus extends server{
9     /*Declaración de Variables*/
10    private $html = "";
11    private $html_segunda="";
12    private $resultados_por_pagina = 0;
13    private $start = "";
14    private $export = "";
15    private $desde_anio;
16    private $hasta_anio;
17    private $numero;
18    private $variable = "";
19
20    /*Método Constructor Scopus*/
21    public function __construct($query, $desde_anio='',
22                                $hasta_anio=''){
23        $rango_fechas = '';
24        foreach(range($desde_anio, $hasta_anio) as $numero){
25            $numero = str_replace($numero, "%22{$numero}%22,t", $numero);
26            $rango_fechas.=$numero;
27
28        }
29
30        /*Parámetros de la URL*/
31        $this->url_base = 'https://www.scopus.com/
32                                results/results.uri?sot=b&sdt=b';
33        $this->url_base_segunda = 'https://www.scopus.com/record/
34                                display.uri?sot=b&sdt=b';
35        $this->url_base_doi = 'https://www.scopus.com/onclick/export.uri?';
36        $this->click = 'onClickExport={"Format":"BIB","View":"CiteOnly"}';
37        $this->url_query = '&s=';
38        $this->id_pagina = '&eid=';
39        $this->record = '&origin=recordpage';
40        $this->titulo = 'TITLE';
41        $this->or = 'OR';
42        $this->cluster = '&cluster=scopusbyr'.$rango_fechas;
43        $this->abstract = 'ABS';
```

```

44     $this->keyword = 'KEY';
45     $this->titleabskey = 'TITLE-ABS-KEY';
46     $this->titleabs = 'TITLE-ABS';
47     $this->query = "(". $query . ")";
48     $this->start = '&offset=';
49     $this->resultados_por_pagina = 20;
50     /*Incluimos el archivo simple_html_dom.php*/
51     include_once "simple_html_dom.php";
52     /*Se crean dos nuevos objetos*/
53     $this->html = new simple_html_dom();
54     $this->html_segunda = new simple_html_dom();
55
56
57 }
58
59
60 public function getPaginaResultado_Titulo(){
61     /*Reemplazar algunos términos
62     de la cadena de búsqueda*/
63     $this->query = str_replace('and', 'AND', $this->query);
64     $this->query = str_replace('or', 'OR', $this->query);
65     $this->query = str_replace(' ', '+', $this->query);
66
67     /*Indicarle al objeto cual es la
68     pagina a la cual queremos extraer la
69     informacion*/
70     $this->html->load($this->getHtmlResultados
71     | ($this->url_base .
72     | $this->url_query .
73     | $this->titulo .
74     | $this->query .
75     | $this->cluster, true));
76     echo "<br>-----SCOPUS-----".<br/>";
77
78
79     /*Buscar cantidad de resultados*/
80     $cantidad = $this->getCantidadResultados();
81     echo "cantidad resultados: {$cantidad}<br />";
82     /*Cantidad de páginas*/
83     $cantidad_paginas = $cantidad / $this->resultados_por_pagina;
84     echo "cantidad páginas: {$cantidad_paginas}<br />";
85     $cantidad_paginas = ceil($cantidad_paginas);
86     echo "cantidad páginas redondeado: {$cantidad_paginas}<br />"

```

```

87 |         /*El bucle continua hasta
88 |         que recorra todas las paginas
89 |         encontradas en la búsqueda
90 |         realizada, en cada iteración
91 |         se llama a la funcion getPagina*/
92 |         for($i = 0; $i < $cantidad_paginas; $i++){
93 |             $this->getPagina_Titulo($i);
94 |
95 |     }
96 |
97 |
98 |     public function getPagina_Titulo($pagina){
99 |
100 |         $this->html = new simple_html_dom();
101 |         /*se realiza la coneccion con la pagina
102 |         de scopus*/
103 |         $this->html->load($this->getHtmlResultados
104 |             ($this->url_base .
105 |             $this->url_query .
106 |             $this->titulo .
107 |             $this->query .
108 |             $this->cluster.
109 |             $this->start .
110 |             (($this->resultados_por_pagina * $pagina)+1), true));
111 |
112 |         /*Obtener formulario con
113 |         id "resultsBody"*/
114 |         $formulario = $this->html->find('#resultsBody', 0);
115 |         /*Obtener el menu donde
116 |         se encuentran los ítems*/
117 |         $lista = $formulario->find('#srchResultsList', 0);
118 |         /*Obtener lista de ítems*/
119 |         $trs = $lista->find('tr.searchArea');
120 |         /*Recorrer cada uno de
121 |         los resultados encontrados*/
122 |         foreach ($trs as $tr) {
123 |             /*Obtener el primer hijo
124 |             de cada ítems*/
125 |             $titulo = $tr->first_child();
126 |             /*Obtener la clase "checkbox"*/
127 |             $div = $titulo->find('.checkbox', 0);
128 |             /*Obtener la etiqueta "input"
129 |             para conseguir el doi*/
130 |             $item = $div->find('input', 0)->id;
131 |             $item = str_replace('eid_', '', $item);
132 |             /*se codifica el doi*/
133 |             $item= urlencode($item);

```

```

134 |         /*Conexión a la Página donde se encuentra
135 |         los metadatos de los resultados*/
136 |         $this->html_segunda = new simple_html_dom();
137 |         $this->html_segunda->load($this->getHtmlResultados
138 |                                 ($this->url_base_doi .
139 |                                 $this->click .
140 |                                 "&origin=resultslist".
141 |                                 "&eid={$item}"));
142 |
143 |         /*Guardar metadatos en un archivo*/
144 |         $file = fopen("archivo.txt", "a");
145 |         fwrite($file, $this->html_segunda . PHP_EOL);
146 |         fclose($file);
147 |         echo "<br><br>";
148 |     }
149 |
150 | }
151 |
1201 | public function getCantidadResultados(){
1202 |     /*Obtener la clase "documentHeader"*/
1203 |     $div_cantidad = $this->html->find('.documentHeader',0)
1204 |     /*Obtener la cantidad de resultado
1205 |     de la clase "resultsCount"*/
1206 |     $resultado = $div_cantidad->find('.resultsCount',0) {
1207 |     $cantidad = $resultado->plaintext;
1208 |     /*Eliminar puntos y comas*/
1209 |     $cantidad = str_replace('.', '', str_replace(',', '', $cantidad));
1210 |     return $cantidad;
1211 |
1212 | }
1213 |
1214 | }

```

Apéndice D

Manual de Usuario.

Este manual pretende ser una sencilla guía que permita al usuario entender toda la funcionalidad de la aplicación, así como los pasos a seguir para una correcta utilización.

Para la elaboración de esta guía se ha utilizado el emulador Wamp Server con la versión 2.5 para Windows.

Antes de utilizar la herramienta es necesario instalar el emulador. Lo primero que se deberá hacer es descargar WampServer. Una vez descargado, se instala, el proceso de instalación, que pedirá seleccionar un directorio de instalación. Por defecto, aparece C:/wamp; se puede dejar en esa opción se acepta y se espera a que acabe la instalación.

Cuando acabe de instalarse, se procede a ejecutar la aplicación; wampserver se ejecutará en segundo plano en la barra de tareas y se tornará de un color verde. Ahora deberá activar un componente necesario de PHP para que la herramienta pueda funcionar

Se deberá dar clic en el ícono verde de WampServer; esto desplegará un menú de opciones. Se selecciona PHP, luego extensiones de PHP y se activa lo siguiente con un clic: **php_curl**.

Para saber si WampServer está correctamente instalado y funcionando, se accede desde el navegador a la siguiente dirección <http://localhost> y deberá aparecer lo siguiente como se muestra en la Figura 66.

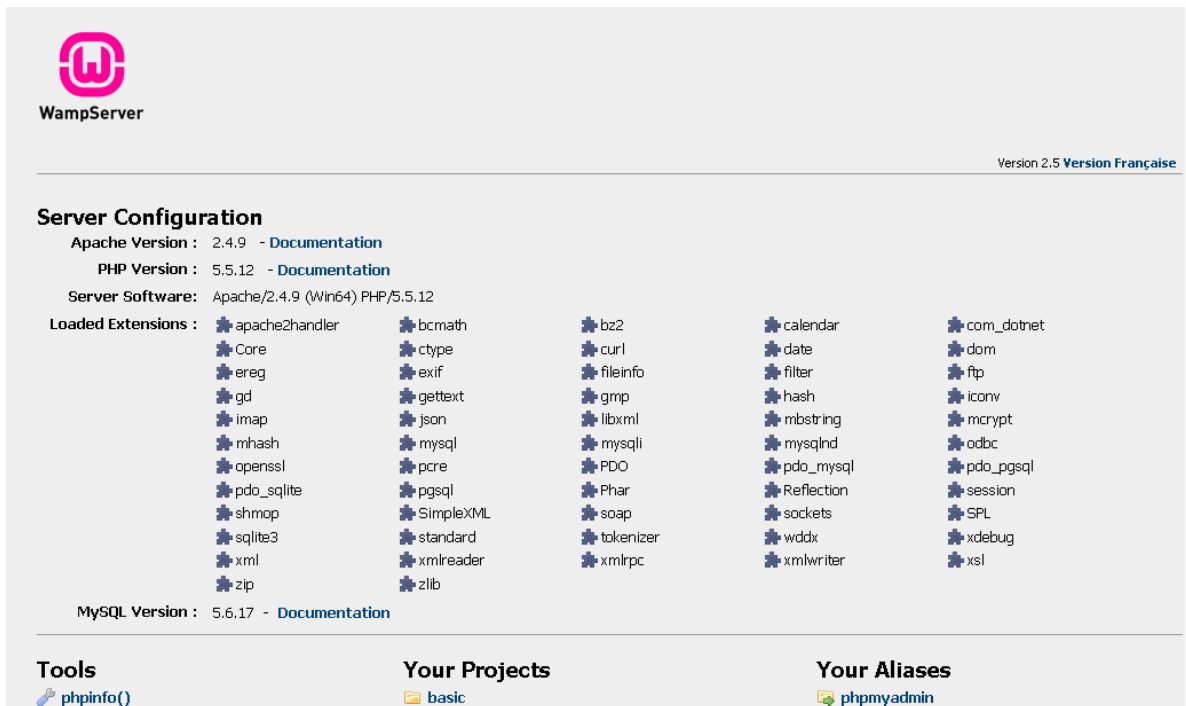


Figura 66: Página de bienvenida de WampServer

Para comenzar, lo primero es ubicar el proyecto BotPao en el directorio que se conoce con el nombre “DocumentRoot”. Por defecto, está en “c:/wamp/www“, que es la carpeta a la que WampServer puede acceder para visualizar las webs en el navegador. Al seleccionar el proyecto aparecerá la siguiente interfaz gráfica que se muestra en la Figura 67.

String de Búsqueda:

Bases de datos a utilizar :	Donde Buscar :	Escoger Rango de Fecha:
<input checked="" type="checkbox"/> ACM	<input checked="" type="checkbox"/> Título	Desde <input type="text" value="marzo"/> <input type="text" value="2014"/>
<input type="checkbox"/> IEEEExplore	<input type="checkbox"/> Abstract	Hasta <input type="text" value="diciembre"/> <input type="text" value="2016"/>
<input type="checkbox"/> Science Direct	<input type="checkbox"/> Palabras Claves	
<input type="checkbox"/> SpringerLink		
<input type="checkbox"/> Scopus		
<input checked="" type="checkbox"/> Taylor and Francis		
<input type="checkbox"/> Emerald		
<input checked="" type="checkbox"/> Wiley and Sons		

Figura 67: Interfaz Gráfica de la herramienta computacional.

En la Figura 68 el usuario deberá introducir la palabra o frase en el cuadro de búsqueda situado en la parte superior de la página, si el usuario no ingresa ninguna cadena de búsqueda en el cuadro y selecciona el botón “Buscar” se desplegará el mensaje de alerta que se muestra en la Figura 69. Si el usuario ingresa una cadena de búsqueda inválida se desplegará el mensaje de alerta como se muestra en la Figura 70.

String de Búsqueda: ["view" or "mapping study") and ("software product line")]

Figura 68: Ingreso de la cadena de búsqueda.

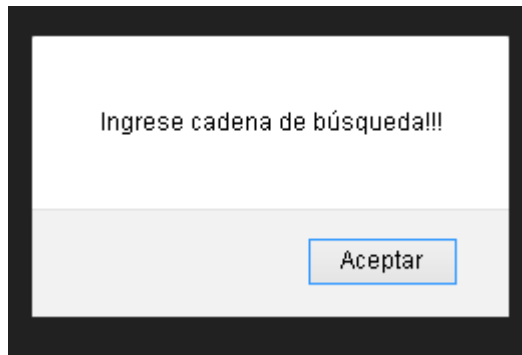


Figura 69: Mensaje de alerta si no ingresa cadena de búsqueda.

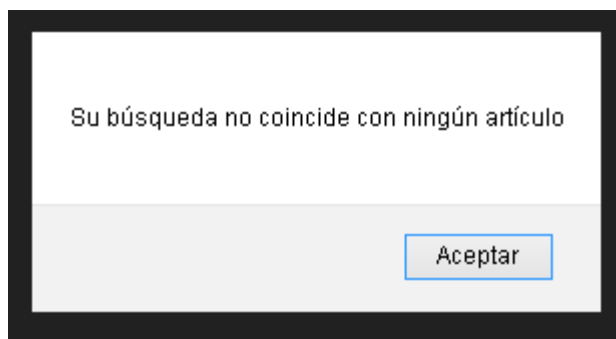
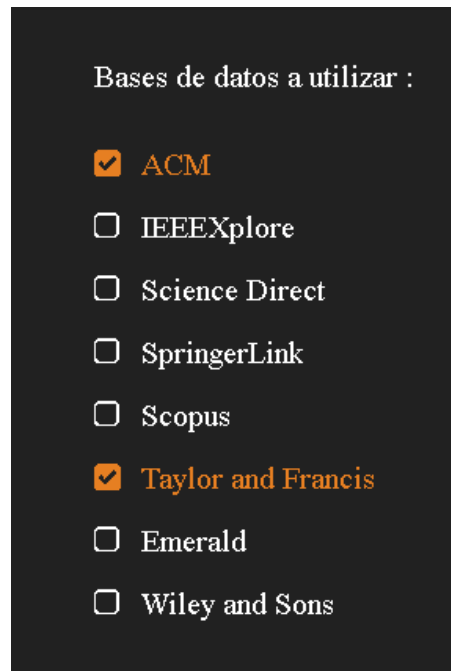


Figura 70: Mensaje de alerta si no se encuentran resultados de la búsqueda.

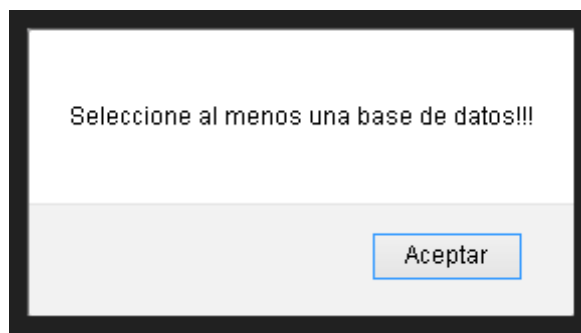
En la Figura 71 se observa las bases de datos en las cuales se va a extraer las referencias bibliográficas según la cadena de búsqueda que introdujo el usuario, en donde se deberá seleccionar al menos una de las opciones para realizar la búsqueda. Si el usuario no selecciona ninguna de las bases de datos para la búsqueda, la herramienta construida desplegará el siguiente mensaje de alerta que se muestra en la Figura 72.



Bases de datos a utilizar :

- ACM
- IEEEExplore
- Science Direct
- SpringerLink
- Scopus
- Taylor and Francis
- Emerald
- Wiley and Sons

Figura 71: Seleccionar bases de datos.



Seleccione al menos una base de datos!!!

Aceptar

Figura 72: Mensaje de alerta bases de datos.

En la Figura 73 se observa los campos definidos en donde el usuario podrá refinar los resultados de la búsqueda, seleccionando al menos uno de los filtros. Si el usuario no selecciona ninguna los campos (Título, Abstract y Palabra clave), la herramienta construida desplegará el siguiente mensaje de alerta que se muestra en la Figura 74.

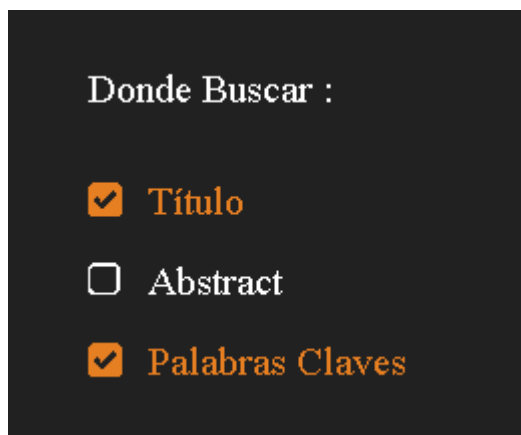


Figura 73: Seleccionar Campos.

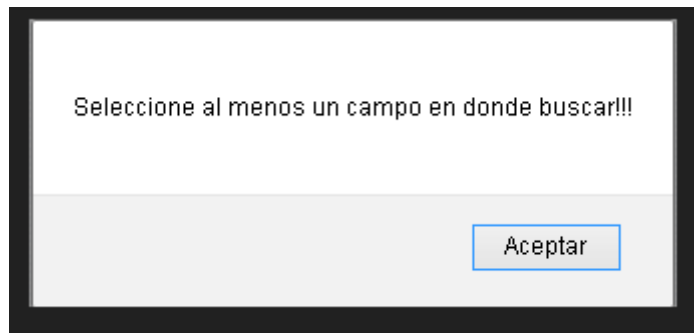


Figura 74: Mensaje de alerta para los campos.

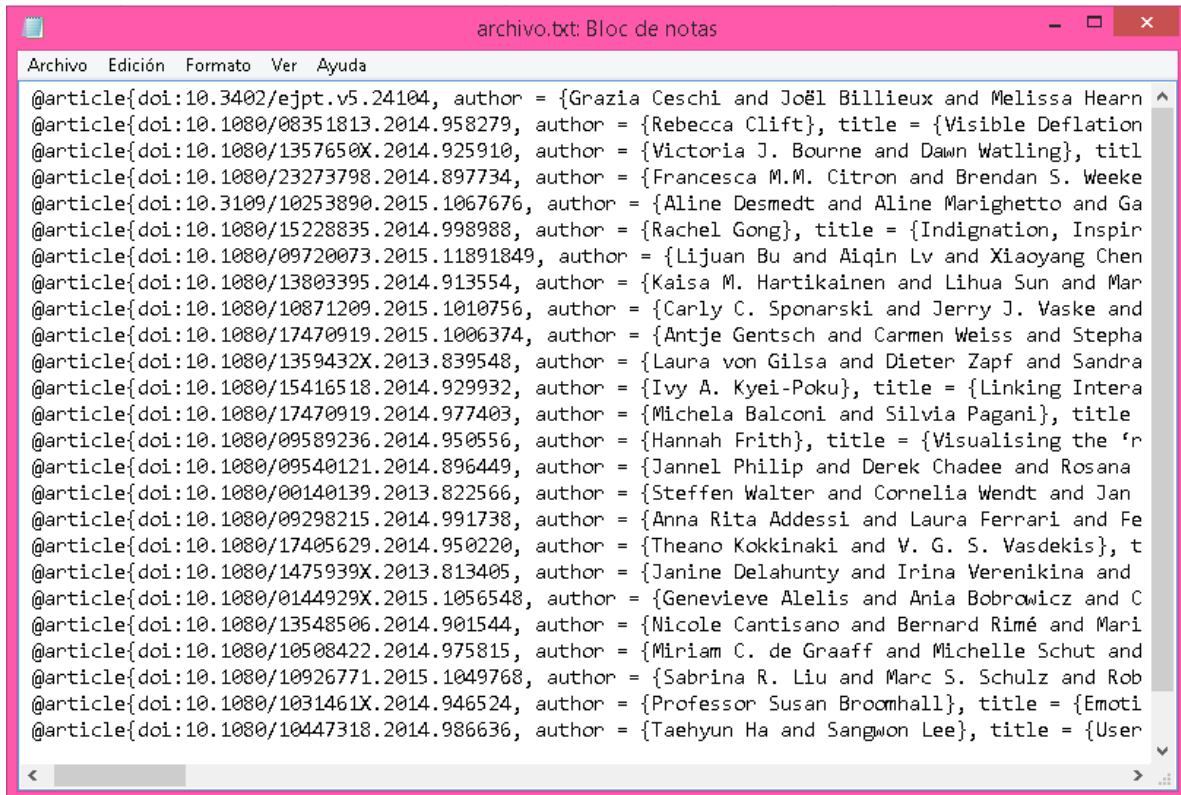
El usuario podrá refinar la búsqueda eligiendo un rango de fechas en el que se quiera buscar.

En la Figura 75 se tiene un rango en donde se podrá elegir la fecha inicio (mes y año) y la fecha término (mes y año).

A dark gray form titled "Escoger Rango de Fecha:". It contains two rows of input fields. The first row is labeled "Desde" and has two dropdown menus: the first is set to "mes" and the second is set to "Año". The second row is labeled "Hasta" and also has two dropdown menus: the first is set to "mes" and the second is set to "Año".

Figura 75: Seleccionar rango de Fechas.

Al presionar el botón buscar, el sistema procederá a hacer la búsqueda conectándose con las bases de datos correspondiente que haya seleccionado el usuario, para que luego extraiga los metadatos de la búsqueda y los guarde en un archivo como se muestra en la Figura 76.



```
archivo.txt: Bloc de notas
Archivo Edición Formato Ver Ayuda
@article{doi:10.3402/ejpt.v5.24104, author = {Grazia Ceschi and Joël Billieux and Melissa Hearn
@article{doi:10.1080/08351813.2014.958279, author = {Rebecca Clift}, title = {Visible Deflation
@article{doi:10.1080/1357650X.2014.925910, author = {Victoria J. Bourne and Dawn Watling}, titl
@article{doi:10.1080/23273798.2014.897734, author = {Francesca M.M. Citron and Brendan S. Weeke
@article{doi:10.3109/10253890.2015.1067676, author = {Aline Desmedt and Aline Marighetto and Ga
@article{doi:10.1080/15228835.2014.998988, author = {Rachel Gong}, title = {Indignation, Inspir
@article{doi:10.1080/09720073.2015.11891849, author = {Lijuan Bu and Aiqin Lv and Xiaoyang Chen
@article{doi:10.1080/13803395.2014.913554, author = {Kaisa M. Hartikainen and Lihua Sun and Mar
@article{doi:10.1080/10871209.2015.1010756, author = {Carly C. Sponarski and Jerry J. Vaske and
@article{doi:10.1080/17470919.2015.1006374, author = {Antje Gentsch and Carmen Weiss and Stepha
@article{doi:10.1080/1359432X.2013.839548, author = {Laura von Gilsa and Dieter Zapf and Sandra
@article{doi:10.1080/15416518.2014.929932, author = {Ivy A. Kyei-Poku}, title = {Linking Intera
@article{doi:10.1080/17470919.2014.977403, author = {Michela Balconi and Silvia Pagani}, title
@article{doi:10.1080/09589236.2014.950556, author = {Hannah Frith}, title = {Visualising the 'r
@article{doi:10.1080/09540121.2014.896449, author = {Jannel Philip and Derek Chadee and Rosana
@article{doi:10.1080/00140139.2013.822566, author = {Steffen Walter and Cornelia Wendt and Jan
@article{doi:10.1080/09298215.2014.991738, author = {Anna Rita Addessi and Laura Ferrari and Fe
@article{doi:10.1080/17405629.2014.950220, author = {Theano Kokkinaki and V. G. S. Vasdekis}, t
@article{doi:10.1080/1475939X.2013.813405, author = {Janine Delahunty and Irina Verenikina and
@article{doi:10.1080/0144929X.2015.1056548, author = {Genevieve Alelis and Ania Bobrowicz and C
@article{doi:10.1080/13548506.2014.901544, author = {Nicole Cantisano and Bernard Rimé and Mari
@article{doi:10.1080/10508422.2014.975815, author = {Miriam C. de Graaff and Michelle Schut and
@article{doi:10.1080/10926771.2015.1049768, author = {Sabrina R. Liu and Marc S. Schulz and Rob
@article{doi:10.1080/1031461X.2014.946524, author = {Professor Susan Broomhall}, title = {Emoti
@article{doi:10.1080/10447318.2014.986636, author = {Taehyun Ha and Sangwon Lee}, title = {User
```

Figura 76: Resultados Guardados en un Archivo.

