



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

Repositorio Abierto de Académicos del Área de Ciencias de la
Computación de Chile

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL EN COMPUTACIÓN

Ignacio Francisco Cuellar Tapia

PROFESOR GUÍA:

Sergio Ochoa Delorenzi

MIEMBROS DE LA COMISIÓN:

Nelson Baloian Tataryan

Fabián Mosso Chávez

SANTIAGO DE CHILE

2023

Resumen

Hay diversas razones por las que es importante contar con un repositorio público de académicos e investigadores; especialmente para las instituciones universitarias; por ejemplo, para formar comités examinadores de tesis o para encontrar potenciales colaboradores para realizar trabajos en conjunto o presentar un proyecto. Aunque hay muchos repositorios de académicos, estos no son abiertos, o bien no permiten búsquedas flexibles, por lo que es difícil poder usarlos para apoyar actividades como las antes mencionadas.

Para contribuir con una potencial solución al problema planteado, esta memoria implementó un repositorio abierto de académicos e investigadores en el área de Ciencias de la Computación, que trabajan en instituciones de educación superior en Chile. El repositorio clasifica a las personas por áreas temáticas, con información homogeneizada y de fácil acceso, que permite a los usuarios consultores (por ejemplo, los coordinadores de programas de pre y postgrado) realizar consultas según diferentes criterios preestablecidos. Este repositorio puede accederse vía una API y a través de un portal Web.

Los mecanismos de alimentación de información son tanto manuales como automáticos. La alimentación manual es realizada por los usuarios administradores de institución y del sistema, y la alimentación automática se lleva a cabo a través de procesos autónomos, encargados de recuperar información desde fuentes de información externas.

El resultado obtenido es un producto mínimo viable desarrollado principalmente en Django (python). Éste usa Django templates, junto con Javascript y Bootstrap para el frontend, y Django y PostgreSQL para el backend. La usabilidad y utilidad del sistema obtenido fue evaluado por usuarios finales, que tenían la necesidad de encontrar académicos/investigadores para apoyar las actividades antes mencionadas. De acuerdo con su opinión el sistema es usable y útil, por lo tanto, podemos decir que se cumplieron los objetivos de la memoria.

Como trabajo a futuro se pueden mencionar varios aspectos de mejora de este sistema; por ejemplo, el refinamiento del sistema de búsqueda, la implementación de notificaciones en caso de cambios, la implementación de nuevos ETLs que permitan la carga automática de más datos, y algunas mejoras en la interfaz que permitan adaptar más rápido a usuarios nuevos.

Tabla de Contenido

1. Introducción	1
1.1. Escenario Actual	2
1.2. Objetivos de la Memoria	5
1.3. Estructura del Documento	6
2. Marco Teórico	7
2.1. Lista de Procesos a Apoyar	7
2.2. Análisis de Potenciales Fuentes de Información de Académicos	8
2.1.1 ORCID	8
2.1.2 DBLP	8
2.1.3 ArnetMiner	9
2.1.4 BibKG: Knowledge Graph of Computer Science Bibliography	9
2.1.5 Páginas Web de los Departamentos de Computación o Informática	9
2.1.6 Vincula	10
2.1.7 Researchgate	10
2.1.8 Mendeley	10
2.1.9 Google Scholar	10
2.1.10 LinkedIn	10
2.1.11 Tabla Comparativa	11
3. Concepción de la Solución	12
3.1. Principales Requisitos de la Solución	12
3.2. Perfiles de Usuario Soportados	12
3.3. Arquitectura del Sistema	13
3.3.1. Carga y actualización de instituciones y unidades	13
3.3.2. Carga y actualización de académicos	14
3.3.3. Carga y actualización de investigadores	14
3.3.4. Consulta de académicos	15
3.3.5. Ambiente operacional del sistema	15
3.4. Modelo de Datos de la Solución	15
4. Implementación de la Solución	19
4.1 Carga de datos	19
Paso 1. Creación de Institución y asignación de páginas web externas	19
Paso 2. Limpieza y agrupación de datos	20
Paso 3. Filtro y aceptación de grupos	20
Paso 4. Selección de académicos por coautor	21
Paso 5. Selección manual de académicos	21
Paso 6. Extracción de datos en DBLP y Aminer	22
4.2. Descripción de la API	22

4.2.1. Base URL	22
4.2.2 Búsqueda básica por nombre	22
4.2.3 Búsqueda avanzada con filtros y keywords	24
4.2.4 Obtención de Académico por ID	25
4.2.5 Obtención de Institución y Unidades con Académicos	27
4.2.5 Obtención de Áreas de Trabajo	28
4.2.6 Obtención de Subáreas	29
4.3. Interfaces de Usuario	30
4.3.1 Página de inicio	30
4.3.2 Servicio de búsqueda	31
4.3.3 Instituciones	32
4.3.4 Perfil de una institución	33
4.3.5 Perfil de un académico	34
5. Evaluación del Sistema	36
5.1. Descripción de los Evaluadores	36
5.2. Descripción de la evaluación basada en System Usability Scale	36
5.3. Descripción de la evaluación basada en Technology Adoption Model	37
5.4. Resultados Obtenidos	38
5.4.1 Resultados de Usabilidad	38
5.4.2 Resultados de Utilidad Percibida	39
5.5. Discusión de los Resultados	39
5.5.1. Análisis de la Usabilidad	40
5.5.2. Análisis de la Utilidad	40
5.5.3 Comentarios extras de evaluadores	40
6. Conclusiones y Trabajo a Futuro	41
7. Bibliografía	43
Anexo: Académicos de Ciencias de la Computación en Chile	44

1. Introducción

Hay un sin número de razones por las cuales es importante, particularmente para las instituciones educativas y científicas, poder identificar rápidamente a los académicos/investigadores que trabajan en áreas de estudio particulares; por ejemplo, en algoritmos dentro del área de Ciencias de la Computación. Las razones van desde la identificación de candidatos para revisar un informe de práctica profesional o dictar un curso, hasta la búsqueda de investigadores para postular a una red de colaboración científica en una cierta área de especialización, o apoyar técnicamente la definición de políticas públicas.

Dependiendo de la actividad que se quiera llevar a cabo, los académicos usualmente deben cumplir con aspectos reglamentarios; por ejemplo, tener el grado de magíster o doctorado según sea el caso. Dichos aspectos (requisitos) pueden tener un alcance institucional, nacional, o inclusive internacional en el caso de convocatorias internacionales.

Dada la gran variedad de actividades a cubrir, y a los aspectos reglamentarios asociados a éstas, los académicos deben ser caracterizados considerando múltiples variables. Por lo tanto, el perfilamiento de estas personas es uno de los desafíos que se debe abordar, así como también el mantenimiento evolutivo de dichos perfiles.

Para ilustrar mejor la necesidad de apoyar estas actividades, consideremos el caso del Departamento de Ciencias de la Computación (DCC) de la Universidad de Chile. En los programas de postgrado que imparte el DCC (Magíster en Ciencias y en TI, y el Doctorado en Computación), los estudiantes deben generar propuestas y documentos de tesis que piden dichos programas.

Algunos de esos informes deben ser evaluados por varios académicos; algunos de los cuales son parte del Departamento (evaluadores internos), y otros reglamentariamente deben ser parte de otras universidades chilenas o extranjeras (evaluadores externos), según sea el caso. Dependiendo del documento a revisar, estos académicos deben cumplir con diversas condiciones, las cuales están principalmente asociadas a las áreas de conocimiento y a los estudios previos de estos evaluadores. Por ejemplo, una propuesta de tesis doctoral sólo puede ser evaluada por académicos que poseen el grado de doctor.

Las personas encargadas de encontrar evaluadores para estos documentos, usualmente los coordinadores de los programas respectivos deben buscar por sus propios medios a los académicos que cumplan con las condiciones establecidas para desempeñar dicho rol. Esta búsqueda se hace siguiendo estrategias ad hoc, debido a la falta de un

repositorio público y unificado que apoye esta labor. Por ejemplo, se busca en grupos de trabajo conocidos, se consulta a profesores expertos en el área por contactos que pudieran realizar la labor, o también se revisan los sitios Web de otras universidades donde podría haber candidatos.

Esta búsqueda manual involucra un esfuerzo importante para los coordinadores de programas, y además suele ser injusta para los evaluadores (tanto internos, como externos), pues aquellos que son más visibles o reconocidos suelen ser a los que más se les pide colaborar con estas evaluaciones. Vale destacar que los académicos no reciben ninguna retribución por esta labor.

Además, seleccionar frecuentemente a profesores de un mismo grupo, conlleva a una sobrecarga de trabajo para esas personas, pudiendo llegar a infortunios, tanto para la persona que es solicitada, como para quien solicita su cooperación. Además, se pierde la oportunidad de vincular a académicos jóvenes (usualmente poco conocidos), quienes podrían llevar a cabo esta tarea y generar relaciones de colaboración con grupos de otras universidades.

La situación descrita es sólo una de las que se da en este escenario, pues también ocurre cuando se buscan revisores para proyectos de investigación o de extensión (por ejemplo, de ANID, CORFO, etc.), o cuando investigadores de una institución buscan colaborar con investigadores de su área dentro o fuera de Chile.

Por otra parte, es importante destacar que la ciencia de la computación se ha transformado en una disciplina transversal que apoya a muchas otras disciplinas, por ejemplo, astronomía, medicina y las ingenierías en general. Por lo tanto, hay investigadores de ciencias de la computación trabajando en centros de investigación y departamentos que no son particularmente de computación o informática, y también lo hacen en centros o iniciativas multidisciplinarias. Identificar a esos investigadores, que podrían apoyar las actividades antes mencionadas es aún más difícil, puesto que son menos visibles para la comunidad. Finalmente, se espera poder escalar en el futuro los servicios de esta plataforma a otros países de la región, y por lo tanto, incluir también a investigadores Latinoamericanos.

1.1. Escenario Actual

En la actualidad no hay un registro centralizado y abierto de los académicos del área de Ciencias de la Computación e Informática de Chile, el cual pudiera facilitar las actividades antes descritas. Están por un lado las plataformas globales de registro de académicos,

como por ejemplo, Google Scholar¹ o Researchgate², que no son plataformas abiertas, y donde es muy difícil buscar por región y área de investigación, dado que están focalizadas en los trabajos de investigación de las personas. Por lo tanto, no son las plataformas más apropiadas para apoyar las labores antes planteadas.

Por otra parte, está la iniciativa Vincula.cl³, conformada por un grupo de universidades chilenas, que busca vincular a las unidades que hacen ciencia, con aquellas organizaciones que deberían utilizarla para apoyar los procesos de toma de decisiones a nivel país, o la definición de políticas públicas (por ejemplo, las cámaras de diputados y de senadores). Lamentablemente, esta plataforma incluye un número reducido de universidades chilenas, donde al momento de la escritura de este documento, no se incluyen universidades como la Universidad de Chile, Universidad Adolfo Ibañez, Universidad Diego Portales, etc., tal como se muestra en la Figura 1.

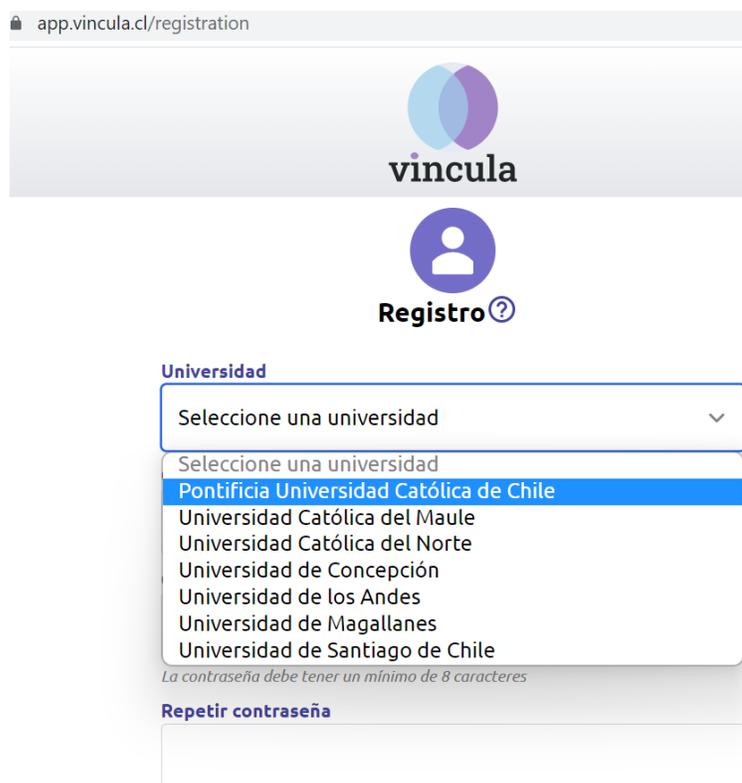


Figura 1. Formulario de Registro de académicos a la Plataforma Vincula.cl.

Respecto a la iniciativa Vincula.cl, se desconoce qué información mantiene de los investigadores (no se explicita en el sitio web), por lo tanto, tampoco se sabe si dicha

¹ <https://scholar.google.com/>

² <https://www.researchgate.net/>

³ www.vincula.cl

información cumple con lo que se necesita para realizar labores como las identificadas en esta memoria.

Por otra parte, en las páginas Web de los Departamentos de Computación o Informática de las Universidades usualmente se encuentran listas de académicos, que eventualmente podrían utilizarse como fuente de información para crear un repositorio unificado. En el Anexo I se muestra una lista de sitios Web de estos Departamentos, y un ejemplo de la información que se expone acerca de estas personas. Como se puede apreciar allí, la información está desperdigada en múltiples sitios web, y en muchos casos desactualizada o con muy poco detalle, como para que alguien pueda utilizarla de forma fácil para identificar candidatos que puedan apoyar algunas de las actividades antes mencionadas.

La Figura 2 muestra un diagrama del ambiente operacional en el que operan los consultores de este tipo de información. Allí se puede ver que los usuarios consultores (por ejemplo, los coordinadores de programas de pre y postgrado) utilizan la Web para acceder a las páginas institucionales, para tratar de identificar académicos candidatos a ser invitados a realizar alguna de las actividades antes descritas. En algunos casos, los académicos cuentan con una página Web personal con más información respecto a él/ella y a sus áreas de interés. Típicamente, hay personal que actualiza la información de los sitios web institucionales, y los mismos académicos son los que actualizan sus páginas Web personales.

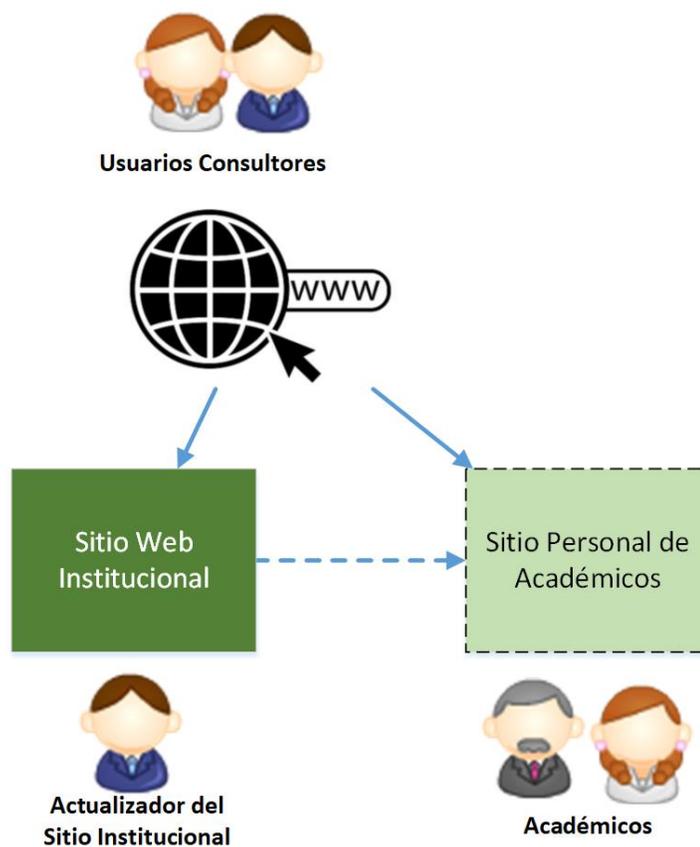


Figura 2. Ambiente operacional del escenario actual.

Debido a lo antes mencionado, surge la oportunidad de crear un repositorio centralizado de académicos, por áreas temáticas, con información homogeneizada y de fácil acceso para los consultores. Este repositorio será de acceso público para quienes necesiten consultarlo, y será actualizado con información de múltiples fuentes; algunos de estos mecanismos de alimentación serán manuales y otros automáticos.

1.2. Objetivos de la Memoria

El objetivo general de este trabajo de memoria consiste en implementar un repositorio de información de académicos por áreas temáticas, que permita al público en general (pero principalmente a instituciones académicas y de investigación) consultarlo con múltiples propósitos. Los objetivos específicos que se desprenden del objetivo general son los siguientes:

1. Definir un template de perfiles de académicos y áreas de expertise, que permita la homogeneización de la información, y les permita a los académicos indicar qué información desean mantener pública y cuál privada.

2. Definir diversos mecanismos (complementarios) para alimentar, perfilar y mantener actualizados los perfiles de los académicos.
3. Aumentar la visibilidad de académicos de diferentes áreas a nivel nacional. Añadiendo académicos de diferentes universidades/instituciones en un solo repositorio de acceso público.
4. Facilitar el proceso de búsqueda de académicos por parte de instituciones académicas y de investigación. Esto mediante mecanismos de consulta de información bajo criterios preestablecidos, y también notificaciones automáticas sobre académicos que trabajan en áreas particulares.
5. Implementar una API que permita el acceso programático a la información de los académicos.

1.3. Estructura del Documento

El presente documento se encuentra estructurado en 6 capítulos. En el capítulo 2 se presenta el marco teórico que permite entender el trabajo realizado en esta memoria, donde destacan los diferentes procesos que se busca apoyar y cuáles son las diferentes fuentes de información que son utilizadas. En el capítulo 3 se presenta la concepción de la solución, incluyendo los principales requisitos del sistema, los perfiles de usuarios soportados, la arquitectura del sistema, el diseño del modelo de datos. En el capítulo 4 se describe la implementación de la solución, donde se detalla cómo se extraen los datos, y cómo acceder a ellos por medio de una API y por medio de una interfaz web. En el capítulo 5 se presenta la evaluación del sistema, en donde se discuten los resultados obtenidos. Finalmente, en el capítulo 6 se presentan las conclusiones y el trabajo a futuro.

2. Marco Teórico

Este capítulo presenta una lista de los principales procesos a apoyar con la aplicación, así como también un análisis de las fuentes de datos (primarias y secundarias) que podrían utilizarse para actualizar la información de los académicos.

2.1. Lista de Procesos a Apoyar

Como se mencionó anteriormente, el sistema implementado busca apoyar y facilitar la realización de diferentes actividades o procesos propios de las instituciones académicas. Estas actividades presentan diferentes naturalezas y requisitos. A continuación, se presentan diferentes procesos (o casos de usos) los cuales se buscan apoyar con el sistema desarrollado en esta memoria:

- Identificación de potenciales evaluadores de informes de práctica/memoria/propuestas de tesis.
- Búsqueda de académicos para formar parte de comisiones de defensa de memorias y tesis.
- Búsqueda de académicos para dictar cursos, seminarios y charlas.
- Búsqueda de revisores para proyectos de investigación o de extensión (ejemplo ANID y CORFO).
- Búsqueda de colaboradores para presentar o llevar a cabo iniciativas de investigación.
- Identificación de autoridades técnicas que puedan apoyar la definición de políticas públicas y la toma de decisión en temas asociados a computación/informática, sistemas de software, Internet, etc.; y también en temas transversales, tales como ciberseguridad y privacidad en ambientes digitales.

Junto con lo anterior, al ser un sistema público y abierto, permite satisfacer las necesidades de personas particulares que requieran la búsqueda o el acceso a la información que se tiene. Se pretende también que este repositorio permita generar estadística por demanda, por ejemplo, por área de interés, por género, por universidad, por país, etc.

2.2. Análisis de Potenciales Fuentes de Información de Académicos

A continuación, se presenta un análisis de las fuentes de información que pueden ser útiles para el trabajo de memoria. Las características que se analizaron de estas fuentes de información fueron las siguientes:

- Existencia de una API⁴ pública: Hace referencia a si la fuente proporciona algún mecanismo para extraer la información mediante una API.
- Límite de resultados: En el caso de que la fuente de datos tuviese API, se analizó si existe un tope máximo de resultados a extraer.
- Formato de los resultados: Se refiere al formato en que la API retorna los datos solicitados. Por ejemplo, estos resultados podrían entregarse en formato JSON o XML.
- Capacidad de identificar investigadores chilenos: Indica si la fuente de datos permite saber el país al que pertenece la institución en la que trabaja el académico.

A continuación, se analizan brevemente las distintas fuentes de datos analizadas.

2.1.1 ORCID

ORCID⁵ es una plataforma que proporciona un identificador único y persistente, al cual está asociado un investigador. Esta plataforma almacena datos como afiliaciones profesionales, becas, publicaciones, redes sociales, página web personal, etc. Si bien esta fuente cuenta con una amplia gama de información, esta suele ser incompleta para la mayoría de los investigadores (por ejemplo, no incluir el país donde trabaja el investigador). Toda la información que presenta es relevante para el trabajo de memoria. Esta aplicación cuenta además con una API pública capaz de entregar los datos en formato JSON.

2.1.2 DBLP

DBLP⁶ es una plataforma que registra principalmente investigadores del área de Ciencias de la Computación, y sus publicaciones. Ésta proporciona información abierta y cuenta con una API pública que permite realizar búsquedas sobre publicaciones y autores según su id. Una vez obtenido un id de un autor, es posible extraer sus publicaciones en formato XML. Esta plataforma tiene como enfoque principal almacenar los trabajos investigativos

⁴ API: Application Programming Interface

⁵ <https://orcid.org/>

⁶ <https://dblp.org/>

y los coautores de los investigadores, por lo que no incluye (o rara vez lo hace) información clave de los investigadores como las áreas de investigación, páginas web, correos electrónicos, afiliaciones, el país donde se trabaja, etc.

2.1.3 ArnetMiner

ArnetMiner⁷ (o Aminer) es una aplicación que provee un servicio de búsqueda y minado para las redes de colaboración con otros investigadores. Ésta cuenta con una API pública que permite buscar un investigador y sus redes. Gracias a lo anterior, es posible hacer un nexo entre investigadores de una misma universidad/institución, por ejemplo, para tratarlos como un conjunto. Aminer proporciona tags o keywords generados a partir de los trabajos investigativos, que permiten realizar una agrupación de estos en diferentes tópicos no estandarizados. Junto con lo anterior, esta plataforma cuenta con datos como los correos electrónicos y las páginas web personales de cada investigador.

2.1.4 BibKG: Knowledge Graph of Computer Science Bibliography

BibKG⁸ es una biblioteca digital, cuyo objetivo es recolectar información de publicaciones y autores desde diferentes fuentes, como DBLP y ArnetMiner, la cual se integra en una base de datos MillenniumDB. De esta biblioteca se pueden obtener los trabajos realizados por distintos investigadores, así como también el identificador único que estas personas tienen en otras fuentes de información, como por ejemplo, DBLP y ORCID. BibKG se encuentra actualmente en desarrollo, por lo que no es posible considerarla aún como una fuente de información estable para apoyar este trabajo de memoria. Aunque se desconoce la existencia de una API pública para BibKG, ésta es una iniciativa muy interesante a la que se continuará haciendo seguimiento.

2.1.5 Páginas Web de los Departamentos de Computación o Informática

La fuente de información más confiable suele ser aquella expuesta por las mismas universidades. Sin embargo, tal como se puede apreciar en los diferentes ejemplos del Anexo I, esta información carece de detalle y el formato en el que se presenta no está estandarizado, llegando a no ser útil en la mayoría de los casos. Ninguna de las páginas listadas en el Anexo cuenta con una API pública que permita extraer información de forma sistemática. Debido a eso, hacer web scraping aparenta ser la mejor alternativa de extracción automática desde estas páginas web; por lo tanto, quedan descartadas como fuentes de información primaria. Es decir, se pueden utilizar como fuente de información secundaria (para completar datos de investigadores ya identificados), pero no para

⁷ <https://www.aminer.org/>

⁸ <https://bibkg.imfd.cl/>

identificar académicos, pues en dichas fuentes de datos usualmente se combina información de investigadores, autoridades y funcionarios, entre otros.

2.1.6 Vincula

Vincula⁹ es una aplicación que permite la comunicación entre productores de conocimiento (investigadores), y tomadores de decisiones (legisladores). Su objetivo es fortalecer una cultura de la evidencia, contribuyendo a la calidad y legitimidad del proceso de formación de la ley, y reduciendo las brechas de género y centralismo existentes. Vincula es una plataforma cerrada para los miembros mencionados anteriormente, por lo que no es posible acceder a sus datos.

2.1.7 Researchgate

Researchgate¹⁰ es una red social para científicos e investigadores que permite compartir papers, realizar preguntas y encontrar colaboradores. Esta aplicación no cuenta con una api pública que permita extraer información, por lo que a priori queda descartada como fuente de información primaria. Entre los datos que proporciona está la institución con la que se encuentra afiliada el investigador, su número de publicaciones y un listado con sus habilidades o experiencia académica.

2.1.8 Mendeley

Mendeley¹¹ es una aplicación web y de escritorio, que permite gestionar y compartir referencias bibliográficas y documentos de investigación. En el año 2020, Mendeley tomó la decisión de eliminar sus feature de perfilado de investigadores, por lo que solo cuenta con los trabajos investigativos de los mismos, dado lo anterior queda descartada como posible fuente de información primaria.

2.1.9 Google Scholar

Google Scholar¹² es un motor de búsqueda que permite encontrar literatura académica. Típicamente los investigadores cuentan con un perfil, junto con su identificador único, su institución asociada, y las áreas de investigación. Esta aplicación web no cuenta con una api pública, por lo que no es factible usarla como fuente de información primaria.

2.1.10 LinkedIn

⁹ <https://vincula.cl/>

¹⁰ <https://www.researchgate.net/>

¹¹ <https://www.mendeley.com/>

¹² <https://scholar.google.com/>

Linkedin¹³ es una red social para profesionales, enfocada en brindar visibilidad a las carreras de estas personas con un objetivo laboral. Esta aplicación permite a los usuarios la creación de su propio perfil, donde uno puede incluir toda su información laboral, educacional, lugar de trabajo, etc. Dentro de esta red se encuentran también los investigadores que son foco de este trabajo. Linkedin no cuenta con una api pública, por lo que queda descartada como posible fuente de información primaria.

2.1.11 Tabla Comparativa

Para sintetizar lo presentado anteriormente, a continuación, se muestra una tabla comparativa con las diferentes fuentes de información analizadas.

Tabla 1. Resumen de información de las diferentes fuentes de información.

Fuente	API pública	Límite de Resultados	Formato de la Información	Identificación de País
ORCID	Sí	máx 10000	JSON	No
DBLP	Sí	máx 1000	JSON/XML	No
ArnetMiner	Sí	máx 100	JSON	No
BibKG	No	N/A	N/A	No
Páginas de Universidades	No	N/A	N/A	Si
Vincula	No	N/A	N/A	Si*
Researchgate	No	N/A	N/A	No
Mendeley	No	N/A	N/A	No
Google Scholar	No	N/A	N/A	No
Linkedin	No	N/A	N/A	Si**

Si*: Brinda acceso, sólo por invitación.

Si**: Brinda acceso, pero de pago.

N/A: No aplicable, no disponible o sin respuesta.

¹³ <https://cl.linkedin.com/>

3. Concepción de la Solución

En este capítulo se presenta una síntesis de los principales requisitos de la solución, un listado de los perfiles de usuarios soportados, la arquitectura del sistema y el modelo de datos.

3.1. Principales Requisitos de la Solución

Teniendo en cuenta lo presentado en los capítulos anteriores, la plataforma web desarrollada debía cumplir con los siguientes requisitos para ser considerado un producto mínimo viable (MVP):

- El sistema debe permitir la alimentación de datos, tanto de forma automática como manual.
- El sistema debe mantener la información actualizada. Ya sea bajo demanda o con revisiones periódicas.
- Los datos obtenidos deben ser completos, permitiendo llevar a cabo los procesos mencionados en el capítulo 2 de forma correcta.
- Los datos obtenidos deben provenir de una fuente confiable, como también debe dar confianza al usuario final.
- El sistema debe extraer datos de APIs estables (que perduren en el tiempo).
- El sistema debe tener la capacidad de extraer datos públicos de las diferentes instituciones.
- El sistema debe permitir la actualización de datos de forma manual para la corrección de errores o aportar a la completitud de esta.
- Se debe generar una correlación de académicos, que permita agilizar la búsqueda de estos.

3.2. Perfiles de Usuario Soportados

Dada la naturaleza de la aplicación, donde las personas realizan búsquedas según sus necesidades sin la necesidad de estar registradas, los principales usuarios soportados son usuarios anónimos (consultores). Sin embargo, al dar la posibilidad de que un usuario

pueda modificar sus propios datos, se implementaron los siguientes perfiles de usuario adicionales:

- Administrador del sistema: Usuario capaz de crear nuevas instituciones, editarlas y asignarles un administrador de Institución. También tiene la capacidad de modificar las diferentes áreas, subáreas de trabajo y dominios de aplicación.
- Administrador de Institución: Usuario capaz de modificar su institución asignada. Esto incluye la creación y edición de Unidades y Académicos de esa institución.
- Académico: Usuario capaz de modificar sus propios datos. Estos son administrados por los dos tipos de usuarios mencionados anteriormente. Los datos modificados por un académico necesitan ser verificados por otros académicos o por el administrador para que sean aceptados como válidos.

3.3. Arquitectura del Sistema

La Figura 3 muestra la arquitectura de la solución, la cual debe dar soporte a cuatro procesos: 1) carga y actualización de instituciones y unidades, 2) carga y actualización de académicos, 3) carga y actualización de investigadores, y 4) la consulta de académicos por diversos criterios. A continuación, se explica cómo se realiza cada uno de estos procesos.

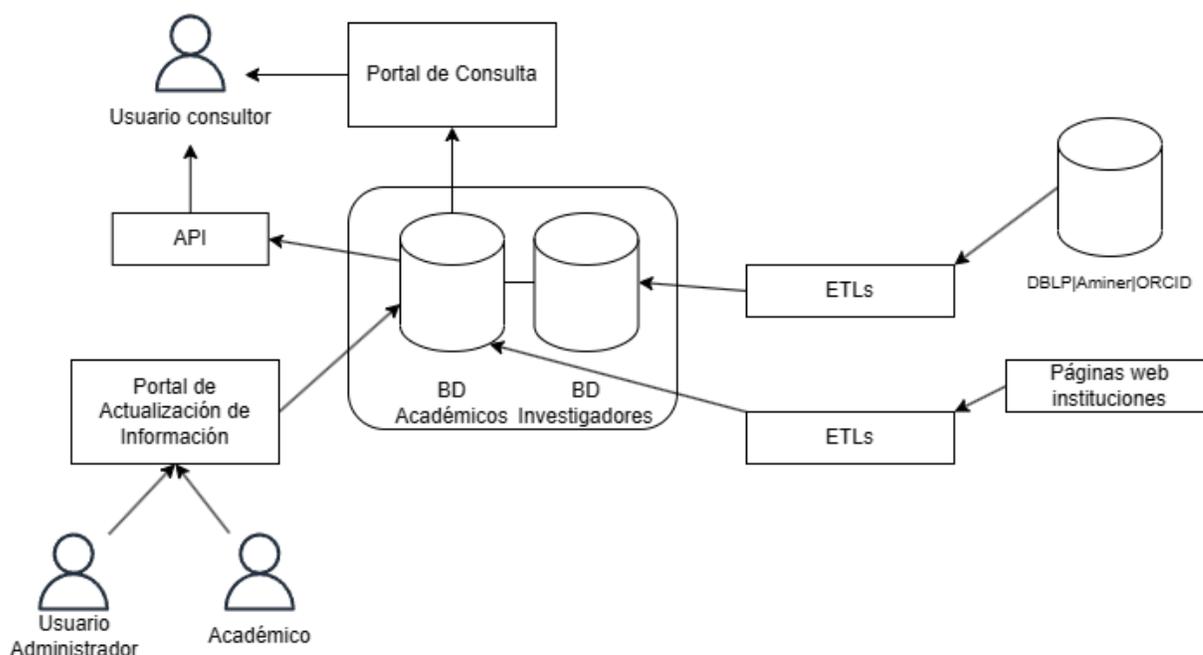


Figura 3. Arquitectura del sistema.

3.3.1. Carga y actualización de instituciones y unidades

El sistema le permite al usuario administrador del sistema cargar instituciones, que corresponde a su nombre, sigla, país de procedencia y página web institucional, y dentro de ellas, unidades con sus respectivas URLs; por ejemplo, en el caso de la institución Universidad de Chile, tiene como unidad a el Departamento de Ciencias de la Computación, donde su URL corresponde a: <https://www.dcc.uchile.cl/>.

Por otra parte, un usuario administrador del sistema puede asignar un "administrador de institución", quien podrá modificar los datos de la institución y la unidad a través del "portal de actualización de información" (Fig. 3). Esta misma labor la pueden realizar los usuarios con rol "administrador del sistema".

3.3.2. Carga y actualización de académicos

Una vez cargada una unidad, el administrador de institución o del sistema puede cargar una o más URLs donde estén reportados los académicos de cada unidad: <https://www.dcc.uchile.cl/nosotros/academicos/>. Este es el primer paso para ingresar un académico/investigador al sistema, luego de haber ingresado la institución y unidad al sistema. Posteriormente, un proceso autónomo identifica a los académicos reportados en dichas páginas web, y los ingresa a la base de datos de académicos. Cada uno queda vinculado a la unidad correspondiente (en este caso, el Departamento de Ciencias de la Computación).

De la misma manera que en el proceso anterior, los usuarios administradores de institución o del sistema pueden cargar o actualizar la información de los académicos.

3.3.3. Carga y actualización de investigadores

Cada académico identificado es un potencial "investigador"; es decir, una persona que hace ciencia en una o más áreas de la computación. Para saber si un académico es además un investigador, el sistema busca automáticamente la información de dicho académico en DBLP, utilizando su nombre como input. Como resultado de esto pueden ocurrir varias cosas:

1. La búsqueda en DBLP arroja 0 resultados, lo que indicaría que el académico no es un investigador, pues no cuenta con publicaciones.
2. La búsqueda arroja sólo un resultado. En ese caso, se asume que el académico corresponde a ese investigador.
3. La búsqueda arroja varios investigadores, como posibles correspondencias con el académico que se tomó como base. En este caso, el académico quedará vinculado

al investigador que tenga mayor número de coautoría con personas de su misma unidad.

Una vez vinculado el perfil del académico con el de un investigador, ETLs buscan automáticamente información en DBLP, Aminer y ORCID, y la cargan en la base de datos de investigadores (Fig. 3). Entre esta información están las áreas en las que la persona trabaja, lo cual es vital para apoyar el proceso de búsqueda de académicos/investigadores.

3.3.4. Consulta de académicos

Las consultas sobre el espacio de información de académicos pueden realizarse a través de dos métodos, tal como se muestra en la Fig. 3: 1) a través del portal de consulta, o 2) a través de la API del sistema. En el primer caso, el usuario ingresa las áreas de procedencia en las que quiere identificar académicos/investigadores, y el sistema le entrega los resultados correspondientes.

En el caso del segundo método, el sistema expone diversos endpoints, que pueden ser utilizados por otros sistemas para hacer búsquedas ad hoc. Por ejemplo, usando estos servicios un sistema de recomendación podría sugerir miembros para conformar un comité de evaluación de una memoria o una tesis.

3.3.5. Ambiente operacional del sistema

El ambiente operacional del sistema cuenta con una base de datos PostgreSQL, la cual es controlada por el backend desarrollado en Django (python). Este backend permite a los usuarios administradores, añadir y actualizar información de los académicos. El backend tiene la tarea de actualizar la base de datos de los investigadores según usuarios consultan sobre sus académicos correspondientes. El backend también cuenta con una API pública que permite la búsqueda y obtención de académicos según diversos filtros.

Los diferentes tipos de usuarios pueden interactuar con el frontend de la aplicación para mantener la información actualizada, o consultar la información de base de datos. El frontend de la aplicación está desarrollado con Django templates, junto con Javascript y Bootstrap. La interacción más relevante que pueden llevar a cabo los usuarios consultores es la búsqueda de académicos, según criterios preestablecidos que facilitan la búsqueda y sin la necesidad de contar con una cuenta.

3.4. Modelo de Datos de la Solución

En las Figuras 4 y 5 se presenta el modelo de datos que permite almacenar toda la información necesaria para llegar a la solución planteada. El modelo de datos está

dividido en dos partes, académicos e investigadores (el segundo es una extensión del primero), según el mecanismo por el cual se cargan los datos. Por un lado, se tienen las tablas que son rellenas de forma manual por un usuario de tipo “administrador de institución”, donde su elemento clave son los académicos (Figura 4). Por el otro lado, están las tablas que son completadas automáticamente por los ETLs, cuyo elemento principal son los investigadores.

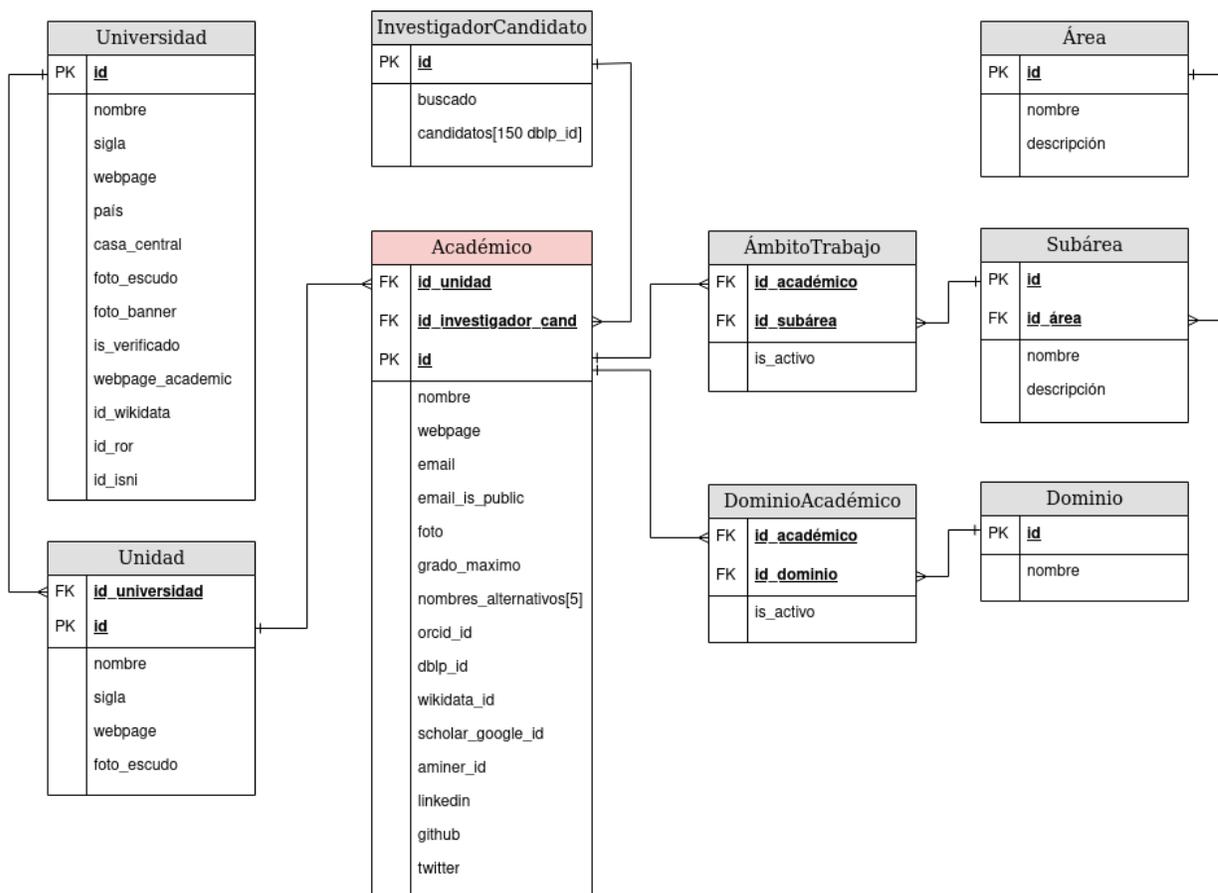


Figura 4. Modelo de datos de académicos.

A continuación, se detalla brevemente cada una de las entidades que forman parte del modelo de datos presentada en la Figura 4.

Universidad: Ésta almacena información pública de universidades. Esta entidad puede ser dividida en unidades para una mejor organización de la información.

Unidad: Subdivisión dentro de una universidad. Éste puede ser un departamento, un instituto/centro específico, o un conjunto de académicos con algún tipo de agrupación formal.

Académico: Esta entidad almacena toda información que sea útil para identificar a un académico y su asociación con una institución y unidad. Éste puede contener

atributos públicos como privados, los cuales son marcados con un flag en caso de ser variables.

Área: Almacena las distintas áreas de trabajo existentes en el marco de la computación. Esta entidad es utilizada principalmente para filtrar a los diferentes académicos, según las áreas en las que ellos trabajan.

Subárea: Almacena las distintas subáreas de un área de trabajo en particular. Al igual que en el caso anterior, ésta es utilizada para filtrar a los diferentes académicos.

ÁmbitoTrabajo: Relaciona a un académico o un investigador, con las distintas subáreas de trabajo definidas.

Dominio: Almacena los dominios de aplicación a los que puede estar asociado un académico. Por ejemplo, un académico de computación puede trabajar en aplicaciones para “transporte” o “medicina”.

DominioAcadémico: Relaciona un académico en específico, con los dominios de aplicación en los que él/ella trabaja. Esta tabla es utilizada principalmente para filtrar a los diferentes académicos.

InvestigadorCandidato: Almacena todos los identificadores de DBLP, ORCID y Aminer que coinciden (por nombre) con un académico en particular. En caso de que ese matching arroje un único candidato, ese investigador será asociado al académico que se utilizó como fuente para la búsqueda. En caso contrario, un usuario administrador (de institución o de sistema) deberá seleccionar el investigador correcto de forma manual.

A continuación, se detalla brevemente cada una de las entidades que forman parte del modelo de datos presentada en la Figura 5.

Investigador: Esta entidad almacena toda información que es posible obtener de un académico por medio de fuentes externas (DBLP, ORCID y Aminer). Éste sólo contiene atributos públicos dada su naturaleza.

CoautorInvestigador: Relaciona dos investigadores que han realizado trabajos de investigación en conjunto. Éste cuenta con un atributo llamado “peso”, que representa el número/cantidad de trabajos publicados en conjunto.

Keyword: Es la metadata utilizada para filtrar académicos. Ésta es generada por fuentes externas a partir de los trabajos de investigación realizados por los investigadores.

KeywordInvestigador: Relaciona las keyword de las publicaciones, con los investigadores correspondientes (autores de las mismas). Esta entidad cuenta con un atributo peso que es el número/cantidad de trabajos de investigación que contienen esa keyword.

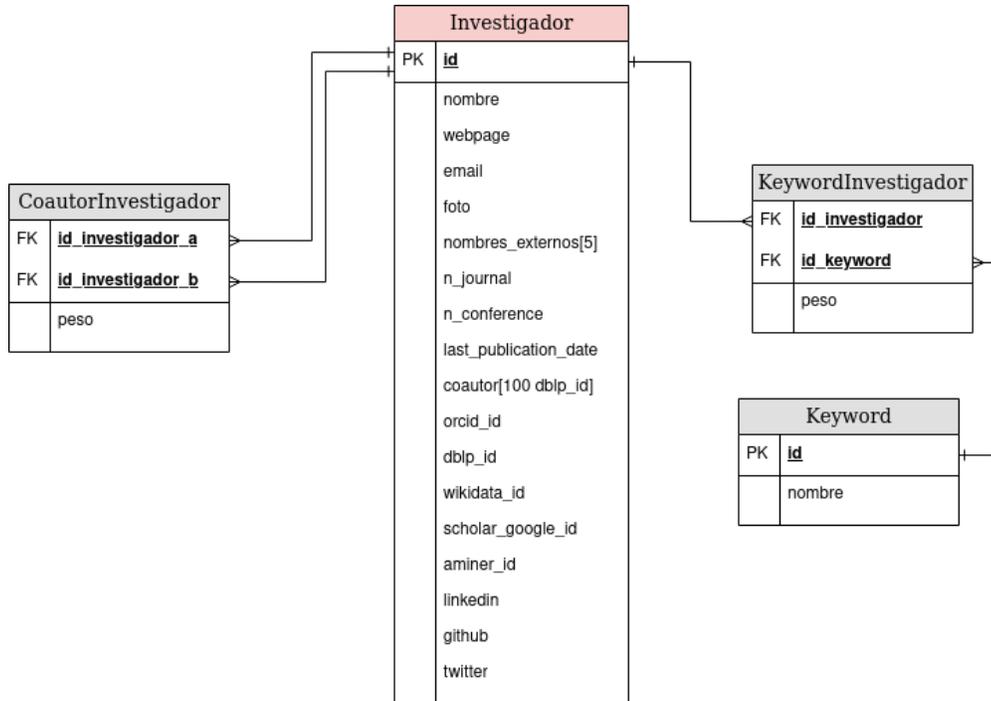


Figura 5. Modelo de datos de investigadores.

4. Implementación de la Solución

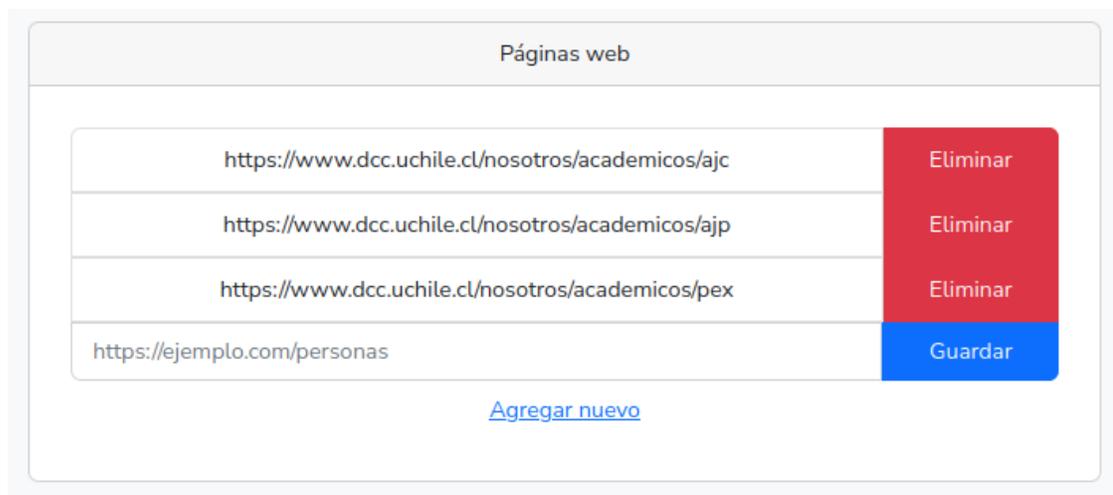
En el presente capítulo se detallan las diferentes secciones que conllevan a la solución del problema planteado. En un principio, se presenta cómo fueron implementados los ETLs (Extract, Transform and Load) que permiten poblar el repositorio de investigadores. Luego, se detalla el mecanismo para extraer estos datos de forma ordenada por medio de una API. Finalmente, se presentan las interfaces de usuarios que permiten consultar estos mismos datos por medio de una página web.

4.1 Carga de datos

Para cargar datos, el sistema lleva a cabo una serie de pasos que permite cargar información de académicos de una institución en específico. Estos pasos son detallados en orden en los siguientes puntos.

Paso 1. Creación de Institución y asignación de páginas web externas

El punto de partida para poblar la aplicación es poder contar con una entidad/institución en específico. Para llevar a cabo la creación de esta, es necesario llenar un formulario que cuenta con campos obligatorios, tales como el nombre de la misma, su sigla y su país de origen. Una vez creada la institución, es posible acceder a sección de carga masiva de académicos, donde es necesario incluir los links donde se listan los académicos que serán incluidos a dicha entidad institución, tal como se muestra en la Figura 6.



The screenshot shows a web interface titled "Páginas web". It contains a table with four rows. Each row has a URL in the first column and an action button in the second column. The first three rows have red "Eliminar" buttons, and the fourth row has a blue "Guardar" button. Below the table is a blue link labeled "Agregar nuevo".

URL	Acción
https://www.dcc.uchile.cl/nosotros/academicos/ajc	Eliminar
https://www.dcc.uchile.cl/nosotros/academicos/ajp	Eliminar
https://www.dcc.uchile.cl/nosotros/academicos/pex	Eliminar
https://ejemplo.com/personas	Guardar

[Agregar nuevo](#)

Figura 6. Sección para agregar links externos con información de académicos (por ejemplo: <https://www.dcc.uchile.cl/nosotros/academicos/ajc>).

Cabe señalar que el sistema no ejecuta ningún código javascript que puedan tener estas páginas externas. Por lo que solo se considera el primer contenido HTML que estas páginas envían en cada petición.

Paso 2. Limpieza y agrupación de datos

Una vez registradas estas páginas externas, es posible iniciar el proceso de limpieza y agrupación de la información. El proceso de limpieza consiste en la extracción de todo contenido que exista entre las etiquetas `<body>`, la eliminación de todo comentario `<!-- -->` y el borrado de emojis o símbolos especiales que no pertenezcan a ningún alfabeto.

Para el proceso de agrupación, se transforma el resultado del proceso de limpieza en un JSON, el cual permite el recorrido de las etiquetas como si de un árbol se tratase. El objetivo de este proceso es agrupar el contenido de las etiquetas que puedan contener texto simple, tales como `<p>`, `<div>`, etc., con otras etiquetas que cuenten con los mismos ancestros. Se puede apreciar una representación visual en la Figura 7, donde Texto1 y Texto3 forman parte de un mismo grupo.

1	<code><body></code>
2	<code><div></code>
3	<code><div></code>
4	<code>Texto1</code>
5	<code><p>Texto2</p></code>
6	<code></div></code>
7	<code><div></code>
8	<code>Texto3</code>
9	<code><p>Texto4</p></code>
10	<code></div></code>
11	<code></div></code>
13	<code></body></code>

Figura 7. Representación visual de una agrupación.

Paso 3. Filtro y aceptación de grupos

Una vez obtenido estos grupos de textos, se inicia el proceso de filtrado y aceptación. Estos grupos comúnmente corresponden a un listado de correos electrónicos, números telefónicos y oficinas, donde los miembros del grupo cuentan con caracteres especiales como `@` o números. Los grupos que no son un listado de nombres de personas son eliminados.

Para el proceso de aceptación se busca a cada uno de los miembros de un grupo en la API de búsqueda de DBLP¹⁴. Allí se guardan todos los resultados de la búsqueda en la tabla *InvestigadorCandidato* de la base de datos. Si al menos un miembro del grupo cuenta con algún resultado de búsqueda, todo el grupo es aceptado y considerado como una potencial lista de académicos. En caso de ser un grupo de más de 10 miembros, se tiene como requisito que al menos dos miembros cuenten con resultados.

Paso 4. Selección de académicos por coautor

Todo candidato a académico que tenga un solo resultado de la búsqueda en el API de DBLP es considerado como un académico “aceptado”. Para cada uno de estos académicos se listan los IDs de sus coautores en una lista comunitaria. Esto se hace con el objetivo de desambiguar los candidatos que tengan más de un resultado de la búsqueda en la API de DBLP, ya que, si alguno de sus resultados pertenece a esta lista, dicho resultado es considerado como el correcto, y considera al académico como “aceptado”, repitiendo el proceso.

Paso 5. Selección manual de académicos

Una vez terminado el proceso anterior, los académicos candidatos y “aceptados” son entregados al usuario para que realice una selección final de los académicos. En caso de ser necesario, éste puede omitir algún resultado o asignar uno de los múltiples resultados que entrega la API de DBLP, si es que éste no fue identificado por algún coautor (Paso 4). El usuario puede seleccionar académicos que no cuenten con resultados en DLBP, o que éste no sea único resultando en perfiles sin datos. Se puede ver un ejemplo en la Figura 8.

<p>Éric Tanter ✓</p> <p>Selecciona el académico correspondiente.</p> <p>Éric Tanter ▼</p> <p>dblp.org/pid/t/EricTanter</p> <p>Omitir</p>	<p>Matías Toro ✓</p> <p>Selecciona el académico correspondiente.</p> <p>Matías Toro ▼</p> <p>dblp.org/pid/169/6836</p> <p>Omitir</p>
<p>Rodrigo Arenas A. ⚠</p> <p>Selecciona el académico correspondiente.</p> <p>Rodrigo Arenas-Galnares ▼</p> <p>dblp.org/pid/254/4598</p> <p>Omitir</p>	<p>Juan Arriagada ?</p> <p>Selecciona el académico correspondiente.</p> <p>▼</p> <p>Omitir</p>

¹⁴ <https://dblp.org/faq/13501473.html>

Figura 8. Ejemplo de selección manual de académicos.

Paso 6. Extracción de datos en DBLP y Aminer

En este punto, la aplicación conoce los nombres de los académicos a ser guardados, y en algunos de ellos conoce también los IDs de DBLP correspondientes. Para los casos en el que se conoce el IDs de DBLP de un investigador, se busca en DBLP sus perfiles, para luego guardar toda información útil como sus nombres usados en internet (ejemplo, Sergio F. Ochoa), el número de papers publicados, y sus IDs de fuentes externas, como por ejemplo su ID de ORCID.

Una vez extraída esta información, se procede a buscar los perfiles de usuarios en Aminer¹⁵ gracias a sus nombres usados en internet. De Aminer se obtiene información pública, como correos electrónicos, páginas web personales y keywords que caracterizan el trabajo del investigador (esta metadata es utilizada para filtrar académicos). Toda esta información es guardada en la base de datos, específicamente en el conjunto de tablas correspondientes a *Investigador* (Figura 5).

4.2. Descripción de la API

Una de las formas de acceder a los datos del sistema es por medio de una API pública. Esta API permite un acceso programático para cualquier sistema externo que requiera los datos con los que se cuenta. La documentación completa se puede encontrar en una de las vistas del frontend de la aplicación, accesible desde el menú superior. A continuación, se presentan los endpoints más importantes del sistema junto con ejemplos ilustrativos para comprender de mejor manera el funcionamiento.

4.2.1. Base URL

La URL base para acceder a los recursos es: `https://hostname/api/v1/`; donde se espera que cualquier versión futura que genere cambios significativos se reduzca simplemente a cambiar la versión.

4.2.2 Búsqueda básica por nombre

Este endpoint permite a los usuarios anónimos buscar académicos basándose en búsquedas simples por nombre. Estos pueden tener un margen de error, por lo que el resultado a entregar es un listado de posibles candidatos ordenados por el nombre que

¹⁵ <https://www.aminer.org/>

contenga mayor similitud. El algoritmo utilizado para esta búsqueda lleva por nombre Trigram Similarity.

Request: `get /buscar/academico`

Headers:

Content-Type: application/json
Accept: application/json

Query Parameters:

query - String con el nombre a buscar.

Response: (list)

score - (float) Valor que indica que tan cercano es la búsqueda con el nombre.

id - (int) ID de académico en el sistema.

nombre - (string) Nombre de académico registrado.

nombre_externo - (string) Nombre utilizado por el académico en sistemas externos.

unidad - (string) Nombre de la unidad a la que pertenece el académico.

institución - (string) Institución a la que pertenece el académico.

institución_sigla - (string) Sigla de la institución.

Ejemplo de request:

```
1 curl --request GET \  
2     --get "https://hostname/api/v1/buscar/academico?query=nombre" \  
3     --header "Content-Type: application/json" \  
4     --header "Accept: application/json"
```

Figura 9. Ejemplo de request para una búsqueda básica por nombre.

Ejemplo de respuesta (200):

```
1 [  
2   {  
3     "score": 0.5,  
4     "nombre": "Nombre encontrado",  
5     "nombre_externo": "Nombre encontrado externo",  
6     "unidad": "Unidad de la Institución X",  
7     "institución": "Institución X",  
8     "institución_sigla": "X",  
9   },  
10  ...
```

Figura 10. Ejemplo de respuesta para una búsqueda básica por nombre.

4.2.3 Búsqueda avanzada con filtros y keywords

Este endpoint permite a los usuarios anónimos buscar académicos basándose en búsquedas y filtros por nombre. La búsqueda por nombre puede tener un margen de error, por lo que el resultado a entregar es un listado de posibles candidatos, ordenados por el nombre que contenga mayor similitud. Los filtros disponibles son por institución, país, áreas, subáreas y keyword.

Request: `get /buscar/academico/avanzada`

Headers:

Content-Type: application/json

Accept: application/json

Query Parameters:

nombre - String con el nombre a buscar.

institucion - String con el nombre de institución a la que pertenece el académico.

pais - Código ISO 3166-1 del país al que pertenece la institución del académico.

area - Área en la cual el académico desempeña sus investigaciones.

subarea - Subárea en la cual el académico desempeña sus investigaciones.

keyword - Metadata generada a partir de las investigaciones del académico.

Response: (list)

score - (float) Valor que indica que tan cercano es la búsqueda con el nombre.

id - (int) ID de académico en el sistema.

nombre - (string) Nombre de académico registrado.

nombre_externo - (string) Nombre utilizado por el académico en sistemas externos.

unidad - (string) Nombre de la unidad a la que pertenece el académico

institución - (string) Institución a la que pertenece el académico

institución_sigla - (string) Sigla de la institución.

Ejemplo de request:

1	curl --request GET \
2	

```

--get
3 "https://hostname/api/v1/buscar/academico/avanzada?nombre=NombreAcademic
4 o&pais=CL&keyword=Mobile" \
--header "Content-Type: application/json" \
--header "Accept: application/json"

```

Figura 11. Ejemplo de request, para una búsqueda avanzada por nombre y filtro.

Ejemplo de respuesta (200):

```

1 [
2   {
3     "score": 0.5,
4     "nombre": "Nombre encontrado",
5     "nombre_externo": "Nombre encontrado externo",
6     "unidad": "Unidad de la Institución X",
7     "institución": "Institución X",
8     "institución_sigla": "X",
9   },
10  ...
11 ]

```

Figura 12. Ejemplo de respuesta, para una búsqueda avanzada por nombre y filtro.

4.2.4 Obtención de Académico por ID

Este endpoint permite a los usuarios anónimos extraer toda la información que se tiene sobre un académico en específico. Esta petición es realizada por cualquier ID que este académico posea, ya sea el identificador otorgado por el sistema mismo, el identificador de DBLP o el identificador de ORCID.

Request: `get /academico`

Headers:

Content-Type: application/json
Accept: application/json

Query Parameters:

id - Identificador otorgado por el sistema.
dblp - Identificador de DBLP
orcid - Identificador de ORCID.

Response:

id - (int) ID del académico en el sistema.
nombre - (string) Nombre del académico registrado.
nombre_externo - (string) Nombre utilizado por el académico en sistemas externos.
unidad - (string) Nombre de la unidad a la que pertenece el académico
institución - (string) Institución a la que pertenece el académico
institución_sigla - (string) Sigla de la institución.
grado_máximo - (string) Grado máximo obtenido por el académico.
email - (string) Correo electrónico público del académico.
webpage - (string) Página web del académico
dblp_id - (string) Identificador de DBLP.
orcid_id - (string) Identificador de ORCID.
keyword[] - (array[string]) Arreglo con keyword del académico.
áreas[subáreas] - ({string, array[string]}) JSON con áreas y las subáreas del ámbito de trabajo del académico

Ejemplo request:

```
1 curl --request GET \  
2     --get "https://hostname/api/v1/academico/?id=3" \  
3     --header "Content-Type: application/json" \  
4     --header "Accept: application/json"
```

Figura 13. Ejemplo de request, para obtención de académico por ID.

Ejemplo de respuesta (200):

```
1 {  
2   "id": 2,  
3   "nombre": "Nombre encontrado",  
4   "nombre_externo": "Nombre encontrado externo",  
5   "unidad": "Unidad de la Institución X",  
6   "institución": "Institución X",  
7   "institución_sigla": "X",  
8   "grado_máximo": "Doctorado",  
9   "email": "email@dominio.com",  
10  "webpage": "https://webpage.com",  
11  "dblp_id": "12/1245",  
12  "orcid_id": "1245-1245-1234-1234",  
13  "keyword": [  
14    "keyword 1",
```

```

15     "keyword 2",
16     ...
17 ],
18 "áreas": [
19     {
20         nombre: "Área nombre",
21         subáreas: [
22             "subárea 1",
23             "subárea 2",
24             ....
25         ]
26     },
27     ...
28 ],
}

```

Figura 14. Ejemplo de respuesta, para obtención de académico por ID.

4.2.5 Obtención de Institución y Unidades con Académicos

Este endpoint permite a los usuarios anónimos extraer toda la información que se tiene sobre una institución en específico. Esta petición es realizada por el nombre o id de la institución. En la respuesta se incluye un listado de académicos de dicha institución, ordenados por sus respectivas unidades.

Request: `get /institucion`

Headers:

Content-Type: application/json

Accept: application/json

Query Parameters:

id - Identificador otorgado por el sistema a la institución

nombre - Nombre de la institución.

Response:

id - (int) ID de la institución en el sistema.

nombre - (string) Nombre de la institución.

sigla - (string) Sigla de la institución.

webpage - (string) Página web de la institución.

unidades[] - (array[string]) Lista con las unidades del sistema

Ejemplo request:

```
1 curl --request GET \  
2     --get "https://hostname/api/v1/institucion/?id=3" \  
3     --header "Content-Type: application/json" \  
4     --header "Accept: application/json"
```

Figura 15. Ejemplo de request, para obtención de institución por ID.

Ejemplo de respuesta (200):

```
1 {  
2   "id": 3,  
3   "nombre": "Nombre encontrado",  
4   "sigla": "SIGLA",  
5   "webpage": "https://webpage.com",  
6   "unidades": [  
7     "Unidad 1",  
8     ...  
9   ],  
10 }
```

Figura 16. Ejemplo de respuesta, para obtención de institución por ID.

4.2.5 Obtención de Áreas de Trabajo

Este endpoint permite a los usuarios anónimos obtener un listado de todas las áreas de trabajo presentes en el sistema; por ejemplo: algoritmos, lenguajes, ingeniería de software, etc. Esto es especialmente útil para realizar posteriores búsquedas de académicos filtradas por área.

Request: `get /areas`

Headers:

Content-Type: application/json

Accept: application/json

Response: (list)

id - (int) ID del área en el sistema.

nombre - (string) Nombre del área.

Ejemplo request:

```
1 curl --request GET \  
2     --get "https://hostname/api/v1/areas" \  
3     --header "Content-Type: application/json" \  
4     --header "Accept: application/json"
```

```
2 --get "https://hostname/api/v1/areas \  
3 --header "Content-Type: application/json" \  
4 --header "Accept: application/json"
```

Figura 17. Ejemplo de request, para obtención de áreas.

Ejemplo de respuesta (200):

```
1 [  
2   {  
3     "id": 3,  
4     "nombre": "Cybersecurity",  
5   },  
6   ...  
7 ]
```

Figura 18. Ejemplo de respuesta, para obtención de áreas.

4.2.6 Obtención de Subáreas

Este endpoint permite a los usuarios anónimos obtener un listado de las subáreas de un área de trabajo en específico presente en el sistema.

Request: `get /subareas`

Headers:

Content-Type: application/json
Accept: application/json

Query Parameters:

id - ID de la área en el sistema.
área - Nombre del área.

Response: (list)

nombre - (string) Nombre de subárea.

Ejemplo request:

```
1 curl --request GET \  
2 --get "https://hostname/api/v1/subareas?área="Cybersecurity" \  
3 --header "Content-Type: application/json" \  
4 --header "Accept: application/json"
```

Figura 19. Ejemplo de request, para obtención de subáreas.

Ejemplo de respuesta (200):

1	[
2	{
3	"nombre": "Security Services",
4	},
5	...
6]

Figura 20. Ejemplo de respuesta, para obtención de áreas.

4.3. Interfaces de Usuario

El sistema cuenta con cinco interfaces de usuario principales. Éstas permiten acceder a la información de una forma orgánica, en la que cualquier usuario sin importar su experiencia con la computadora pueda encontrar la información que desee. Las vistas son responsivas y capaces de ser utilizadas por medio de smartphones. Todas las interfaces cuentan con un menú de navegación en la barra superior que permite acceder de forma directa a las distintas vistas (Figura 21).



Figura 21. Menú de navegación situado en la parte superior.

4.3.1 Página de inicio

La Figura 22 muestra la página principal (o de inicio) del sistema. Ésta corresponde a la primera vista que un usuario encuentra al entrar en la aplicación. La página presenta una barra de búsqueda que permite encontrar al académico deseado. Este buscador también cuenta con menús desplegables para filtrar la información de búsqueda, permitiendo incluso obtener un listado de académicos pertenecientes a cierta subárea de interés. Una vez realizada la búsqueda, el usuario es redirigido a la interfaz de búsqueda conteniendo los resultados.



Figura 22. Vista de la página principal.

4.3.2 Servicio de búsqueda

La Figura 23 muestra la interfaz de búsqueda que, al igual que la vista de página principal, permite realizar búsquedas por medio de una barra con filtros. El objetivo de esta vista es mostrar los resultados de las búsquedas. Ésta cuenta con dos pestañas que permiten ver los resultados de académicos o de instituciones; estas pestañas se deshabilitan en caso de que la búsqueda arroje 0 resultados (o sea, que no haya académicos que coincidan con el patrón de búsqueda).

En el caso que la búsqueda de académicos sea exitosa, la interfaz muestra un listado con cuadros resumen de la información más importante de cada académico. Este listado se muestra ordenado según el siguiente orden de prioridad:

1. Qué tan parecido es el string de búsqueda con el nombre del académico.
2. Si el string es parecido a algún keyword, se ordena por el peso de la relación keyword-académico.

Como ejemplo se puede ver en la Figura 23, que el texto buscado es 'user experience', donde es reconocido 'User Experience' como keyword, y el primer resultado es aquel que posee un mayor peso con este keyword. Para recalcar el keyword reconocido, éste se resalta en rojo en los cuadros de resumen.

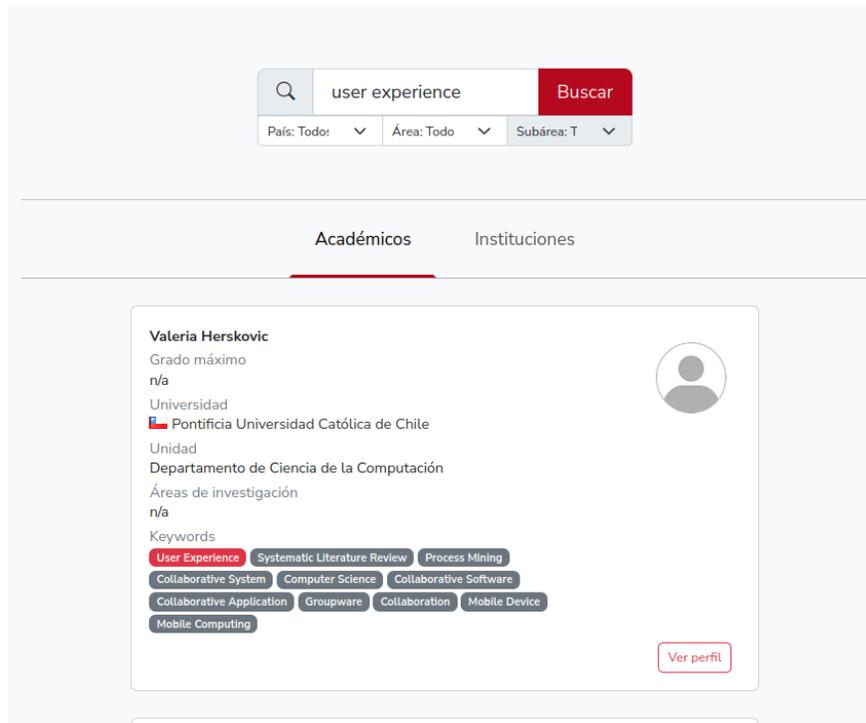


Figura 23. Interfaz de búsqueda.

4.3.3 Instituciones

En la Figura 24 se muestra la interfaz de las instituciones. Ésta corresponde a un listado paginado, con cuadros resumen de las instituciones que están ingresadas en el sistema. Estos cuadros muestran el nombre de la institución, su sigla, su página web y el país de origen. Este listado puede ser filtrado según su país de procedencia, gracias a un listado desplegable en la esquina superior derecha. Aquellas instituciones que posean un administrador de instituciones son marcadas con la palabra “Verificado”, dado que se asume que este administrador certifica y corrige la información registrada por los ETLs.

Desde esta interfaz el administrador del sistema puede crear nuevas instituciones, pulsando el botón ‘Nueva’ únicamente visible para administradores. Este botón abre un modal que contiene un formulario con los campos Nombre, Sigla, Página Web y País, el cual es verificado tanto por el frontend como el backend a la hora enviar la petición de creación de la institución.

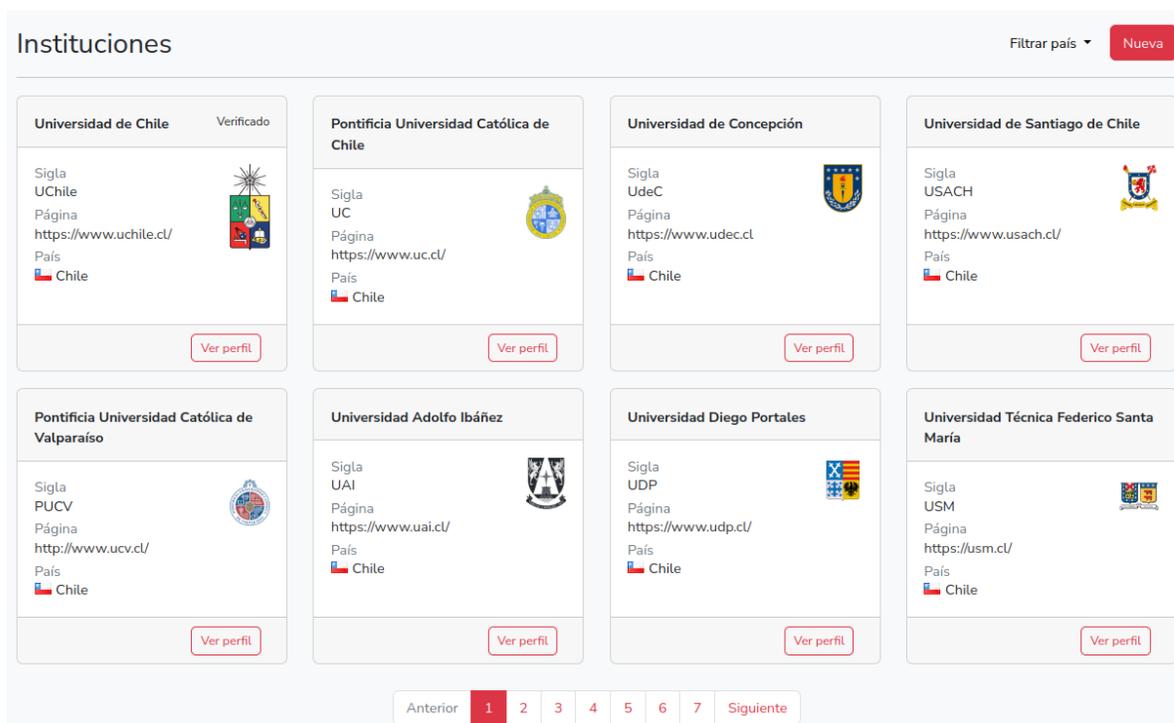


Figura 24. Interfaz con el listado de instituciones.

4.3.4 Perfil de una institución

En la Figura 25 se muestra la interfaz del perfil de una institución. En la parte superior se muestra el nombre de la institución, su sigla, un botón que abre en una nueva pestaña la página web de la institución, y otro botón que abre un menú desplegable con los identificadores de la institución en otros sistemas como wikidata¹⁶.

En la parte inferior se muestran en forma de pestañas las diferentes unidades que son parte de la institución. Estas pestañas muestran en una tabla los diferentes académicos que pertenecen a la unidad seleccionada. Esta tabla tiene como campo el nombre del académico, su identificador de DBLP, su identificador de ORCID, y la cantidad de artículos publicados en journals y en conferencias internacionales. Al hacer clic en uno de los nombres de los académicos, el sistema redirecciona al usuario al perfil del académico.

En esta vista el usuario administrador de institución puede editar la información de la institución, crear unidades, editar unidades y eliminar unidades. También puede crear nuevos académicos, cambiarlos de unidad y eliminarlos. Finalmente, al administrador se le permite cargar académicos por medio de páginas web externas, tal como fue indicado en la sección 4.1: Carga de datos.

¹⁶ <https://www.wikidata.org>

Universidad de Chile (UChile) ID (Verificado) [Ver como editor](#)

DCC

Departamento de Ciencias de la Computación

Mostrar registros por página Buscar:

#	Nombre	DBLP	ORCID	N° Journals	N° Conferencias
1	Andrés Abeliuk	118/6644		8	20
2	Nelson Baloian	b/NelsonBaloian	0000-0003-1608-6454	42	125
3	Jérémy Barbay	56/5867	0000-0002-3392-8353	17	34
4	María Cecilia Bastarrica	b/MCeciliaBastarrica	0000-0002-8616-2144	18	55
5	Felipe Bravo Márquez	04/8612	0000-0002-2153-4306	15	31
6	Benjamín Bustos	15/5056	0000-0002-3955-361X	40	70
7	Eduardo Graells-Garrido	132/8956	0000-0003-0722-5969	7	30
8	Claudio Gutiérrez	g/ClaudioGutierrez	0000-0002-4559-6544	27	74
9	Francisco J. Gutiérrez	161/3505	0000-0002-5285-8901	14	38
10	Alejandro Hevia	h/AlejandroHevia	0000-0002-4774-0693	7	22

Mostrando página 1 de 7 Anterior **1** 2 3 4 5 6 7 Siguiente

Figura 25. Vista del perfil de una institución.

4.3.5 Perfil de un académico

En la Figura 26 se muestra el perfil de un académico. En esta se vista se puede observar toda la información que se tiene sobre un académico. En caso de tener asignado un identificador de DBLP, el nombre asignado por este servicio se muestra entre paréntesis. Junto al nombre se puede apreciar un botón que abre una lista desplegable, la cual contiene todos los identificadores externos que el académico posee.

El perfil divide la información a mostrar en secciones según la naturaleza de los datos. En la primera sección se muestran los datos personales del académico, éste incluye su página web personal y su correo electrónico público. En la siguiente sección se muestra la institución y la unidad a la que pertenece el académico, y su país al que pertenece la institución.

En la tercera, cuarta y quinta sección se muestra la actividad científica que esta persona tiene en el ámbito académico; donde se indica la cantidad de artículos en journals, en conferencias internacionales, sus áreas de investigación y las coautorías asociadas a él/ella. Adicionalmente se muestran los coautores a distancia dos, esto quiere decir los coautores de los coautores. Esto permite mostrar más académicos que están probablemente involucrados en las mismas áreas de investigación. Finalmente, se muestran las keywords (palabras claves) obtenidas por Aminer. Estas keywords caracterizan las temáticas sobre las que investiga el académico.

Un usuario académico puede modificar de forma manual los datos propios; incluyendo su nombre, su página web, su correo electrónico público y el identificador DBLP asociado. Este último permite actualizar la información de publicaciones del académico. Un usuario administrador de institución puede hacer lo mismo sobre los investigadores que administra; o sea, los que pertenecen a su institución y unidad.

Sergio Ochoa (Sergio F. Ochoa) Doctorado ID

Datos
Página web: <http://users.dcc.uchile.cl/~sochoa/>
Email: sochoa@dcc.uchile.cl

Institución
 Universidad de Chile
Unidad: Departamento de Ciencias de la Computación
País:  Chile

73 Journals 151 Conferencias

Áreas de investigación
Computer Software
Software Engineering Software Development Environment And Technology

Coautores

[81] José A. Pino	
[46] Roc Messeguer	
[29] Rodrigo M. Santos	
[28] Francisco J. Gutiérrez  Universidad de Chile	 
[26] Luis A. Guerrero	
[23] Valeria Herskovic	 

Distancia 2

[18] Carolina Fuentes	
[16] Iyubanit Rodriguez	
[25] Nancy Hitschfeld  Universidad de Chile	 
[22] Wolfram Luther	
[12] José I. Benedetto	
[17] Felipe Baytelman	

Keywords
Mobile Collaboration Mobile Device Mobile Collaborative Application Mobile Computing Software Engineering Mobile Shared Workspaces Collaborative Learning Groupware Groupware System Collaborative Work Older Adults
Mobile Collaborative Work Collaborative System Communication Support Collaborative Application Social Interaction Latin America Mobile Worker Decision Support System Software Development

Figura 26. Perfil de un académico.

5. Evaluación del Sistema

En este capítulo se indica cómo se evaluó el sistema, los resultados obtenidos y qué es lo que se puede concluir acerca de la usabilidad y utilidad de esta plataforma. Para determinar la usabilidad se utilizó el System Usability Scale (SUS), mientras que para evaluar la utilidad percibida se utilizó una parte de la encuesta del Technology Adoption Model (TAM).

Para llevar a cabo la evaluación, el sistema fue hospedado en la plataforma DigitalOcean¹⁷, el cual es un proveedor estadounidense de servidores virtuales privados. Para esta versión de testing, se cargaron datos reales de académicos que fueron recolectados desde internet. Dicha versión está disponible en: <http://143.110.227.147/>. A continuación se describe a los participantes, las pruebas realizadas y los resultados obtenidos.

5.1. Descripción de los Evaluadores

Los evaluadores fueron seis académicos de cuatro universidades distintas, todos ellos tenían la necesidad de identificar académicos que pudieran revisar informes o propuestas de estudiantes, así como también que pudieran formar parte de comisiones de defensa de memorias y tesis. A continuación, se listan estos usuarios con su respectiva universidad y rol dentro de la misma.

- U1: Universidad de Chile - Jefe del área de desarrollo de sistemas.
- U2: Universidad de Chile - Ex-coordinador del MTI.
- U3: Pontificia Universidad Católica de Chile - Jefe de carrera.
- U4: Universidad de Talca - Jefe de carrera.
- U5: Universidad de Talca - Académico.
- U6: Universidad de Santiago de Chile - Académico.

5.2. Descripción de la evaluación basada en System Usability Scale

El System Usability Scale (SUS) es una herramienta de medición que permite evaluar la percepción de los usuarios sobre el sistema con respecto a la usabilidad. Para ello se cuantifica la eficacia, eficiencia y satisfacción que perciben los usuarios luego de alcanzar objetivos al utilizar el sistema.

Para la evaluación, los usuarios calificaron 10 ítems en una escala Likert; es decir, deben asignar puntajes que van del 1 al 5, basados en el nivel de acuerdo con la afirmación que

¹⁷ <https://www.digitalocean.com/>

se les plantea. Las respuestas con puntaje 1 son equivalentes a estar “muy en desacuerdo” con las afirmaciones, mientras que las de puntaje 5 son equivalentes a estar “muy de acuerdo”. Las afirmaciones están intercaladas de manera negativa y positiva para evitar sesgos a la hora de responder.

Para determinar el puntaje final de los evaluadores, es necesario transformar las puntuaciones asignadas para cada afirmación a un nuevo número (dependiendo si es positiva o no), sumarlos y multiplicarlos por 2,5 para convertir las puntuaciones originales de 0 al 40 a una escala de 0 al 100. La transformación para las afirmaciones positivas se puede encontrar en la ecuación 1, donde simplemente se les resta el punto base, mientras que la transformación de las afirmaciones negativas se encuentra en la ecuación 2, donde se invierten las puntuaciones. Finalmente se utiliza la ecuación 3, para transformar las escalas y obtener la puntuación final. Para interpretar este puntaje final, la encuesta SUS establece que si el puntaje es superior al 68 el resultado es considerado como positivo, y por lo tanto, el software es usable.

$I_i :=$ Puntaje ítem i . $IP :=$ Ítems Positivos. $IN :=$ Ítems Negativos

$$IP = (I1 - 1) + (I3 - 1) + (I5 - 1) + (I7 - 1) + (I9 - 1) \quad (1)$$

$$IN = (5 - I2) + (5 - I4) + (5 - I6) + (5 - I8) + (5 - I10) \quad (2)$$

$$R = 2,5 * (IP + IN) \quad (3)$$

Las afirmaciones o ítems pertenecientes al cuestionario son las siguientes:

1. Creo que me gustaría usar la plataforma frecuentemente.
2. Pienso que el sistema es innecesariamente complejo.
3. El sistema es fácil de usar.
4. Creo que necesitaré asistencia de un técnico para usar el sistema.
5. Pienso que las funcionalidades del software están bien integradas.
6. La plataforma tiene demasiadas inconsistencias.
7. Creo que la mayoría de las personas aprenderán a usar el sistema rápidamente.
8. El sistema no es agradable para su uso.
9. Me siento seguro al usar el sistema.
10. Necesito aprender muchísimas cosas antes de lograr avanzar en el uso del software.

5.3. Descripción de la evaluación basada en Technology Adoption Model

El modelo de adopción tecnológica (TAM) busca explicar los factores que llevan a un individuo a adoptar o no un determinado sistema de información. Este modelo explica el proceso de adopción a partir de dos factores: *utilidad percibida* y *facilidad de uso percibida*. La utilidad percibida es el grado en el que una persona cree que el uso del sistema mejoraría su desempeño laboral, y la facilidad de uso percibida es el grado en que una persona cree que el uso del sistema será libre de esfuerzo.

Dado que la facilidad de uso percibida ya se está midiendo con el cuestionario SUS, sólo se evaluó mediante este modelo la utilidad percibida. Se evaluaron 3 aseveraciones con el mismo sistema de puntuación que en el ítem anterior, asignando puntos del 1 al 5. El puntaje final corresponde al promedio ponderado de las aseveraciones (se invierte el puntaje de la aseveración negativa), como se muestra en la ecuación 4, donde un 3,5 corresponde a una utilidad positiva.

$$A_i := \text{Puntaje aseveración } i.$$

$$R = (A1 + ((6 - A2) + A3)/3 \tag{4}$$

A continuación, se anuncian las aseveraciones pertenecientes al cuestionario TAM:

1. El uso de la plataforma puede mejorar el desempeño de mi labor.
2. El uso de la plataforma no facilita la realización de mi labor.
3. En general considero que la plataforma puede ser útil en mi labor.

5.4. Resultados Obtenidos

En esta sección se muestran los resultados de usabilidad y utilidad percibida.

5.4.1 Resultados de Usabilidad

A continuación, en la tabla 2, se presentan los resultados del cuestionario SUS. En cada fila se muestran los puntajes asignados por cada usuario.

Tabla 2. Resultados de aplicar la System Usability Scale.

Ítem	I1	I2	I3	I4	I5	I6	I7	I8	I9	I10	Resultado [0, 100]
Usuario 1	5	1	5	1	5	1	4	1	5	1	97,5

Usuario 2	5	1	4	1	5	1	4	1	5	1	95
Usuario 3	4	2	4	3	4	2	3	2	4	2	70
Usuario 4	4	1	5	1	5	1	4	2	5	1	92,5
Usuario 5	4	2	4	2	4	1	4	1	4	1	82,5
Usuario 6	5	1	5	1	5	1	4	1	4	1	95
Promedio	4,5	1,3	4,5	1,5	4,7	1,2	3,8	1,3	4,5	1,2	88,75

5.4.2 Resultados de Utilidad Percibida

A continuación, en la tabla 3, se presentan los resultados del cuestionario TAM. En cada fila se muestran los puntajes asignados por cada usuario.

Tabla 3. Resultados de aplicar la encuesta del Technology Adoption Model.

Ítem	I1	I2	I3	Resultado [1, 5]
Usuario				
Usuario 1	5	1	5	5
Usuario 2	5	1	5	5
Usuario 3	5	1	5	5
Usuario 4	5	2	4	4,3
Usuario 5	4	2	4	4
Usuario 6	4	1	4	4,3
Promedio	4,7	1,3	3,8	4,4

5.5. Discusión de los Resultados

En esta sección se discuten los resultados de ambas encuestas, tanto de manera global, como en particular para ciertos indicadores que presenten un mayor interés.

Adicionalmente, se incluye un resumen de los comentarios extras proporcionados por los evaluadores.

5.5.1. Análisis de la Usabilidad

Los resultados de la encuesta SUS presentan un buen promedio general, ya que superan el umbral preestablecido de 68 puntos para ser considerado usable, alcanzando un puntaje de 88,75 sobre 100 puntos. Dentro los ítems que destacan las evaluaciones, se encuentra el punto 7 que menciona “Creo que la mayoría de las personas aprenderán a usar el sistema rápidamente”. Este punto cuenta con el menor promedio individual, pero no llegando a tener un valor negativo. Una de las soluciones para agilizar el aprendizaje de uso de la herramienta, puede ser la exposición de ejemplos y el uso de tooltips en palabras/acciones que sean difíciles de comprender.

Los ítems mejor evaluados son los ítems 6 y 10, que reflejan un sistema consistente, el cual no requiere de conocimientos previos para su uso. Esto permite abrir las puertas a todo tipo de usuario que requiera el uso de esta plataforma.

5.5.2. Análisis de la Utilidad

Los resultados de la encuesta TAM también presentan un buen promedio general, superando el umbral mínimo de 3,5 puntos para ser considerado útil, obteniendo un promedio final 4,4 sobre 5 puntos. Con los resultados obtenidos se puede asegurar que la plataforma permite mejorar el desempeño de las actividades o procesos mencionados en el marco teórico, haciendo que estas sean más fáciles de realizar.

5.5.3 Comentarios extras de evaluadores

Junto a los cuestionarios, a los evaluadores se les dio un espacio para dejar comentarios abiertos. Allí se obtuvieron respuestas positivas, mencionando que el sistema además de ser usable y útil, es de buena calidad e interesante. Además, se obtuvo un comentario mencionando que la API está bien estructurada y documentada.

En cuanto a las respuestas negativas, se mencionó en varias ocasiones la necesidad de poblar la base de datos con más datos, y ajustar los resultados de búsqueda por proximidad. Esto abre una oportunidad para realizar diversos trabajos a futuro.

6. Conclusiones y Trabajo a Futuro

Varias actividades o procesos propios de las instituciones académicas tienen la necesidad de buscar académicos; por ejemplo, para conformar un comité examinador de una memoria o tesis, para evaluar informes académicos, o para dictar cursos o seminarios, entre otros. Debido a la falta de un repositorio abierto, que permita realizar búsquedas filtradas de académicos e investigadores, este trabajo de memoria buscó apoyar y facilitar dichas actividades. Es por esto que el objetivo de esta memoria consistió en la implementación de un repositorio de académicos, el cual pueda ser consultado por el público en general a través de dos mecanismos: un portal web y una API. Ambos canales de consulta permiten ingresar diferentes filtros, que facilitan la obtención de la información requerida. Además, los servicios del repositorio permiten actualizar la información de estas personas, a través de mecanismos manuales y automáticos.

Por otra parte, se definieron perfiles de académicos y de investigadores, y también las áreas de expertise de estos, homogeneizando la información de esas personas. Esta información es obtenida mediante mecanismos automatizados y manuales, los cuales son especialmente útiles para corregir o actualizar información. El frontend de la aplicación permite a los usuarios anónimos realizar búsquedas y navegar por el sistema, democratizando este espacio de información.

Para la evaluación de la aplicación, se realizaron dos encuestas a seis académicos para medir la usabilidad y utilidad de la misma. Estos académicos, que contaban con la necesidad de usar una herramienta como ésta, asignaron un puntaje promedio que superó el umbral esperado para considerar el sistema como usable y útil. Adicional a lo anterior, los académicos comentaron que el sistema es de buena calidad e interesante, y que cuenta con una API bien estructurada y documentada.

Con respecto a los objetivos propuestos, se puede concluir que se cumplieron en su totalidad, puesto que se implementó el sistema propuesto con todos sus requisitos funcionales mínimos y necesarios para que éste pueda ser utilizado.

En lo que respecta al trabajo futuro, se encuentra el refinamiento del sistema de búsqueda, dado que el algoritmo por proximidad muestra algunos resultados erróneos. Por lo tanto, éste podría ser mejorado, dando la posibilidad de deshabilitar los keyword que no correspondan con lo buscado, o dar sugerencias al momento de estar escribiendo en la barra de búsqueda.

Dentro de los comentarios de los evaluadores, se encuentra la escasez de los datos, por lo que como trabajo futuro se pueden implementar nuevos mecanismos automatizados que permitan ampliar más la base de datos actual. Junto a esto, realizar una búsqueda exhaustiva de las instituciones de toda Latinoamérica para obtener un resultado más completo.

En relación con la experiencia de usuario, es necesario utilizar algún mecanismo que enseñe a los usuarios nuevos a adaptarse al sistema. Esto puede ser en forma de ejemplos en la página principal, además de incluir textos explicativos en los casos que se encuentren términos que resulten poco familiares o confusos para un usuario común.

Un servicio interesante para esta plataforma es un mecanismo de notificaciones, en el que un usuario pueda suscribirse como seguidor de un cierto académico o de un grupo de académicos. El suscriptor debería recibir notificaciones, por ejemplo, por medio de correos electrónicos, en caso de que estos académicos tengan algún cambio en su información.

Dado que permitir la edición de los datos a los mismos académicos puede ser utilizada con intenciones deshonestas, se puede implementar un mecanismo de petición de cambios al editar perfiles. Estas peticiones podrían ser aceptadas o rechazadas por un usuario moderador de la aplicación.

7. Bibliografía

- [1] DBLP Computer Science Bibliography. Schloss Dagstuhl - Leibniz Center for Informatics (German: Schloss Dagstuhl - Leibniz-Zentrum für Informatik GmbH). Disponible en: <https://dblp.org>. Accedido el 17 de julio de 2023.
- [2] ORCID. Disponible en: <https://orcid.org/>. Accedido el 17 de julio de 2023.
- [3] ArnetMiner. Disponible en: <https://www.aminer.org/>. Accedido el 17 de julio de 2023.
- [4] BibKG. Disponible en: <https://bibkg.imfd.cl/>. Accedido el 17 de julio de 2023.
- [5] Vincula. Disponible en: <https://vincula.cl/>. Accedido el 17 de julio de 2023.
- [6] ResearchGate. Disponible en: <https://www.researchgate.net/>. Accedido el 17 de julio de 2023.
- [7] Mendeley. Disponible en: <https://www.mendeley.com/>. Accedido el 17 de julio de 2023.
- [8] Google Scholar. Disponible en: <https://scholar.google.com/>. Accedido el 17 de julio de 2023.
- [9] LinkedIn. Disponible en: <https://www.linkedin.com/>. Accedido el 17 de julio de 2023.
- [10] López, Néstor. Herramienta de apoyo a revisiones sistemáticas de la literatura en el área de la computación. Universidad de Chile. Disponible en: <https://repositorio.uchile.cl/bitstream/handle/2250/170457/Herramienta-de-apoyo-a-revisiones-sistematicas-de-la.pdf?sequence=1>. Accedido el 17 de julio de 2023.
- [11] Ley, Michael. DBLP — Some Lessons Learned. Proceedings of the VLDB '09, Lyon, France. August 24-28, 2009. Disponible en: <https://dblp.org/xml/docu/dblpxml.pdf>
- [12] Brooke, J., 1996. — SUS-A quick and dirty usability scale. Usability evaluation in industry, 189 (196), pp.4-7.
- [13] Davis, Fred D., 1989. — Perceived usefulness, perceived ease of use, and user acceptance of information technology. MIS Quarterly 13(3): 319-340.

Anexo: Académicos de Ciencias de la Computación en Chile

A continuación, se indican las páginas Web de los Departamentos de Computación o Informática de distintas Universidades de Chile. En éstas se listan los académicos que trabajan en dichas instituciones.

1.- Universidad de Chile

LandingPage: <https://www.dcc.uchile.cl/>

Académicos Jornada Completa: <https://www.dcc.uchile.cl/nosotros/academicos/ajc>

Académicos Jornada Parcial: <https://www.dcc.uchile.cl/nosotros/academicos/ajp>

2.- Pontificia Universidad Católica de Chile

LandingPage: <https://dcc.uc.cl/>

Académicos: <https://dcc.uc.cl/people>

3.- Universidad de Concepción

LandingPage: <https://fi.udec.cl/pregrado/ingenieria-civil-informatica/>

Académicos: <https://fi.udec.cl/pregrado/ingenieria-civil-informatica/academicos/>

4.- Universidad de Santiago de Chile

LandingPage: <https://www.dmcc.usach.cl>

Académicos: https://www.dmcc.usach.cl/rol_personas/academicos/

5.- Pontificia Universidad Católica de Valparaíso

LandingPage: <https://pucv.cl/>

Académicos: <https://pucv.cl/pucv/pregrado/ingenieria-en-informatica>

6.- Universidad Adolfo Ibáñez

LandingPage: <https://ingenieria.uai.cl/>

Académicos: <https://ingenieria.uai.cl/academicos/>

7.- Universidad Diego Portales

LandingPage: <https://eit.udp.cl/>

Académicos: <https://eit.udp.cl/nuestra-escuela/academicos/>

8.- Universidad Técnica Federico Santa María

LandingPage: <https://www.inf.utfsm.cl/>

Académicos: <https://www.inf.utfsm.cl/investigacion/personas>

9.- Universidad Austral de Chile

LandingPage: <http://informatica.uach.cl/>

Académicos: <http://informatica.uach.cl/profesores/>

10.- Universidad de Talca

LandingPage: <https://ingenieria.otalca.cl/Departamento/Detalles/1>

Académicos: <http://ingenieria.otalca.cl/Facultad/Academicos>

11.- Universidad de la Frontera

LandingPage: <https://icinformatica.ufro.cl/>

Académicos: <http://fica.ufro.cl/wp-content/uploads/2018/03/Depto.-de-Ciencias-de-la-Computaci%C3%B3n-e-Infom%C3%A1tica.pdf>

12.- Universidad de los Andes

LandingPage: <https://www.uandes.cl/carrera/ingenieria-civil-en-ciencias-de-la-computacion/>

Académicos: <https://www.uandes.cl/carrera/ingenieria-civil-en-ciencias-de-la-computacion/academicos/>

13.- Universidad de Valparaíso

LandingPage: <https://informatica.uv.cl/>

Académicos: <https://informatica.uv.cl/escuela/academicos-jornada-completa>

14.- Universidad del Desarrollo

LandingPage: <https://ingenieria.udd.cl/carrera/ingenieria-civil-en-informatica-e-innovacion-tecnologica/home/>

Académicos: <https://ingenieria.udd.cl/carrera/ingenieria-civil-en-informatica-e-innovacion-tecnologica/academicos/>

15.- Universidad del Bío-Bío

LandingPage: <https://www.mcc.ubiobio.cl/web/index.php>

Académicos: <http://www.mcc.ubiobio.cl/academicos.php#aca>

16.- Universidad de Tarapacá

LandingPage: <https://dici.uta.cl/>

Académicos: <https://dici.uta.cl/departamento/academicos/>