



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

INTEGRACIÓN Y EVOLUCIÓN DE SISTEMAS DE INFORMACIÓN DEL DCC

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL EN COMPUTACIÓN

FRANCISCO JAVIER MADRID CABEZAS

PROFESOR GUÍA:  
SERGIO OCHOA DELORENZI

MIEMBROS DE LA COMISIÓN:  
GONZALO NAVARRO BADINO  
NELSON BALOIAN TATARYAN

SANTIAGO DE CHILE  
2017

## Resumen

Desde hace varios años, el Departamento de Ciencias de la Computación (DCC) de la Universidad de Chile ha venido desarrollando sus sistemas de información como si fueran islas. Estos sistemas apoyan principalmente la gestión de actividades académicas, docentes y económico-financieras del DCC. Aunque esta infraestructura de software ha funcionado relativamente bien, la estrategia de desarrollo de sistemas desacoplados limita el crecimiento de dicha infraestructura de cara al futuro. Por esa razón el DCC ha decidido integrar estas islas, tanto a nivel de datos como de servicios, para permitir un desarrollo más armónico y controlado de sus sistemas de información. Se espera que esta integración facilite especialmente el reuso de información, manteniendo una única fuente de verdad para las distintas áreas de negocio (pregrado, postgrado, educación continua, investigación, etc.).

Este trabajo de memoria realizó una prueba de concepto que abordó no sólo la integración de dos sistemas de información del DCC, sino también la reingeniería de uno de ellos y el desarrollo completo del otro. Las aplicaciones en cuestión son el Sistema Administrador de Recursos (SAR) y el Sistema Administrador de Noticias y Eventos (SANE).

Para llevar a cabo la integración de estos sistemas se diseñó e implementó una arquitectura de microservicios que es capaz de contener a las aplicaciones críticas del Departamento. Apoyándose en dicha arquitectura se modificó y extendió el sistema SAR, agregándole por ejemplo funcionalidad para permitir la administración de puestos de trabajos de alumnos de postgrado y profesores visitantes. Por otra parte, el proceso de reservas de recursos (por ejemplo, salas de reuniones) se independizó de las secretarías, permitiendo que académicos y funcionarios puedan hacer reservas según su rol. La nueva versión del sistema SAR está actualmente en producción.

Por otra parte, se diseñó e implementó un nuevo sistema de software, el cual permite mantener y gestionar las noticias y eventos del Departamento (por ejemplo, charlas, defensas de tesis, etc.). Éste permite además alimentar automáticamente otros recursos de entrega de información del Departamento; por ejemplo, su página Web oficial. El sistema SANE también adhiere a la arquitectura de microservicios definida, y permite la interacción con el SAR a través de una API (Application Programming Interface).

El proyecto piloto que buscaba la integración de ambos sistemas a través de una arquitectura de microservicios resultó exitoso, pues no sólo se alcanzaron los objetivos iniciales, sino que además los usuarios finales se mostraron muy satisfechos con las soluciones obtenidas. Este piloto muestra un camino para que en adelante se integren otros sistemas a la nueva infraestructura de software del DCC, independientemente de que estas aplicaciones sean nuevas o legadas. Las nuevas aplicaciones deberán contemplar la intercomunicación con la infraestructura creada, utilizando mecanismos de autenticación adecuados para garantizar la seguridad de sus operaciones.

En resumen, más allá de las ventajas propias de contar con más y mejores servicios de software para apoyar las operaciones del Departamento, este trabajo de memoria buscó determinar la factibilidad y el esfuerzo requerido para integrar los sistemas del DCC a través de una arquitectura de microservicios; y el resultado obtenido fue altamente positivo.

*A mis padres. Sin su esfuerzo y entrega incondicional esto no hubiese sido posible.*

# Agradecimientos

A mi familia por respaldarme durante todo el transcurso de esta aventura, y mi profesor guía Sergio Ochoa por apoyarme en este trabajo de memoria resolviendo las inquietudes que aparecieron en el camino.

A todo el equipo de la asociación sin fines de lucro Epistemonikos, en especial a Daniel Pérez Rada y Gabriel Rada, por darme la oportunidad de desarrollar al máximo mis capacidades, tanto técnicas como interpersonales. También al equipo del Área de Infotecnologías (ADI) y Centro Tecnológico Ucampus por ayudarme a transformar una de mis pequeñas ideas en lo que hoy se conoce como U-Cursos móvil para Android.

A mis amigos del *Culto*, *Bacon* y *Amiguitos* por toda la buena onda y supportear en los malos momentos.

Muchas gracias a Tomás Díaz, Sebastián Ferrada, Valentín Muñoz y Jocelyn Simmonds, por darme una mano en la confección de este documento en L<sup>A</sup>T<sub>E</sub>X, y a Karina Parga por *un arte de documentación*.

# Tabla de contenido

<b>1. Introducción</b>	<b>1</b>
1.1. Justificación del trabajo de memoria . . . . .	1
1.2. Objetivos de la memoria . . . . .	3
1.3. Estructura del documento . . . . .	4
<b>2. Marco teórico</b>	<b>5</b>
2.1. Arquitecturas orientadas a servicios . . . . .	5
2.1.1. Servicios web . . . . .	7
2.1.2. Lenguajes de especificación de interfaces para servicios web . . . . .	7
2.2. Desarrollo basado en microservicios . . . . .	8
2.3. Estrategias de reingeniería de software . . . . .	8
2.4. Herramientas utilizadas . . . . .	9
<b>3. Sistema administrador de recursos</b>	<b>10</b>
3.1. Situación inicial . . . . .	10
3.2. Principales requisitos de la solución . . . . .	11
3.3. Usuarios de la solución . . . . .	12
3.4. Arquitectura de la solución . . . . .	13
3.4.1. Arquitectura física . . . . .	13
3.4.2. Arquitectura lógica . . . . .	14
3.5. Módulos del sistema . . . . .	15
3.6. Modelo de datos . . . . .	17
3.7. Cambios internos de la versión 2.1 . . . . .	18
3.8. Principales vistas del sistema . . . . .	18
3.8.1. Ingreso al sistema . . . . .	18
3.8.2. Menú principal . . . . .	19
3.8.3. Reserva de recursos . . . . .	20
3.8.4. Detalle de una reserva de recurso . . . . .	21
3.8.5. Creación de una reserva personal . . . . .	22
3.8.6. Creación de una reserva para un tercero . . . . .	23
3.8.7. Reserva de puestos de trabajo . . . . .	24
3.8.8. Lista de salas y oficinas . . . . .	25
3.8.9. Panel de administración . . . . .	26
<b>4. Sistema administrador de noticias y eventos</b>	<b>27</b>
4.1. Situación inicial . . . . .	27

4.2.	Metodologías y herramientas de desarrollo empleadas . . . . .	28
4.3.	Arquitectura de la solución . . . . .	30
4.4.	Modelo de datos . . . . .	30
4.5.	Usuarios de la solución . . . . .	32
4.6.	Principales vistas del sistema . . . . .	32
4.6.1.	Home del sitio . . . . .	32
4.6.2.	Lista de eventos . . . . .	33
4.6.3.	Detalles de un evento . . . . .	35
4.6.4.	Lista de noticias . . . . .	36
4.6.5.	Detalles de una noticia . . . . .	37
4.6.6.	Registro de usuario . . . . .	38
4.6.7.	Login de usuario . . . . .	39
4.6.8.	Interfaz de navegación . . . . .	40
4.6.9.	Perfil de usuario . . . . .	41
4.6.10.	Administración de contenidos . . . . .	42
4.6.11.	Creación de eventos y noticias . . . . .	43
4.6.12.	Administración interna . . . . .	44
<b>5.</b>	<b>Evaluación de las soluciones</b>	<b>45</b>
5.1.	Evaluación técnica . . . . .	45
5.2.	Evaluación de usuarios . . . . .	46
<b>6.</b>	<b>Conclusiones y trabajo a futuro</b>	<b>48</b>
	<b>Bibliografía</b>	<b>49</b>
<b>A.</b>	<b>Interfaces</b>	<b>52</b>
A.1.	Interfaces del SANE . . . . .	52
A.2.	Documentación del SANE . . . . .	57
<b>B.</b>	<b>Diagramas</b>	<b>58</b>
B.1.	Diagramas del SAR . . . . .	58
B.2.	Diagramas del SANE . . . . .	59

# Índice de tablas

5.1. Tests coverage del front-end y back-end junto con cantidad de tests del SANE	46
5.2. Complejidad ciclomática promedio e índice de mantenibilidad promedio del SANE . . . . .	46

# Índice de ilustraciones

1.1. Escenario de interacción entre sistemas . . . . .	3
2.1. Estructura básica de una SOA . . . . .	6
2.2. Escenario multi-sistema apoyado por una SOA . . . . .	6
3.1. Interfaz principal del SAR v1.0 . . . . .	11
3.2. Arquitectura física de 3 capas del SAR v2.1 . . . . .	13
3.3. Módulos del SAR v2.1 . . . . .	15
3.4. Diagrama Entidad-Relación simplificado del SAR v2.1 . . . . .	18
3.5. Interfaz de login de usuarios del SAR v2.1 . . . . .	19
3.6. Menú principal del SAR v2.1 . . . . .	19
3.7. Interfaz de usuario para reservas de recursos . . . . .	20
3.8. Detalle de la reserva de un recurso . . . . .	21
3.9. Interfaz de creación de una reserva (personal) de un recurso . . . . .	22
3.10. Interfaz de creación de reserva para un tercero . . . . .	23
3.11. Interfaz de reserva de puestos de trabajo en oficinas . . . . .	24
3.12. Interfaz de lista de salas y oficinas . . . . .	25
3.13. Interfaz de administración de recursos y usuarios . . . . .	26
4.1. Panel de administración de contenidos de la página oficial del DCC . . . . .	28
4.2. Panel de creación de eventos de la página oficial del DCC . . . . .	29
4.3. Diagrama básico de la arquitectura interna del SANE . . . . .	30
4.4. Modelo Entidad-Relación simplificado del SANE . . . . .	31
4.5. Interfaz de usuario mostrada en el <i>home</i> del SANE . . . . .	33
4.6. Lista de eventos registrados . . . . .	34
4.7. Detalles de un evento registrado en el SANE . . . . .	35
4.8. Lista de noticias registradas . . . . .	36
4.9. Detalle de una noticia . . . . .	37
4.10. Interfaz de registro de un usuario . . . . .	38
4.11. Interfaz de login de usuarios . . . . .	39
4.12. Interfaz principal de un usuario logueado . . . . .	40
4.13. Perfil de un usuario del SANE . . . . .	41
4.14. Interfaz de administración del SANE . . . . .	42
4.15. Interfaz de creación de eventos . . . . .	43
4.16. Interfaz de administración interna del SANE . . . . .	44
5.1. Vistas de noticias del SANE renderizadas en un dispositivo Apple iPhone 6 . . . . .	47

A.1. Interfaz de administración de eventos del SANE . . . . .	52
A.2. Interfaz de administración de imágenes del SANE . . . . .	53
A.3. Interfaz de administración de documentos del SANE . . . . .	54
A.4. Interfaz de administración de lugares del SANE . . . . .	54
A.5. Interfaz de administración de organizadores del SANE . . . . .	55
A.6. Interfaz de administración de tags del SANE . . . . .	55
A.7. Interfaz de administración de usuarios del SANE . . . . .	56
A.8. Interfaz de documentación del SANE en HTML . . . . .	57
B.1. Diagrama Entidad-Relación detallado del SAR según el framework Django .	58
B.2. Diagrama Entidad-Relación detallado del SANE según el framework Django	59

# Capítulo 1

## Introducción

En el contexto actual muchas organizaciones requieren planificar y difundir apropiadamente las actividades que llevan a cabo. Esto se hace con el objetivo de llegar a la mayor cantidad de interesados posibles, en el momento apropiado, y ojalá de manera automática (desatendida). El Departamento de Ciencias de la Computación (DCC) de la Facultad de Ciencias Físicas y Matemáticas de la Universidad de Chile no escapa de esta realidad, debido a que en sus dependencias se realizan diversos eventos académicos y de extensión, donde muchos de ellos requieren ser dados a conocer a la comunidad; por ejemplo, el inicio de cursos de diplomados y postítulos, defensas de tesis, charlas de investigación, exámenes de candidatura, etc. Todos estos eventos tienen una población destinataria a la cual llegar con la difusión de manera apropiada, y también tienen recursos (particularmente salas) asociados a ellos. Por lo tanto, cualquier sistema administrador de eventos tendrá que manejar prioridades y disponibilidad de recursos a la hora de planificar y gestionar una actividad de este tipo.

Como parte de este trabajo de memoria se desarrolló una aplicación Web que permite registrar, administrar y dar a conocer, bajo un esquema productor-consumidor, los eventos y noticias del DCC. Como complemento a esto se realizó la reingeniería del actual Sistema de Administración de Recursos (SAR), el cual permite administrar los espacios físicos compartidos (principalmente salas del Departamento) para que estos puedan ser utilizados en diversos eventos. Ambos sistemas deben poder interactuar a nivel de datos y servicios a través de una API (*Application Programming Interface*), con el objetivo de mantener una coordinación entre ellos. Para ello, se definió una arquitectura de microservicios que permitirá a futuro integrar fácilmente otros sistemas del DCC.

### 1.1. Justificación del trabajo de memoria

Este trabajo de memoria abordó varios desafíos, los cuales consistieron en el desarrollo de dos sistemas de información para el Departamento, y su integración bajo una arquitectura de microservicios. Particularmente, el primer desafío fue realizar la reingeniería y puesta en producción de la versión 2.0 del Sistema de Administración de Recursos (SAR). La versión anterior, aunque estaba en producción, presentaba diversas limitaciones como por ejemplo:

- Sistema poco mantenible dado que utilizaba tecnologías obsoletas, con poco o ningún soporte para los navegadores Web de la actualidad.
- Incapacidad para administrar los puestos de trabajo, por ejemplo, los asignados en las oficinas de estudiantes de postgrado y profesores visitantes.
- Acceso limitado, ya que poca gente (principalmente los funcionarios y algunos académicos) tenían acceso para reservar salas.

El segundo desafío consistió en el desarrollo del Sistema Administrador de Noticias y Eventos (SANE) del DCC. Este sistema fue creado desde cero, y permite crear y administrar las novedades que el Departamento quiere dar a conocer a su comunidad. El sitio provee diversos métodos para que nuevas aplicaciones puedan acceder a su información. Debido a que muchos de los eventos del DCC están asociados a un espacio físico (es decir, al lugar donde se va a realizar la actividad), el sistema debía interoperar a nivel de datos con la nueva versión del SAR.

El SANE además administra usuarios y suscripciones, pero la distribución de eventos y noticias está a cargo de otra aplicación, actualmente en desarrollo en el DCC: el Sistema de notificaciones de Noticias y Eventos (SINE). El SANE también representa la fuente de datos que proveerá automáticamente de información a la futura página Web oficial del Departamento.

La Figura 1.1 muestra a grandes rasgos cómo interactúan los sistemas antes descritos y el sitio Web del DCC. Los principales involucrados en el uso de estas aplicaciones son las secretarías, académicos y el personal de Coordinación de Comunicaciones del DCC. Estos últimos además de registrar y administrar las noticias y eventos, los difunden a través de redes sociales; particularmente Facebook y Twitter.

Dado que los sistemas de información de apoyo a las actividades del DCC han sido desarrollados como islas (es decir, sin interacción entre ellas), incluyendo al sistema SAR, el tercer desafío abordado en esta memoria consistió en determinar cómo integrarlos a nivel de datos y servicios sin poner en riesgo su operación. Esta restricción para llevar a cabo la integración surgió debido a que algunos de estos sistemas estaban en producción (particularmente el SAR), y no podían dejar de prestar servicio al Departamento para así poder realizarle los ajustes correspondientes.

Dada esta necesidad de interacción entre el SAR y el SANE, y considerando el hecho de que otros sistemas también necesitaban interactuar entre ellos, se pensó en concebir una solución que pudiera dar cabida a la integración de sistemas a nivel Departamental. Para ello se diseñó una arquitectura de microservicios para integrar no sólo el SAR y el SANE, sino también agregar otros sistemas legados (islas) a este esquema en el corto plazo. Se espera que esta integración permita reutilizar datos, manteniendo una única fuente de verdad según el área de negocio de la que se trate. También se espera que este esquema de integración permita que la infraestructura de software del DCC crezca y evolucione de manera segura y controlada.

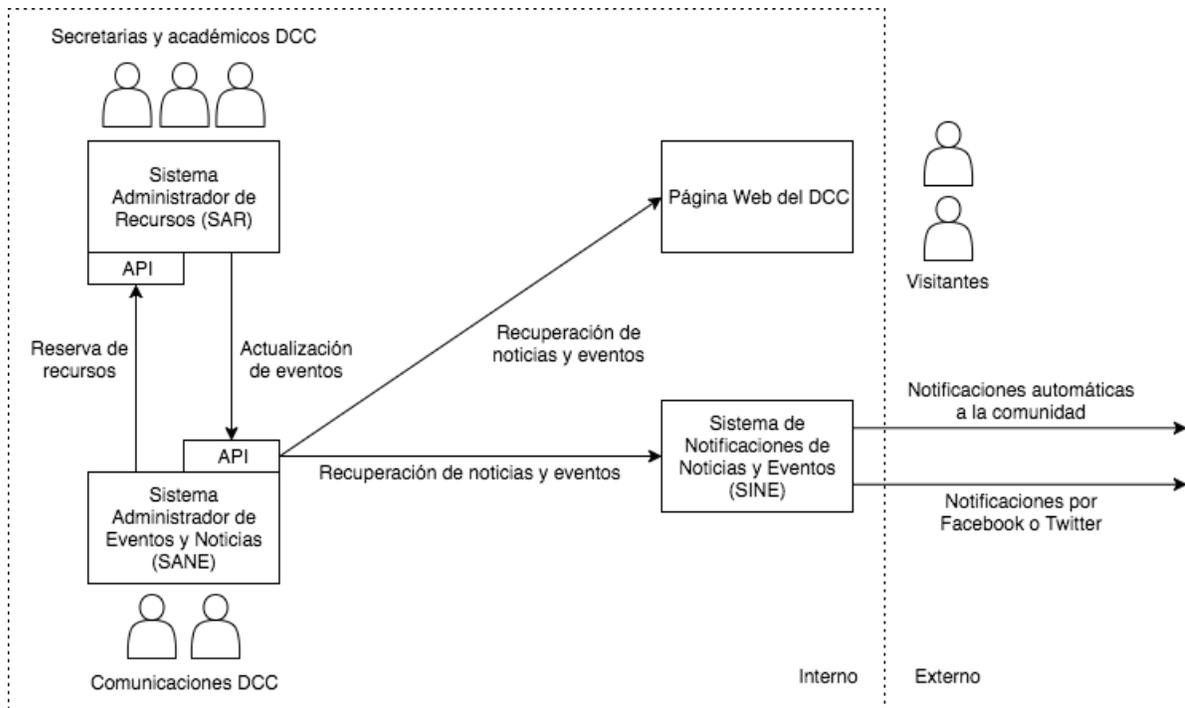


Figura 1.1: Escenario de interacción entre sistemas

## 1.2. Objetivos de la memoria

El objetivo general de este trabajo de memoria fue determinar cómo integrar y evolucionar dos sistemas de información del DCC, con una visión de largo plazo. En primer lugar se realizó la reingeniería de un sistema legado (conocido como SAR) del Departamento, y se puso en producción la nueva versión (SAR 2.0). Cabe destacar que esta versión no sólo extiende la funcionalidad del sistema legado, sino que además permite la integración con otros sistemas.

Por otra parte, también se desarrolló el sistema SANE, complementario al anterior, el cual consiste en una aplicación Web que en conjunto con el SAR permite gestionar (de manera transparente para los usuarios) los eventos y noticias del DCC, junto con los recursos de espacio físico que usualmente se necesita para estos.

Este desarrollo multi-sistema surge a partir de las nuevas necesidades del Departamento, y de las limitaciones del SAR v1.0 indicadas en la justificación de la propuesta. Por lo tanto los objetivos específicos que se desprenden del objetivo general antes enunciado, son los siguientes:

- Realizar la reingeniería necesaria del sistema SAR para permitir su integración con otros sistemas del Departamento (en primera instancia con el SANE).
- Implementar el SANE como una aplicación Web que permita registrar, administrar y compartir por demanda, los eventos y noticias del Departamento. La aplicación debe además permitir la suscripción de usuarios a múltiples canales de notificaciones relacionados a los tipos de eventos.
- Crear una API para cada uno de estos sistemas con el fin de que puedan ser utilizados

como recursos independientes e integrables, tanto a nivel de datos como de servicios por otras aplicaciones. Estas APIs deben permitir a las aplicaciones antes mencionadas, integrarse bajo una arquitectura de microservicios.

### **1.3. Estructura del documento**

El siguiente capítulo describe las arquitecturas de microservicios, el cual es el marco de trabajo sobre el que se sustenta la integración de sistemas propuesta. Posteriormente en el capítulo 3 se muestra el desarrollo de la nueva versión del SAR, y de manera similar en el capítulo 4 se presenta el desarrollo del sistema SANE. En el capítulo 5 se evalúan las aplicaciones a nivel técnico y de usuarios; y se presentan los resultados de dichas evaluaciones. Finalmente el capítulo 6 presenta las conclusiones y el trabajo a futuro.

# Capítulo 2

## Marco teórico

En este capítulo se introducen los conceptos involucrados en el trabajo de memoria realizado. También se describe brevemente la arquitectura, metodologías y herramientas utilizadas en este trabajo de memoria.

### 2.1. Arquitecturas orientadas a servicios

Según la W3C, “las Arquitecturas Orientadas a Servicios consisten en un conjunto de componentes que pueden ser invocadas, y cuyas descripciones de interfaz pueden ser públicas y descubiertas” [11]. Entenderemos por componente a “un objeto de software, que interactúa con otros componentes, encapsulando cierta funcionalidad o conjunto de éstas. Una componente tiene una interfaz claramente definida y su comportamiento es similar al del resto de los componentes dentro de la misma arquitectura” [11]. Típicamente estos componentes son llamados servicios Web (o WS, por sus siglas en inglés).

Considerando la definición anterior podemos decir que una Arquitectura Orientada a Servicios (SOA, por sus siglas en inglés) está conformada por un conjunto de programas, que cumplen un rol predefinido y que interactúan entre ellos a través de interfaces de programación (APIs), para lograr el resultado deseado. Los tipos de componentes que forman parte de una SOA son los siguientes (Figura 2.1): servicios, proveedores de servicios, consumidores de servicios, bus de servicios, el motor de integración y el directorio de servicios. A continuación se explican brevemente cada uno de ellos.

Un servicio Web es un programa de software que encapsula su funcionalidad detrás de una interfaz especificada en WSDL (Web Service Description Language), y que se comunica con otros programas a través de mensajes en XML según lo definido por su interfaz. En una SOA, un WS representa la unidad de trabajo realizada por un proveedor de servicios para lograr los resultados finales deseados por el consumidor de dicho servicio. Los roles de proveedor y de consumidor son realizados por agentes de software en representación de sus propietarios [13].

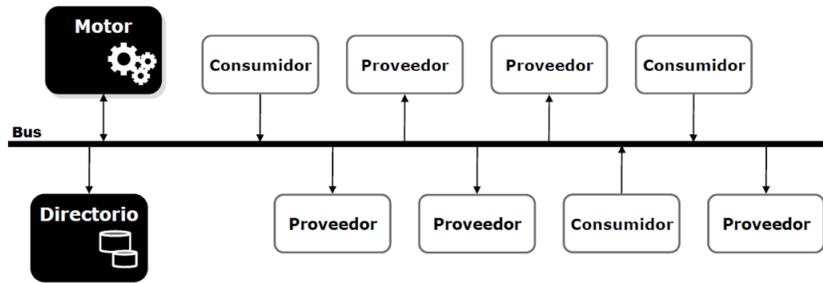


Figura 2.1: Estructura básica de una SOA

El bus de servicios permite la interconexión entre todos los participantes y lleva además un registro de las interacciones que se están realizando. En otras palabras administra y facilita las interacciones entre todos los involucrados.

El repositorio de servicios es un componente opcional (pero altamente recomendado), que almacena y pone a disposición los WS para que otros los consuman. Aunque estos mismos WS pueden almacenarse directamente en los proveedores de servicio, el repositorio provee un directorio dinámico de WS que indica quién es el actual proveedor de ese servicio, o si el WS requerido es parte de los servicios que éste almacena. El contar con este directorio facilita enormemente el acceso a parte de los consumidores, permitiendo la reubicación dinámica de servicios sin necesidad de que los consumidores tengan que cambiar (ajustar) su código fuente.

Finalmente el motor de integración es el encargado de componer (integrar) los WS de manera apropiada, para obtener así resultados o servicios más complejos. El motor procesa recetas para la composición de servicios (especificadas a través de scripts), integrando y componiendo ordenadamente los resultados entregados por los WS invocados. A este proceso se lo conoce como “orquestación de servicios”.

Este tipo de arquitectura apunta a la reutilización de componentes (WS) y a la integración de éstos según las necesidades de la organización. En el caso del DCC este tipo de arquitectura es muy adecuada, puesto que desacopla la funcionalidad de los sistemas y les permite convivir de manera razonable, es decir, manteniendo el control sobre la evolución de los mismos y sobre las fuentes de verdad de la información. La figura 2.2 muestra un ejemplo esquemático de cómo sería la integración de servicios en una SOA, por ejemplo en el caso del DCC.

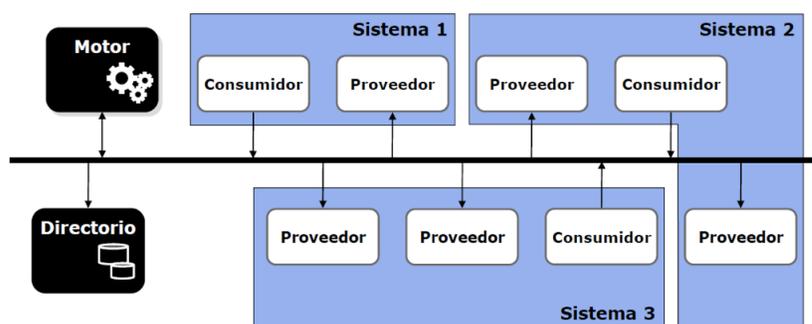


Figura 2.2: Escenario multi-sistema apoyado por una SOA

### 2.1.1. Servicios web

Existen muchas formas de describir qué es un servicio Web. Por ejemplo, una primera definición se refiere a ellos como “un conjunto de aplicaciones o de tecnologías con capacidad para interoperar en la Web” [19]. Estas aplicaciones o tecnologías intercambian datos entre sí, con el objetivo de ofrecer servicios unos a otros. Los proveedores ofrecen sus servicios como procedimientos remotos, y los consumidores solicitan un servicio llamando a estos procedimientos a través de la Web.

Una segunda definición de WS dice que “un servicio Web es un sistema de software identificado por una URI (Uniform Resource Identifier), cuyas interfaces públicas y enlaces son definidos y escritos usando XML (eXtended Markup Language)” [1]. Su definición puede ser descubierta por los sistemas de software que pretenden consumir dicho WS. Los consumidores pueden luego interactuar con el WS de acuerdo a la forma descrita en su definición, utilizando mensajes basados en XML enviados por protocolos de Internet.

Una tercera definición indica que “un servicio Web es un sistema de software diseñado para apoyar interacciones interoperables entre máquinas a través de una red” [4]. Los WS cuentan con una interfaz descrita en un formato procesable por una máquina (particularmente en WSDL -Web Services Description Language-). Las aplicaciones que interactúan con el WS lo hacen de la manera prescrita por su descripción, y usualmente mediante mensajes SOAP (Simple Object Access Protocol, unidad básica de comunicación entre servicios). Típicamente los mensajes son enviados por HTTP con una serialización en conjunto con otros estándares relacionados con la Web.

A partir de las definiciones previas, podemos imaginar a un servicio Web como un tipo particular de SOA, donde sus componentes se conectan remotamente vía Web.

### 2.1.2. Lenguajes de especificación de interfaces para servicios web

Existen dos lenguajes ampliamente reconocidos para la descripción y descubrimiento de servicios Web: WSDL (Lenguaje de Definición de servicios Web) y UDDI (Descubrimiento, Descripción e Integración Universal) [5]. WSDL es un estándar implantado por la W3C, el cual describe a los servicios usando seis clases de elementos y está implementado en XML.

Por otra parte, el estándar UDDI es una especificación diseñada para permitir a los procesos (en nuestro caso, de negocios), encontrarse y compartir información. El directorio UDDI es global, público y está en línea. Esto permite a los procesos de una organización (por ejemplo, a los del DCC) describir sus servicios manera uniforme, descubrir los servicios que proveen las diferentes unidades con las que la organización interactúa (por ejemplo, los servicios de la Facultad) y comprender los métodos necesarios para manejar los procesos particulares. Este estándar funciona como un directorio telefónico, almacenando *metadata* acerca del negocio. Hoy en día podemos ver que estos dos lenguajes se usan de manera complementaria. Por ejemplo, la especificación de servicios Web suele describirse usando WSDL, y luego se pone a disposición el WS en Internet o en una Intranet a través de UDDI sin ser esto estrictamente necesario.

## 2.2. Desarrollo basado en microservicios

En los últimos 3 o 4 años se ha acentuado a nivel mundial el desarrollo de aplicaciones Web como microservicios. Esta estrategia de desarrollo es una variante de SOA, con énfasis en la creación de pequeños sistemas desacoplados e intercomunicados, lo que conlleva múltiples beneficios en comparación a las aplicaciones Web monolíticas tradicionales [7]. Entre las ventajas más importantes de usar microservicios está la capacidad de responder de manera rápida a las nuevas necesidades de los usuarios, ya que el proceso de desarrollo, comprobación y distribución de estas aplicaciones es mucho más expedito. Además, permite la independencia de las tecnologías a emplear en cada uno de los componentes, tales como lenguajes de programación, *frameworks* de desarrollo y servicios adicionales que pudiesen ser necesarios, sin repercutir en otros componentes.

Uno de los requisitos fundamentales de la arquitectura de microservicios es que todos sus componentes puedan comunicarse entre sí, bajo un protocolo en común. En la actualidad la mayoría de los servicios realizan esto mediante peticiones a un conjunto determinado de *endpoints* HTTP, que corresponden a la *Application Programming Interface* (API) de los sistemas, que generalmente proporcionan información en formato JSON. Estos endpoints pueden ser de tipo público o privado; para estos últimos se requiere autenticación a través de mecanismos tales como OAuth [12] o JSON Web Tokens [14]. Este esquema de trabajo permite además la comunicación entre aplicaciones de escritorio y móviles, sin duplicar esfuerzos de desarrollo.

## 2.3. Estrategias de reingeniería de software

La reingeniería de software consiste en el análisis y modificación de un software existente para generar una nueva implementación. Típicamente, la reingeniería de los sistemas de software busca mejorar diversos aspectos, tales como experiencia de usuario, utilidad, mantenibilidad y confiabilidad, entre otros [18]. Dependiendo de diversos factores se puede estimar el impacto de los cambios en la aplicación, y elegir el mejor camino para realizar el proceso de reingeniería. Estos aspectos a considerar son muy diversos, por ejemplo, si existe documentación del proyecto, si se conoce la estructura o el código fuente del producto a intervenir, si existe información complementaria al código fuente, etc.

En el caso de la reingeniería realizada al sistema SAR, el autor tenía amplio conocimiento del software, y además había documentación de apoyo para saber cómo intervenir el producto. Por lo tanto, la reingeniería del SAR se abordó a paquete completo; es decir, se intervino la arquitectura, la base de datos y la funcionalidad del sistema. La información que da cuenta de ese trabajo quedó registrada a nivel de requisitos y de diseño (tanto arquitectónico como detallado).

## 2.4. Herramientas utilizadas

Existen varios *frameworks* consolidados para crear el *back-end* de microservicios. Uno de ellos es Ruby on Rails<sup>1</sup> (Ruby), que es utilizado por herramientas y sitios Web como por ejemplo GitHub<sup>2</sup>, Airbnb<sup>3</sup> y Kickstarter<sup>4</sup>. Por otra parte también está disponible Django<sup>5</sup> (Python) que también es empleado por reconocidos sitios o herramientas Web, tales como Instagram<sup>6</sup>, Pinterest<sup>7</sup> y Bitbucket<sup>8</sup>. Ambos frameworks han demostrado ser robustos, usables y útiles para llevar a cabo la implementación de aplicaciones Web de alta demanda.

Desde el año 2015, el desarrollo del *front-end* de las aplicaciones Web se ha visto afectado por la proliferación de innumerables herramientas de desarrollo para el lenguaje JavaScript. Si bien éstas cumplen con el objetivo de ayudar a crear el software necesario, agregan complejidad innecesaria, sobre todo por la cantidad de librerías adicionales que son requeridas. Esto último se traduce en aplicaciones proclives a errores por dependencias<sup>9</sup>.

Por otra parte, algunos equipos de desarrollo de software han optado por el camino de usar lenguajes transcompilados como CoffeeScript<sup>10</sup>, ya que es imposible evitar el uso de JavaScript en el *front-end* de las aplicaciones, debido a su presencia en los navegadores Web modernos. Esta decisión tampoco está exenta de problemas, ya que no promueve un estándar de desarrollo, afectando la mantenibilidad de los sistemas de cara al futuro. En este escenario los equipos de ingeniería deben analizar cuidadosamente las tecnologías de *front-end* a utilizar, con el fin de asegurar la fácil evolución de las aplicaciones.

---

<sup>1</sup>Ruby on Rails <http://rubyonrails.org/>

<sup>2</sup>GitHub <https://github.com/about>

<sup>3</sup>Airbnb <https://www.airbnb.cl/about/about-us>

<sup>4</sup>Kickstarter <https://www.kickstarter.com/about>

<sup>5</sup>Django <https://www.djangoproject.com/>

<sup>6</sup>Instagram <https://www.instagram.com/about/us/>

<sup>7</sup>Pinterest <https://about.pinterest.com/>

<sup>8</sup>Bitbucket <https://bitbucket.org/>

<sup>9</sup>How an irate developer briefly broke JavaScript <https://www.theverge.com/2016/3/24/11300840/how-an-irate-developer-briefly-broke-javascript>

<sup>10</sup>CoffeeScript <http://coffeescript.org/>

# Capítulo 3

## Sistema administrador de recursos

A continuación se describe brevemente la situación del SAR versión 1.0 al momento de iniciar este trabajo de memoria. Luego se muestra en detalle las especificaciones actualizadas de la versión 2.1 desarrollada en esta memoria, usando como base a la versión 2.0 de la aplicación [15]. Cabe destacar que esta última fue creada utilizando la metodología de desarrollo incremental, vista en el contexto del curso Ingeniería de Software II del DCC.

La versión 2.0 del SAR desarrollada desde cero en el curso Ingeniería de Software II durante el semestre de otoño 2016, obtuvo resultados exitosos. Sin embargo no se logró poner en producción ya que no fue probado en su totalidad por los usuarios finales. Además, carecía de los métodos de comunicación necesarios para que sistemas externos pudiesen interoperar con él, dado que eso no estaba dentro del alcance del proyecto. Es importante aclarar que el autor de esta memoria fue el líder del equipo de desarrollo que creó el SAR 2.0, y además fue el responsable (junto a su profesor guía) de realizar la reingeniería de dicho producto, con el fin de obtener la solución reportada en este documento; es decir el SAR 2.1.

### 3.1. Situación inicial

Hasta principios de 2017, el DCC mantenía en producción un sistema Web creado el año 2003 por estudiantes del curso Ingeniería de Software II, que permitía administrar sus recursos en línea (principalmente salas, *notebooks* y proyectores) y reservarlos, según su disponibilidad. A pesar de la utilidad de dicha solución, ésta tenía diversas limitaciones, como por ejemplo:

- Las tecnologías utilizadas en su implementación estaban obsoletas (PHP versión 4, MySQL versión 3 y HTML versión 4).
- Su modelo de datos no permitía la asignación de puestos de trabajo para estudiantes de posgrado y profesores visitantes. Por lo tanto, el manejo de esos recursos se hacía de manera manual (en papel).
- A pesar de que este sistema fue diseñado para ser utilizado por académicos y funcio-

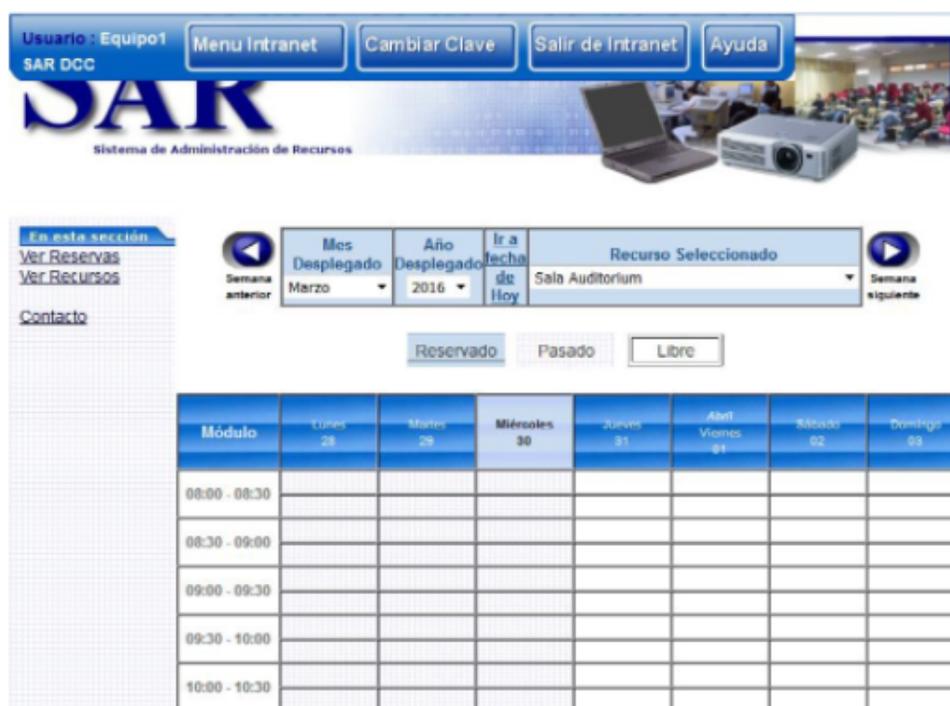


Figura 3.1: Interfaz principal del SAR v1.0

narios, su dependencia de la Intranet del Departamento<sup>1</sup> hacía que la gran mayoría de las reservas fuesen realizadas sólo por las secretarías, generando dependencia de ellas a la hora de reservar un recurso.

El sistema SAR v1.0 ha brindado soporte a un conjunto de usuarios determinados, los cuales podían hacer reservas (como por ejemplo, las secretarías y los académicos), y otros que podían además administrar los recursos (es decir, crear, modificar o eliminar recursos) como es el caso de los administradores.

## 3.2. Principales requisitos de la solución

La versión 2.1 del SAR implementa todas las funcionalidades consideradas en la versión 1.0, y además agrega las siguientes:

- Permite administrar puestos de trabajo en las oficina de estudiantes de doctorado, profesores visitantes y postdoctorandos. Los puestos pertenecen a sus respectivas oficinas, y son indistinguibles unos de otros dentro de la misma sala.
- Permite utilizar bloques de 15 minutos para asignar un recurso en el tiempo, a excepción de los puestos de trabajo, que son reservados por bloques de un día completo. Esto hace más eficiente el uso que se le da a los espacios físicos compartidos; por ejemplo, a las salas de reuniones o auditorios.

<sup>1</sup>Intranet DCC <https://intranet.dcc.uchile.cl/>

- Permite crear nuevos recursos y usuarios desde la aplicación, a través de la vista del administrador del sistema.
- Permite que usuarios reserven por otros usuarios o personas invitadas (o sea, a nombre de otro), dejando un registro de esta información al momento de reservar.
- Exige una descripción del motivo de cada reserva antes de realizarla. Esto permite saber para qué se utilizan mayoritariamente los espacios del DCC, y eventualmente hacer planificación sobre el espacio disponible en función de esta información.
- Avisa mediante un *e-mail* a un usuario cuya reserva esté siendo solicitada por otro usuario; es decir, cuando existe una superposición de reservas.
- Permite modificar o cancelar una reserva ya publicada. Esto es posible de realizarse sobre las reservas propias; en otro caso, sólo usuarios con más prioridad pueden modificar o cancelar reservas de otros. En este último caso el sistema notifica a todos los directamente involucrados, tanto en la reserva como en la operación de cancelación.
- Entrega poder de reserva sobre diferentes tipos de recursos a diferentes tipos de usuarios. Esto se hace a través de un sistema de permisos que considera prioridades sobre los recursos compartidos.
- Genera diferentes vistas (interfaces de usuario) dependiendo de los permisos de cada persona.
- Permite visualizar de diferentes formas el estado de los recursos según su nivel de uso u ocupación.

Dentro del marco del trabajo de memoria se establecen las siguientes restricciones a la solución:

- No se implementarán prioridades de reserva para usuarios de una misma categoría (por ejemplo “académico”). En estos casos el único orden que se respeta es el orden de llegada.
- Deben mantenerse las funcionalidades del SAR 1.0, sin cambiar de manera radical su forma de operación (paradigma de interacción con el usuario).

### 3.3. Usuarios de la solución

Los usuarios del sistema están separados en dos categorías: usuarios directos e indirectos. A continuación se describen cada uno de ellos.

#### Usuarios directos

- **Jefa de estudios:** Puede distribuir de forma fácil y eficiente a los alumnos de postgrado del DCC en las oficinas asignadas para este fin. En otras palabras, asignarle puestos de trabajo a dichos estudiantes. Además, puede reservar recursos como cualquier otro usuario.
- **Secretaria de investigación:** Puede asignar profesores visitantes y postdoctorandos a puestos de trabajo en las oficinas destinadas para tal fin.

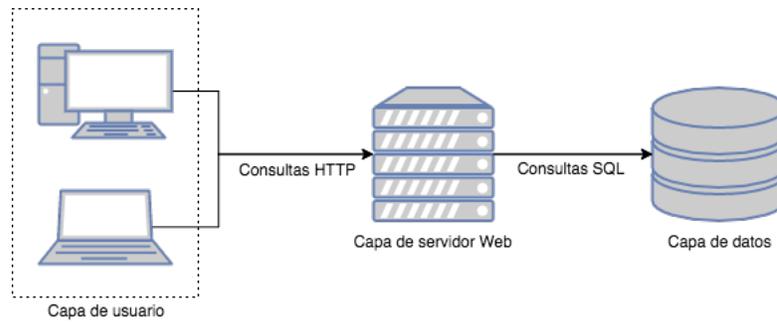


Figura 3.2: Arquitectura física de 3 capas del SAR v2.1

- **Administrador de sistemas:** Puede administrar todo el sistema, incluyendo el agregar o eliminar recursos del SAR.
- **Académico:** Puede realizar reservas de recursos.

### Usuarios indirectos

- **Alumno de postgrado:** El sistema le permite contar con un espacio de forma fija por la duración de su estadía en el departamento.
- **Profesor invitado o postdoctorando:** El sistema le permite agilizar el proceso de asignación de un espacio temporal durante su estadía al DCC.

## 3.4. Arquitectura de la solución

A continuación se presenta la arquitectura física y la arquitectura lógica del SAR 2.1.

### 3.4.1. Arquitectura física

Este sistema está diseñado para funcionar como una aplicación Web estándar, sustentada en el modelo de tres capas: usuario, servidor web y servidor de datos. En la figura 3.2 se muestra la interacción entre cada una de éstas.

**Capa de usuario:** Compuesta por cualquier dispositivo capaz de ejecutar un navegador Web, que cuente con una conexión a Internet. En este caso el dispositivo se contacta con el servidor Web mediante el protocolo HTTP seguro.

**Capa de servidor web:** Esta capa es la responsable de responder a las solicitudes (*requests*) de los usuarios. Se rige bajo la lógica de la aplicación, a la hora de manejar las interacciones entre las capas. Además, se comunica con los siguientes componentes:

- **Capa de servidor de datos:** Realiza consultas SQL con el fin de obtener información necesaria para responder las peticiones.

- **Intranet DCC:** Efectúa consultas de identificación dedicadas exclusivamente a la autenticación de usuarios.

**Capa de servidor de datos:** Esta capa es la responsable de almacenar la información del sistema, de acuerdo al modelo de datos creado. Utiliza PostgreSQL versión 9.5 o superior como motor de gestión, y sólo interactúa con el servidor Web respondiendo sus consultas SQL. Tanto la capa de servidor Web como la del servidor de datos, se encuentran en dependencias del DCC.

### 3.4.2. Arquitectura lógica

A continuación se describe brevemente las funcionalidades disponibles (capacidades) para cada tipo de usuario descritos en la subsección 3.3.

#### Capacidades del usuario académico

El *Académico* es el usuario base del sistema. Éste tiene permisos para revisar la disponibilidad de horarios de todos los recursos comunes del Departamento (conformados por salas de reuniones y auditorios principalmente) y para realizar reservas en horarios libres. En el caso de necesitar un recurso ya reservado, el académico puede realizar una solicitud al dueño de la reserva original para liberar el horario. De forma adicional puede acceder a un registro que contiene la información de sus propias reservas, donde podrá eliminarlas o consultar información respecto a ellas.

#### Capacidades de los usuarios jefe de estudios y secretaria de investigación

Estos dos tipos de usuario cuentan con acceso a todas las funcionalidades del usuario académico, pero además pueden administrar los recursos catalogados como “oficinas compartidas” que corresponden a recursos restringidos. En el caso de la *jefa de estudios* esos recursos corresponden a oficinas de estudiantes de postgrado, y en el caso de la *secretaria de investigación* los recursos corresponden a oficinas de postdoctorandos y profesores visitantes. La administración de los recursos restringidos incluye permisos para consultar su disponibilidad, sumado a la capacidad para realizar asignaciones sobre estos y eliminarlas.

#### Capacidades del usuario administrador de sistema

El *Administrador del sistema* cuenta con acceso a todas las funcionalidades del usuario académico. De forma adicional es capaz de administrar la lista de usuarios de la aplicación, por lo tanto tiene la facultad de agregar un usuario, modificar sus permisos o eliminarlo del sistema. También tiene la facultad de agregar, modificar o eliminar recursos del registro, y marcarlos como compartidos o restringidos, sin la necesidad de trabajar directamente sobre el servidor de datos.

## 3.5. Módulos del sistema

La figura 3.3 muestra los principales módulos del SAR versión 2.1 y la relación entre ellos. El módulo principal es el *MD0001 - Módulo de gestión de vistas* que es el que permite la interacción directa con otros módulos e implementa parte de la lógica de coordinación entre módulos. A continuación se especifica la funcionalidad que implementa cada uno de ellos.

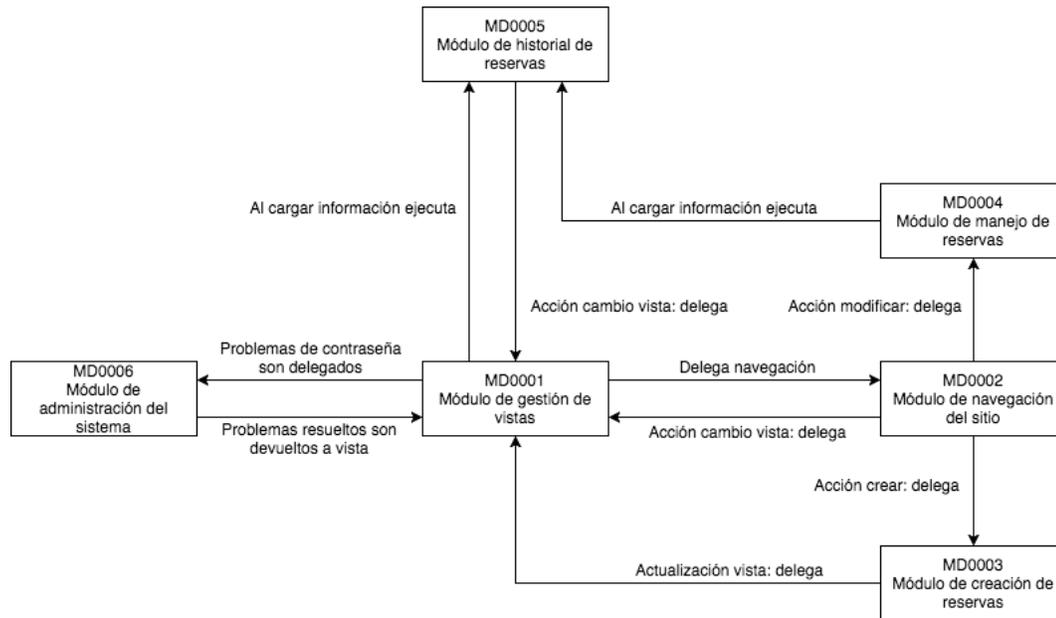


Figura 3.3: Módulos del SAR v2.1

### MD0001 - Módulo de gestión de vistas

Este módulo es el encargado de la implementación y manejo de las vistas de la aplicación. El proceso que lleva a cabo este módulo es el siguiente:

- El usuario ingresa al sitio a través de un login.
- En caso de que el usuario tenga problemas con la contraseña, entonces se lo delega al módulo MD0006 - Administración del sistema, para su manejo.
- Permite la generación dinámicamente vista de reservas, leyendo la información proporcionada por el módulo MD0005 - Módulo de historial de reservas.
- La navegación de cada vista se delega al módulo MD0002 - Módulo de navegación del sitio.

### MD0002 - Módulo de navegación del sitio

Este módulo es el encargado de implementar y mantener todas las herramientas que facilitan la navegación y el uso de la plataforma SAR; en otras palabras, permite que los

usuarios puedan acceder directamente a lo que necesitan sin mayores complicaciones. El proceso que apoya este módulo es el siguiente:

- Permite a un usuario interactuar con todas las barras de menú, buscadores y botones del sistema SAR.
- Si la acción es creación de reservas, entonces lo delega al módulo MD0003 - Módulo de creación de reservas.
- Si la acción es modificación de reserva, entonces lo delega al módulo MD0004 - Módulo de manejo de reservas.

### **MD0003 - Módulo de creación de reservas**

Se encarga de todo lo relacionado a la creación de las reservas, a fin de que éstas sean lo más completas posible y se puedan almacenar correctamente en la base de datos. El proceso que realiza este módulo es el siguiente:

- Para ingresar una nueva reserva al SAR se debe rellenar toda la información solicitada en el formulario (Web) de reserva.
- Al terminar de ingresar la información solicitada, ésta se guarda en la base de datos del sistema.
- Finalmente se delega la actualización de la vista, al módulo MD0001 - Módulo de gestión de vistas.

### **MD0004 - Módulo de manejo de reservas**

Este módulo es el encargado llevar a cabo las modificaciones realizadas a las reservas (siempre que corresponda), contemplando desde modificar la reserva misma, hasta eliminarla para siempre, evitando problemas de colisión con otras reservas y con la base de datos. El proceso que realiza este módulo es el siguiente:

- Recupera la información de las reservas desde la base de datos del sistema. Esta operación es ejecutada por el módulo MD0005 - Módulo de historial de reservas.
- Despliega la información en pantalla y permite que el usuario seleccione la reserva a cambiar.
- Una vez que el usuario seleccionó una reserva, el módulo carga la información en un formulario Web y le permite (si corresponde) actualizar los campos requeridos, o bien eliminar la reserva. Estos cambios se guardan en la base de datos.
- Finalmente se lleva a cabo la actualización de vista a través del módulo MD0001 - Módulo de gestión de vistas.

## MD0005 - Módulo de historial de reservas

Este módulo se encarga de recopilar la información de las reservas para un cierto rango de tiempo, y luego mostrarlas de manera amigable. El módulo también provee estadísticas de uso de espacios en oficinas. El proceso que se lleva a cabo es el siguiente:

- Al igual que en el caso anterior, la primera acción es recuperar la información de las reservas desde la base de datos.
- Se compara la información de las reservas contra la fecha (o el rango de fechas) seleccionada para distinguir las reservas anteriores de las futuras.
- Si se requiere generar estadísticas, éstas se calculan y luego se delega el trabajo de visualización al módulo MD0001 - Módulo de gestión de vistas.

## MD0006 - Módulo de administración del sistema

Este es el encargado de los temas administrativos internos del SAR, y está directamente relacionado con la implantación del sistema. Contempla los permisos para administrar el sistema, portar bases de datos, implantar el sistema en el servidor de producción y gestionar el cambio de contraseñas para los usuarios del sistema (específicamente la recuperación de éstas). El proceso que lleva a cabo este módulo es el siguiente:

- Si el usuario entra por problemas de contraseña, entonces se ejecutan las operaciones requeridas para poder recuperar dicha clave, y luego se actualiza la vista delegando ese trabajo al módulo MD0001 - Módulo de gestión de vistas.
- Si el usuario es el administrador, entonces el módulo le entrega una vista específica para él, la cual es estática y permite ingresar los datos directamente en la base de datos.
- El módulo permite además editar los recursos y sus tipos. A esto lo hace de forma directa y sin delegar el trabajo a otros módulos.

## 3.6. Modelo de datos

En la figura 3.4 se muestra el diagrama Entidad-Relación del SAR v2.1. La entidad *Reservation* se encarga de almacenar los datos de la reserva y posee varios atributos: una sala *Room*, un grupo de reservas *ReservationGroup*, un tipo de reserva *ReservationType* y múltiples bloques de reserva *ReservationBlock* de 15 minutos; estos últimos facilitan la consulta de colisiones de reservas en un rango de fechas. Por otro lado, la entidad *Room* contiene la información de las salas disponibles, los grupos de usuarios autorizados para realizar una reserva, una lista de usuarios administradores del recurso y un grupo de salas *RoomGroup*. Los usuarios del sistema están vinculados a la entidad *UserInformation*, la cual permite guardar los datos adicionales del usuario sin modificar la clase de autenticación provista por Django. Las entidades *User*, *Group*, *Permission* y *ContentType* son provistas por el *framework* para realizar la gestión de usuarios, grupos y permisos en el sistema. El detalle de cada entidad se puede apreciar en la figura B.1 del apéndice B.

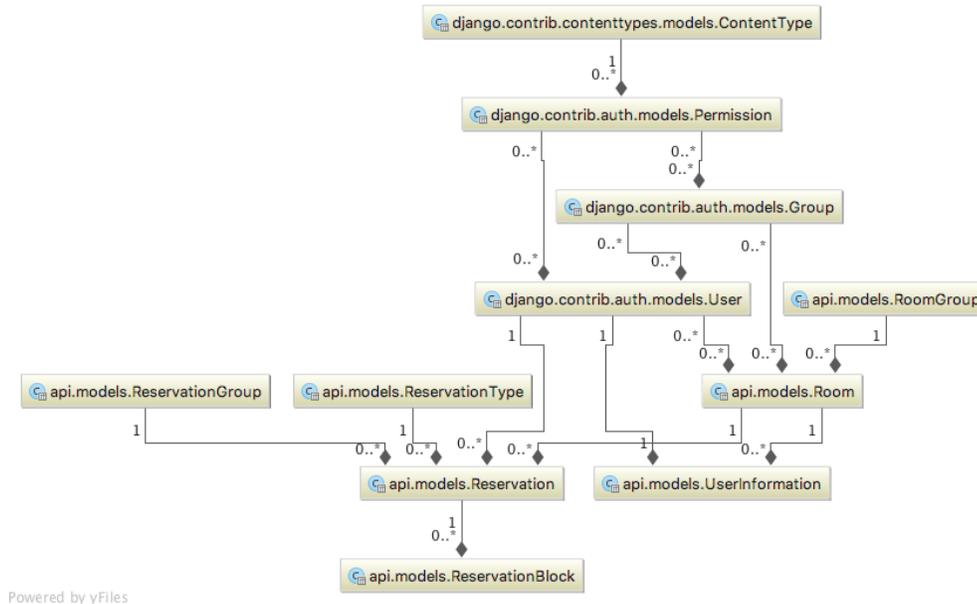


Figura 3.4: Diagrama Entidad-Relación simplificado del SAR v2.1

### 3.7. Cambios internos de la versión 2.1

En esta nueva versión del sistema se actualizaron las dependencias de éste para que pudiera funcionar con tecnologías más modernas, particularmente con la versión del framework Django 1.11 LTS, el motor de bases de datos PostgreSQL versión 9.5.7 y el framework Angular versión 1.6.4. Además, se corrigieron errores importantes del sistema, y se implementaron nuevos métodos de autenticación vía tokens de usuario, utilizando el esquema de autorizaciones de OAuth versión 2.0 [12]. Finalmente, se realizó el refactoring del nombre de clases y funciones del sistema, y se generó una suite de *tests unitarios* que permitieron evaluar el correcto funcionamiento de los servicios, tanto de los nuevos como de los legados (implementados en la versión 2.0).

### 3.8. Principales vistas del sistema

A continuación se presentan las principales interfaces del SAR versión 2.1.

#### 3.8.1. Ingreso al sistema

Como se puede ver en la figura 3.5, la autenticación de los usuarios se realiza a través de la validación de un nombre de usuario y clave. Este proceso consiste en generar una petición de acceso al sistema de correos electrónicos del DCC. Si las credenciales son válidas, entonces el sistema le permite al usuario con los permisos propios de su perfil; de lo contrario se le muestra un mensaje de error.

A screenshot of a login interface. It features a light gray rounded rectangle containing two white input fields. The top field is labeled 'Usuario' and the bottom field is labeled 'Contraseña'. Below these fields is a blue button with the text 'Ingresar' in white.

Figura 3.5: Interfaz de login de usuarios del SAR v2.1

### 3.8.2. Menú principal

En la figura 3.6 se muestra el menú principal de la aplicación, disponible en la parte superior de la interfaz de usuario.

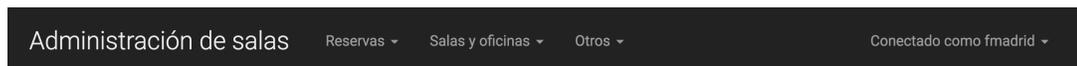


Figura 3.6: Menú principal del SAR v2.1

Componentes interactivos de la figura 3.6:

- La opción de menú rotulada como “Administración de Salas” es un *link* clickeable que redirige al usuario al *home* del sistema.
- La opción de menú identificada como “Reservas” despliega un menú *dropdown* para seleccionar vistas entre reservas de recursos comunes y oficinas.
- La opción de menú rotulada como “Salas y oficinas” despliega un menú *dropdown* para seleccionar vista de salas y oficinas (lista).
- La opción de menú indicada como “Otros” despliega un menú *dropdown* para seleccionar una vista que muestra los usuarios del sistema (lista).
- La opción de menú rotulada como “Conectado como <usuario>” despliega un menú *dropdown* con distintas opciones de configuración relativas al usuario; incluyendo el cambio de palabra clave para la autenticación.

### 3.8.3. Reserva de recursos

En la figura 3.7 se muestra la interfaz de usuario para la reservas de recursos.

Administración de salas   Reservas ▾   Salas y oficinas ▾   Otros ▾   Conectado como fmadrid ▾

Recursos comunes

Semana anterior   29-06-2017   Semana siguiente

Sala: Auditorio Ramon Picarte ⓘ

Módulo	Lunes 26-06-2017	Martes 27-06-2017	Miércoles 28-06-2017	Jueves 29-06-2017	Viernes 30-06-2017	Sábado 01-07-2017	Domingo 02-07-2017
08:00 - 08:30							
08:30 - 09:00							
09:00 - 09:30							
09:30 - 10:00							
10:00 - 10:30							
10:30 - 11:00							
11:00 - 11:30							
11:30 - 12:00							
12:00 - 12:30		12:00 - 13:30 Clases		12:00 - 13:30 Clases			
12:30 - 13:00		Jocelyn Simmonds Presentaciones finales...	12:15 - 14:30 Charla de Invitado Karin Riquelme Díaz Charla empresa Arkano	Jocelyn Simmonds Presentaciones finales...	12:30 - 14:00 Reunión Clave Sandra Gaez Mansilla Reunión con el cuerpo ...		
13:00 - 13:30							
13:30 - 14:00							
14:00 - 14:30							
14:30 - 15:00			14:30 - 16:00 Actividad Social Ana Martinez Eduel Chile Retornal				
15:00 - 15:30							

Figura 3.7: Interfaz de usuario para reservas de recursos

A continuación se explican brevemente las componentes interactivas que se muestran en la figura 3.7:

- Los botones “Semana anterior” y “Semana siguiente” permiten cambiar el periodo de visualización de las reservas en siete días.
- El campo de fecha en el formulario permite ingresar la fecha deseada a través de un calendario que se despliega automáticamente al hacer click sobre ese campo.
- El menú dropdown “Sala” permite seleccionar la sala (oficina, sala de reuniones o auditorio) sobre la cual se pretende hacer una reserva.
- El botón “i” despliega una ventana modal con información de los distintos tipos de reservas.
- El calendario semanal está dividido en bloques de 15 minutos, que es la unidad mínima para la reserva de espacios en el sistema.

### 3.8.4. Detalle de una reserva de recurso

En la figura 3.8 se muestra la interfaz de usuario con el detalle de la reserva de un recurso.

Administración de salas

Detalles reserva

**Sala**  
Auditorio Ramon Picarte

**Reserva asignada a**  
Sandra Gaez Mansilla

**Email**  
sandra@dcc.uchile.cl

**Teléfono**  
+56-229784330

**Fecha Inicio**  
2017-06-30 12:30

**Fecha Fin**  
2017-06-30 14:00

**Descripción**  
Reunión con el cuerpo académico del programa (Acreditación del Doctorado)

**Categoría**  
Reunión Clave

Solicitar reserva Modificar reserva Cancelar reserva Cerrar

14:30 - 15:00  
15:00 - 15:30  
15:30 - 16:00

Actividad Social  
Ana Martinez  
Fútbol, Chile-Portugal, ...

Figura 3.8: Detalle de la reserva de un recurso

A continuación se explican brevemente las componentes interactivas de la figura 3.8:

- El formulario Web muestra múltiples campos con los detalles de la reserva.
- El botón “Solicitar reserva” permite solicitar la reserva al usuario asignado actualmente, o sea, a quien hizo la reserva.
- El botón “Modificar reserva” permite realizar cambios en ésta.
- El botón “Cancelar reserva” elimina la reserva del sistema, previa confirmación del usuario.
- El botón “Cerrar” cierra la ventana modal.

### 3.8.5. Creación de una reserva personal

Si el usuario hace un click sobre un bloque disponible de un recurso, por ejemplo para el auditorio Ramón Picarte como se muestra en la figura 3.7, se le da la posibilidad de reservar ese recurso en ese horario. En la figura 3.9 se muestra la interfaz de creación de una reserva personal de un recurso; en este caso es el mismo auditorio Ramón Picarte.

The image shows a modal window titled "Nueva reserva" (New reservation) overlaid on a calendar interface. The modal contains the following elements:

- Sala seleccionada:** A text field containing "Auditorio Ramon Picarte".
- Para quién desea reservar:** Two radio buttons, "Para mi" (selected) and "Para otro".
- Fecha Inicio:** A text field with "01-07-2017".
- Hora Inicio:** A text field with "12:30".
- Hora Término:** Two spinners showing "12" and "45".
- Repetición:** A dropdown menu with "No" selected.
- Fecha Término:** A text field with a calendar icon.
- Categoría:** A dropdown menu with "Elija una categoría de la lista".
- Descripción:** A large text area.
- Buttons:** "Reservar" (Reserve) and "Cerrar" (Close) at the bottom right.

The background shows a calendar grid with time slots from 12:00 to 15:30. Some slots are occupied by activities like "Clases", "Charla de Invitado", "Presentaciones finales...", "Charla empresa Arkano", "Actividad Social", and "Reunión Clave".

Figura 3.9: Interfaz de creación de una reserva (personal) de un recurso

Las componentes interactivas que se muestran en la figura 3.9 son las siguientes:

- El campo con el nombre de la sala (deshabilitado) corresponde al recurso a reservar.
- Los botones “Para mi” y “Para otro” permiten indicar si la reserva es para quien está reservando (en ese caso la reserva es personal), o o bien es para otra personal (para un tercero).
- El formulario Web muestra múltiples campos para indicar los detalles de la reserva.
- El botón “Reserva” permite efectuar la reserva, registrándola en el sistema.
- El botón “Cerrar” cierra la ventana modal.

### 3.8.6. Creación de una reserva para un tercero

En la figura 3.10 se muestra la interfaz de creación de una reserva de un recurso para un tercero. Este formulario es similar al anterior, sin embargo se pide especificar la persona que va a utilizar el recurso reservado y también sus datos de contacto por las dudas que sea necesario consultar algo con ella.

The image shows a modal window titled "Nueva reserva" with a close button (X) in the top right corner. The form contains the following elements:

- Sala seleccionada:** A text field containing "Auditorio Ramon Picarte".
- Para quién desea reservar:** Two buttons: "Para mí" and "Para otro".
- Nombre de a quién se reserva:** A text input field with a "Usuario Nuevo?" checkbox to its right.
- Teléfono de a quién se reserva (opcional):** A text input field with the placeholder "Formato +56 123456789".
- Correo de a quién se reserva:** A text input field.
- Fecha Inicio:** A text input field with "01-07-2017".
- Hora Inicio:** A text input field with "12:30".
- Hora Término:** Two spinners for hours and minutes, currently showing "12" and "45".
- Repetición:** A dropdown menu with "No" selected.
- Fecha Término:** A text input field with a calendar icon.
- Categoría:** A dropdown menu with the text "Elija una categoría de la lista".
- Descripción:** A large text area for notes.
- Buttons:** "Reservar" (blue) and "Cerrar" (white) at the bottom right.

Figura 3.10: Interfaz de creación de reserva para un tercero

Las componentes interactivas que se muestran en la figura 3.10 son las siguientes:

- Similar al formulario anterior, el campo con el nombre de la sala (deshabilitado) indica el recurso que está siendo reservado.
- Los botones “Para mí” y “Para otro” permiten cambiar entre una reserva personal o para un tercero, y cambiar el formulario de reserva en base a eso.
- Múltiples campos para indicar detalles de la persona asignada para la reserva; o sea, aquel que va a hacer uso del recurso.
- Múltiples campos para indicar detalles de la reserva.
- El botón “Reserva” permite efectuar la reserva en el sistema.
- El botón “Cerrar” cierra la ventana modal.

### 3.8.7. Reserva de puestos de trabajo

Tal como se mencionó antes, las oficinas son recursos restringidos que sólo pueden ser reservados por usuarios con roles especiales. Por ejemplo la *jefa de estudios* puede reservar puestos de trabajo en oficinas de estudiantes de postgrado, y la *secretaria de investigación* puede reservar puestos de trabajo en oficinas de postdoctorandos y profesores visitantes. En la figura 3.11 se muestra la interfaz de reserva de puestos de trabajo en oficinas.

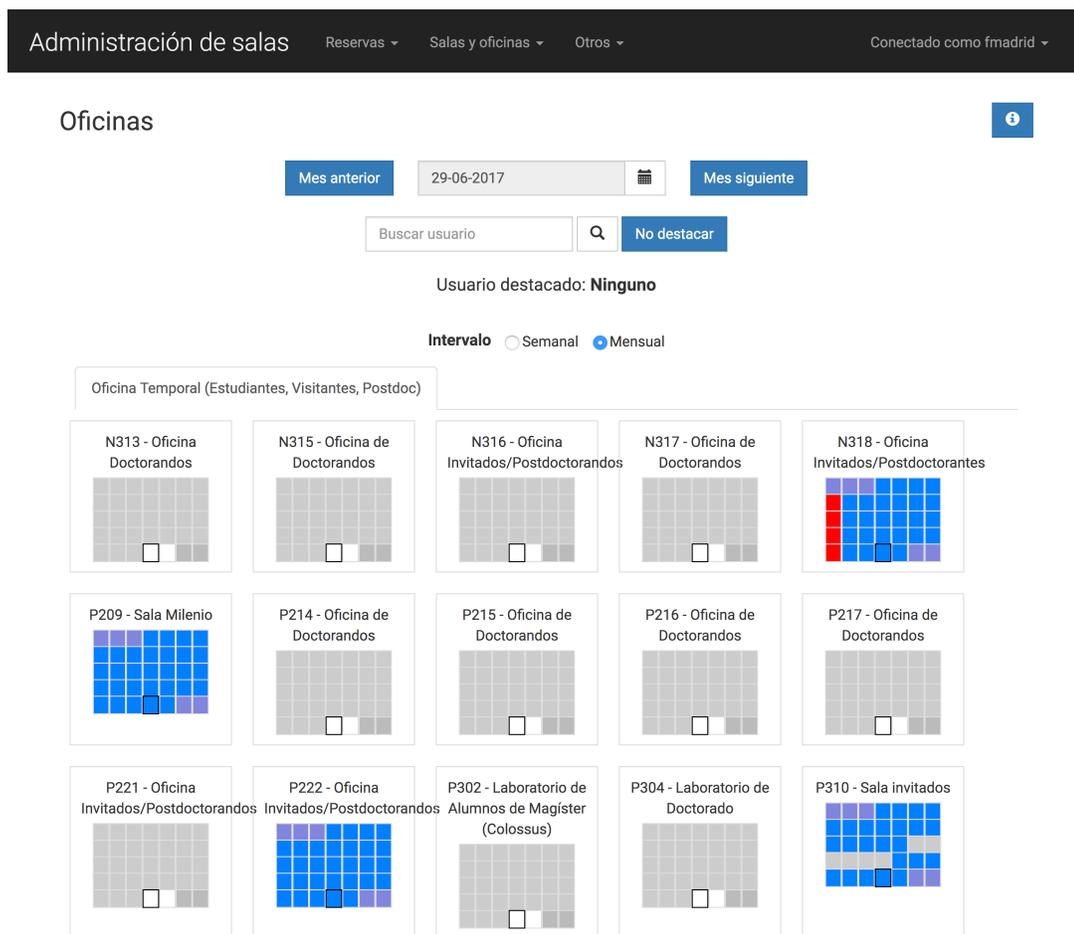


Figura 3.11: Interfaz de reserva de puestos de trabajo en oficinas

Las componentes interactivas que se muestran en la figura 3.11 son las siguientes:

- El botón “i” despliega una ventana modal con la información que indica el significado de los distintos colores usados en el calendario.
- Los botones “Mes anterior” y “Mes siguiente” permiten cambiar el periodo de las reservas (en 1 mes) a visualizar por el usuario.
- El campo con la fecha y el calendario emergente son dos controles que le permiten al usuario seleccionar un mes específico a visualizar.
- El campo “Buscar usuario”, como su nombre lo indica, permite buscar las reservas que ha hecho un usuario dentro del período seleccionado.
- El botón “No destacar” borra el resultado de la búsqueda de reservas de un usuario en específico.

- Los selectores de intervalo permiten cambiar el periodo de reservas entre semanal y mensual. En función de eso será la visualización de la información que el sistema le entregará al usuario.
- Calendarios diarios con la información de la cantidad de puestos de trabajo reservados, desplegada mediante una descripción emergente. El color de los bloques en cada día depende del tipo de ocupación de los puestos de trabajo. Por ejemplo, los bloques en gris indica tiempo pasado (o sea, días anteriores a la fecha actual), los bloques blancos indican que la oficina está vacía, azul indica que está ocupada pero no completamente, y rojo indica ocupación completa).

### 3.8.8. Lista de salas y oficinas

En la figura 3.12 se muestra el listado de las salas y oficinas registradas en el sistema; cada una con la información correspondiente. Esta vista está disponible para todos los usuarios del sitio y se puede acceder a ella mediante un hipervínculo del menú principal.

The screenshot shows a web interface with a dark header bar containing the title 'Administración de salas' and navigation links: 'Reservas', 'Salas y oficinas', and 'Otros'. On the right side of the header, it says 'Conectado como fmadrid'. Below the header, the page title 'Lista de salas y oficinas' is displayed. The main content is a table with the following columns: 'Nombre', 'Tipo', 'Capacidad', 'Descripción', 'Grupos de usuario autorizados', 'Administradores', and 'Teléfono'. The table lists six entries: Auditorio Ramon Picarte, Laboratorio Lorenzo, N307 - Of. Claudio Gutierrez, N308 - Of. Alex Bergel, N313 - Oficina Doctorandos, and N315 - Oficina de Doctorandos.

Nombre	Tipo	Capacidad	Descripción	Grupos de usuario autorizados	Administradores	Teléfono
Auditorio Ramon Picarte		60	.	<ul style="list-style-type: none"> <li>• Postgrado</li> <li>• Investigación</li> <li>• Milenio</li> <li>• Funcionarios</li> <li>• Académicos</li> <li>• Aplicaciones Externas</li> </ul>	<ul style="list-style-type: none"> <li>• Hernán Cuevas</li> </ul>	.
Laboratorio Lorenzo		30	.	<ul style="list-style-type: none"> <li>• Postgrado</li> <li>• Investigación</li> <li>• Milenio</li> <li>• Funcionarios</li> <li>• Académicos</li> </ul>	<ul style="list-style-type: none"> <li>• Hernán Cuevas</li> </ul>	.
N307 - Of. Claudio Gutierrez		1	Oficina de Claudio Gutierrez		<ul style="list-style-type: none"> <li>• Rodrigo Seguel Gallardo</li> <li>• Hernán Cuevas</li> <li>• Sergio Aguilera</li> </ul>	+56 22 9780658
N308 - Of. Alex Bergel		1	Oficina de Alexandre Bergel		<ul style="list-style-type: none"> <li>• Rodrigo Seguel Gallardo</li> <li>• Hernán Cuevas</li> <li>• Sergio Aguilera</li> </ul>	+56229784956
N313 - Oficina Doctorandos		2	Oficina de Doctorandos	<ul style="list-style-type: none"> <li>• Postgrado</li> </ul>	<ul style="list-style-type: none"> <li>• Angelica Aguirre</li> <li>• Sandra Gaez Mansilla</li> </ul>	+56 22 9780883
N315 - Oficina de Doctorandos		3	Oficina de Doctorado	<ul style="list-style-type: none"> <li>• Postgrado</li> </ul>	<ul style="list-style-type: none"> <li>• Angelica Aguirre</li> <li>• Sandra Gaez</li> </ul>	+56 22 9780873

Figura 3.12: Interfaz de lista de salas y oficinas

Las componentes interactivas de la interfaz de usuario mostrada en la figura 3.12 son sólo los encabezados de la tabla son clickeables permitiendo el ordenamiento automático

de la información según la columna seleccionada. En cada columna se muestra información relevante, la cual es usualmente empleada por el usuario para determinar si un cierto recurso tiene las características que él o ella necesita para realizar su actividad (por ejemplo, una reunión de trabajo).

### 3.8.9. Panel de administración

En la figura 3.13 se muestra la interfaz del panel de administración de la aplicación SAR.

Administración de salas BIENVENIDO/A, FRANCISCO VER EL SITIO / TERMINAR SESIÓN

Panel de administración

API	
Bloques de reserva	+ Añadir ✎ Modificar
Grupos de salas	+ Añadir ✎ Modificar
Informaciones de usuarios	+ Añadir ✎ Modificar
Reservas	+ Añadir ✎ Modificar
Salas	+ Añadir ✎ Modificar
Tipos de reserva	+ Añadir ✎ Modificar

AUTENTICACIÓN Y AUTORIZACIÓN	
Grupos	+ Añadir ✎ Modificar
Usuarios	+ Añadir ✎ Modificar

DJANGO OAUTH TOOLKIT	
Access tokens	+ Añadir ✎ Modificar
Applications	+ Añadir ✎ Modificar
Grants	+ Añadir ✎ Modificar
Refresh tokens	+ Añadir ✎ Modificar

Acciones recientes

Mis acciones

- ✎ Auditorio Ramon Picarte  
Sala
- + DCC Comunicaciones - Produccion  
Application
- ✎ comunicaciones  
Usuario
- ✎ comunicaciones  
Usuario
- + comunicaciones  
Usuario
- + Aplicaciones Externas  
Grupo
- ✎ dperovic  
Información de usuario
- ✎ ppoblete  
Información de usuario
- ✎ kriquelm  
Información de usuario
- ✎ jsimmond  
Información de usuario

Figura 3.13: Interfaz de administración de recursos y usuarios

Esta vista permite administrar todos los registros del sistema mediante el panel incorporado en el framework Django, y está sólo disponible para los usuarios de tipo Administrador. Según el recurso que se seleccione, se accede a las pantallas de administración propias de ese recurso.

# Capítulo 4

## Sistema administrador de noticias y eventos

En este capítulo se describe el sistema administrador de noticias y eventos del DCC (SANE), el cual fue construido desde cero. Como se mencionó anteriormente, el objetivo de este sistema es ser la única fuente de verdad en torno a este tipo de información. Con esto se busca que todos los sistemas que muestran o distribuyen información sobre noticias y eventos del Departamento, tomen la información desde el SANE.

Por otra parte, el personal del área de comunicaciones del Departamento, que debe lidiar diariamente con diversos tipos de información y canales de distribución, ya no tendrá que hacerlo a partir de la entrada en producción de este sistema. Con el SANE en producción bastará con registrar la información en el sistema, y éste se encargará de entregarla a los consumidores de la misma; por ejemplo, las pantallas públicas del DCC, la página Web oficial y los canales de distribución de la misma (Twitter, Facebook y correo). Si bien el sistema SANE todavía no está en producción, esto se debe únicamente a que los mecanismos de difusión de la información (que son parte de otro trabajo de memoria) aún no están terminados. A continuación se presenta la situación inicial del DCC en este ámbito, y luego se describe la aplicación implementada.

### 4.1. Situación inicial

Actualmente, el DCC realiza la gestión y la difusión de las noticias y eventos a través de su página Web, y también publicando esa información en los sitios oficiales del Departamento en Facebook y Twitter. Como se puede apreciar en las figuras 4.1 y 4.2 la página del Departamento contiene además otros administradores de contenidos, por lo que realizar la mantención de dicha aplicación es demasiado complejo. Además la arquitectura del sitio Web es monolítica.

Otro punto importante es que el sitio Web utiliza versiones antiguas de librerías para el *front-end*, por lo que no provee capacidades básicas de las aplicaciones actuales, como por

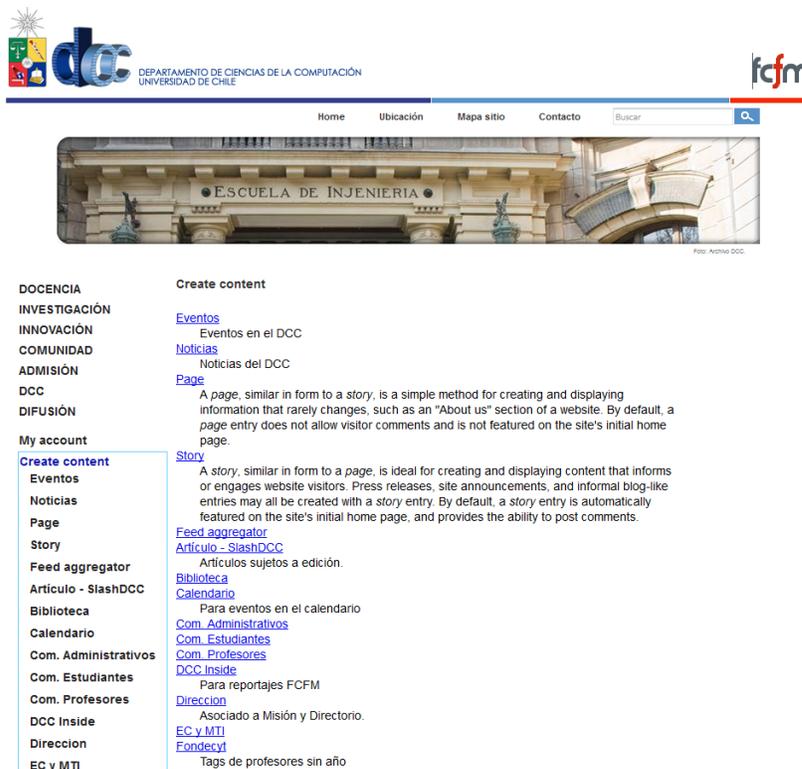


Figura 4.1: Panel de administración de contenidos de la página oficial del DCC

ejemplo el ajuste automático al tamaño de pantalla del dispositivo desde el que se accede. Esta capacidad conocida como *responsiveness* está disponible en la mayoría de los sitios Web, por lo que no contar con ella es una muestra de obsolescencia del portal Web del Departamento.

Por otra parte, la interfaz de creación de noticias y eventos sólo permite un formato (estética) para mostrar dicha información, lo cual limita el estilo de despliegue del contenido a publicar.

## 4.2. Metodologías y herramientas de desarrollo empleadas

Debido a la cercanía con los usuarios y con la contraparte técnica, en el desarrollo de este sistema se empleó un desarrollo ágil. Cada semana se realizaron reuniones con los funcionarios del Área de Comunicaciones para evaluar las interfaces de usuario que se iban diseñando. Además, una vez al mes se efectuaron reuniones técnicas con el Prof. Daniel Perovich, jefe del Área de Desarrollo Estratégico del DCC, para determinar las características y ajustar las propuestas de diseño para la arquitectura de microservicios esperada. En las primeras reuniones se realizaron levantamientos informales de requisitos y pruebas de concepto empleando un prototipo del sitio.

El desarrollo del software se llevó a cabo particularmente bajo la filosofía de Extreme

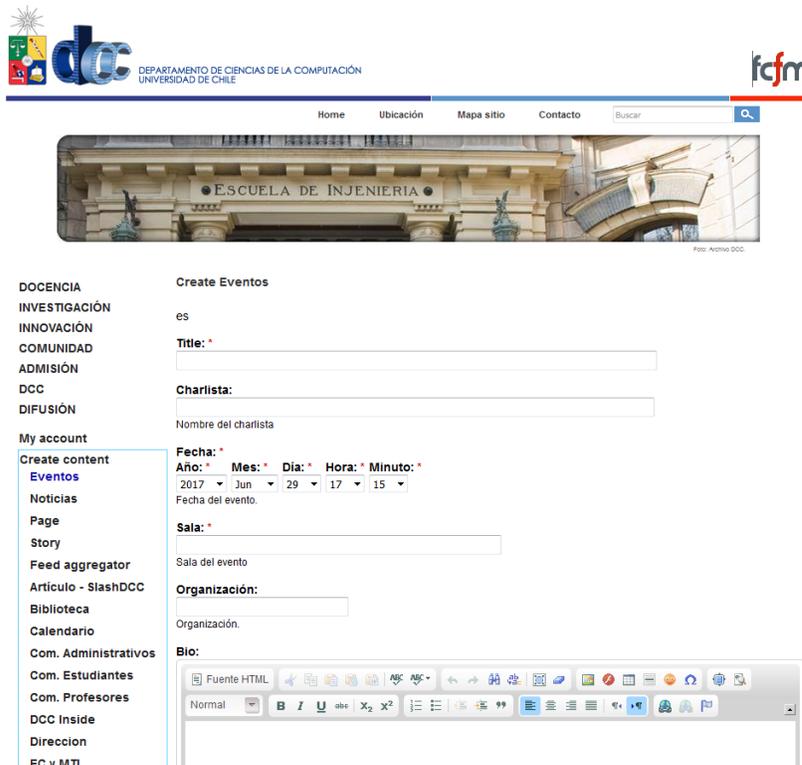


Figura 4.2: Panel de creación de eventos de la página oficial del DCC

Programming [2] (XP), la cual promueve principios y valores para mejorar la calidad del software y el tiempo de respuesta ante las necesidades de los clientes. Para el versionamiento del código se usó git-flow, basado en el modelo de *branching* de Vincent Driessen [8]. Además se emplea la metodología de programación Test Driven Development [3] (TDD) para garantizar el correcto funcionamiento de los artefactos de software durante las diversas etapas del desarrollo.

Para asegurar la mantenibilidad del sitio y la optimización del registro de la información de éste por parte de las máquinas de búsqueda (SEO - Search Engine Optimization [10]), se tomó la decisión de generar las vistas HTML con el sistema provisto por Django en conjunto con el framework AngularJS<sup>1</sup> versión 1.6.4. A pesar de no ser la última versión creada por Google, ésta posee soporte debido a que no requiere el uso de lenguajes transcompilados, por lo que se puede integrar fácilmente a aplicaciones Web existentes sin requerir setups de desarrollo complejos, como en el caso de React<sup>2</sup> de Facebook.

Al igual que el SAR, el SANE también utiliza el framework Django 1.11 LTS, el motor de bases de datos PostgreSQL versión 9.5.7, y los métodos de autenticación vía tokens de usuario bajo el esquema de operación de OAuth 2.0. Adicionalmente, como se puede apreciar en la figura A.8 del apéndice A, se genera la documentación del proyecto en formato HTML utilizando la librería Sphinx<sup>3</sup> de Python.

<sup>1</sup>AngularJS <https://angularjs.org/>

<sup>2</sup>React <https://facebook.github.io/react/>

<sup>3</sup>Sphinx <http://www.sphinx-doc.org/>

### 4.3. Arquitectura de la solución

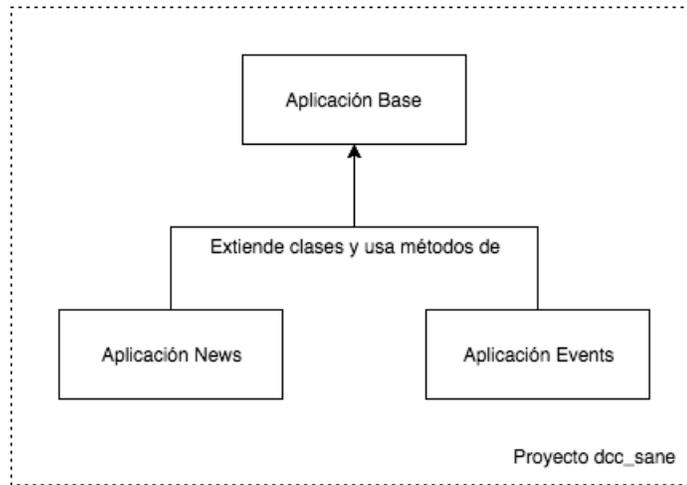


Figura 4.3: Diagrama básico de la arquitectura interna del SANE

Al igual que el SAR v2.1, el SANE posee una arquitectura de tres capas y su API provee los *endpoints* necesarios para solicitar la información de las noticias y eventos desde otros sistemas. En el caso de la reserva interna de espacios del DCC, el sistema realiza la solicitud al servidor del SAR mediante peticiones HTTP seguro. Como se puede ver en la figura 4.3, internamente la aplicación consiste en un proyecto de Django `dcc_sane` que contiene las siguientes tres aplicaciones:

- **Aplicación Base:** Ésta contiene los modelos y la implementación base de las vistas HTML y los *endpoints* de la API. Además, contiene los *templates* HTML base de todas las vistas del sistema, como es el caso de la barra superior de navegación.
- **Aplicación News:** Ésta provee las vistas HTML y los *endpoints* de la API para el manejo de noticias.
- **Aplicación Events:** De manera similar a la anterior, esta aplicación proporciona las vistas HTML y los *endpoints* de la API, pero para el manejo de eventos.

La estructura de las noticias y eventos se almacena en la base de datos en formato JSON. La ventaja de esto es que puede ser procesada fácilmente por aplicaciones que requieran mostrar el contenido. En el caso del SANE, esto se realiza mediante directivas del framework AngularJS, las cuales encapsulan el contenido HTML y el código JavaScript necesario para desplegar la información.

### 4.4. Modelo de datos

En esta sección se muestra el modelo de datos del SANE, según la estructura modelo-vista-controlador implementada por el framework Django. La entidad *News* se encarga de almacenar los datos de las noticias, y posee una imagen (*Image*) opcional para mostrar en la lista de noticias, junto con múltiples tags (*Tag*) para clasificar el contenido de ésta. De la

misma manera los eventos también tienen una imagen opcional y múltiples tags, pero además contienen un organizador (*Organizer*), una reserva de lugar (*PlaceReservation*) y múltiples registros de usuarios a eventos del sitio (*EventUserRegistration*).

Cabe destacar que la entidad *Tag* posee un atributo *hidden* que permite asignar tags ocultos a los contenidos. Estos sirven para la clasificación de la información por parte de aplicaciones externas, tales como el SINE (Sistema de notificaciones de Noticias y Eventos) y el sitio Web del DCC.

La entidad *PdfDocument* se encarga de guardar los documentos en formato PDF. Ésta no tiene relación con otras entidades, dado que se incorpora dinámicamente a la estructura JSON de las noticias y eventos. Los usuarios del sistema están vinculados a la entidad *UserInformation* para guardar los datos adicionales del usuario, sin modificar la clase de autenticación provista por Django. Por su parte, la entidad *Auth-ActivationToken* almacena el token de autenticación para validar la creación de cuenta de un usuario. Las entidades *User*, *Group*, *Permission* y *ContentType* son provistas por el *framework* para realizar la gestión de usuarios, grupos y permisos en el sistema. En la figura 4.4 se muestra las relaciones entre las entidades. El detalle de cada entidad se puede apreciar en la figura B.2, en el apéndice B.

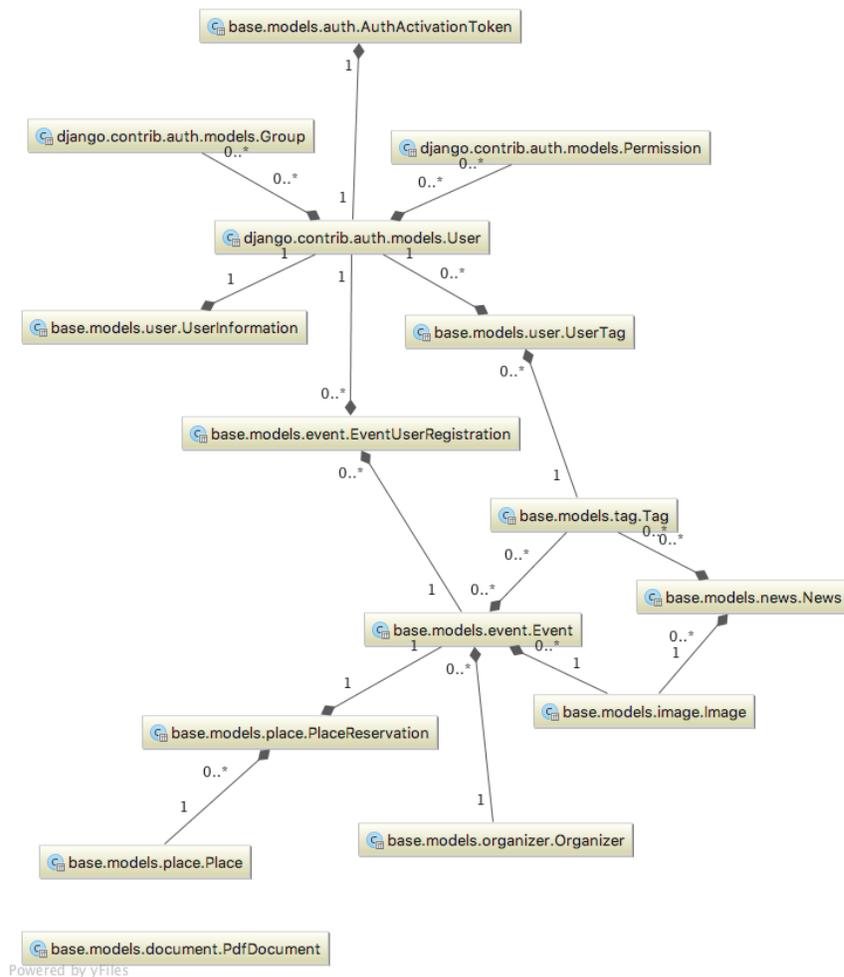


Figura 4.4: Modelo Entidad-Relación simplificado del SANE

## 4.5. Usuarios de la solución

A continuación se describen los diferentes roles de usuario del SANE.

### Usuario anónimo

Puede acceder a las noticias y eventos en el sistema. También puede compartir la información del sitio en las redes sociales Twitter, Facebook, Google+ y LinkedIn. Además tiene la opción de registrarse en la aplicación para recibir información personalizada.

### Usuario registrado

Puede realizar todas las acciones de un usuario anónimo y adicionalmente tiene la opción de indicar su asistencia a eventos y recibir notificaciones sobre nuevos contenidos de interés configurables en su perfil.

### Administrador de contenidos

Puede crear, modificar y eliminar los eventos del sitio y noticias e información asociada a estos, tales como imágenes, documentos, lugares y organizadores. Además puede administrar la información de los perfiles de usuarios del sistema.

### Administrador de sistemas

Tiene permisos para crear, modificar y eliminar toda la información del sitio. También puede acceder al panel de administración interna de la aplicación.

## 4.6. Principales vistas del sistema

A continuación se muestran las principales interfaces del SANE. En la sección A.1 del apéndice A se encuentran figuras adicionales correspondientes a las interfaces secundarias del sistema.

### 4.6.1. Home del sitio

En la figura 4.5 se aprecia la interfaz del *home* de la aplicación. Esta es la vista principal del sitio y está disponible para todos los usuarios de la aplicación.



## Noticias

02 de Mayo de 2017

### Estudiante de Doctorado asistirá a la entrega de los Premios Turing

Francisco Gutiérrez fue seleccionado como uno de los 100 estudiantes a nivel mundial que asistirán a la entrega del ACM A. M. Turing Award, considerado el Nobel de Computación.

13 de Abril de 2017

### Nueva Revista Bits

En su edición número 14 ofrece como tema principal "En las redes de Chile".

11 de Abril de 2017

### Facebook visitó el DCC para reclutar estudiantes

Con gran asistencia de estudiantes de toda la Facultad se llevó a cabo el workshop organizado por Facebook y el Departamento de Ciencias de la Computación (DCC).

## Eventos

16 de Junio de 2017 - 12:00

### Charla: "Poppy, an open source robotics platform"

20 de Abril de 2017 - 19:00

### DCC te invita a Wine and Cheese

El centro de la web semántica realizará este evento el 20 de abril.

20 de Abril de 2017 - 18:00

### Encuentro de Innovación y Emprendimiento

Evento a realizarse este 20 de abril en la Sala B04

Figura 4.5: Interfaz de usuario mostrada en el *home* del SANE

Las componentes interactivas que se muestran de la figura 4.5 son las siguientes:

- La barra de navegación superior posee vínculos a las vistas de noticias, eventos, registro de usuarios, login y búsqueda de información.
- El "carrusel" de novedades permite visualizar las noticias y eventos más importantes del Departamento. El contenido puede ser modificado por los administradores de contenidos.
- Los vínculos a noticias y eventos anteriores permiten acceder al detalle del contenido por demanda.

### 4.6.2. Lista de eventos

En la figura 4.6 se muestra la interfaz de la lista de eventos disponibles en el sistema. Esta vista puede ser accedida por todos los usuarios de la aplicación.

## Próximos eventos



### Tech Sessions por Arkano Software

28 de Junio de 2017 - 12:00

[Registrar asistencia](#)

## Eventos anteriores

Nombre	Fecha
<a href="#">Charla: "Poppy, an open source robotics platform"</a>	16 de Junio de 2017
<a href="#">Coloquio Cloud Computing y Derecho</a>	12 de Junio de 2017
<a href="#">DCC te invita a Wine and Cheese</a>	20 de Abril de 2017
<a href="#">Encuentro de Innovación y Emprendimiento</a>	20 de Abril de 2017

© 2017 DCC - Universidad de Chile

Figura 4.6: Lista de eventos registrados

Las componentes interactivas que se aprecian en la figura 4.6 son las siguientes:

- Al igual que en el caso anterior, la barra de navegación superior posee vínculos a las vistas de noticias, eventos, registro de usuarios, *login* y búsqueda de información.
- La lista de próximos eventos permite visualizar la información relevante de cada uno de estos. En particular sus afiches, encabezados, fechas y botones de registro de asistencia (los últimos disponibles solo para usuarios registrados).
- Los vínculos a eventos anteriores permiten acceder al detalle de los eventos ya ocurridos.

### 4.6.3. Detalles de un evento

En la figura 4.7 se observa la interfaz de usuario que muestra los detalles de los eventos disponibles en la aplicación. Esta vista puede ser accedida por todos los usuarios del sitio.

Comunicaciones DCC Noticias Eventos Registro Ingresar

---

## Coloquio Cloud Computing y Derecho

La legalidad de la gestión de información en la nube



Afiche del evento



**Lugar**  
Sala Matilde Brandau  
Facultad de Derecho  
Universidad de Chile

**Dirección**  
Av. Santa María 076, 6to piso  
[Ver mapa](#)

**Fecha del evento**  
12 de Junio de 2017  
10:00 - 12:30

**Organizador**  
Comunicaciones DCC  
[comunicaciones@dcc.uchile.cl](mailto:comunicaciones@dcc.uchile.cl)  
229780657

[Evento anterior](#)

Figura 4.7: Detalles de un evento registrado en el SANE

A continuación se explican brevemente las componentes interactivas que están presentes en la figura 4.7:

- Nuevamente, la barra de navegación superior se mantiene como en los casos anteriores.
- La interfaz muestra múltiples botones para compartir el evento a través de diferentes redes sociales.
- También se muestra el vínculo con mapa emergente indicando el lugar del evento.
- El botón de registro de asistencia al evento permite a los usuarios registrados indicar si asistirán o no al evento.

## 4.6.4. Lista de noticias

En la figura 4.8 se aprecia la interfaz de usuario correspondiente a la lista de noticias disponibles en el sistema. Esta vista puede ser accedida por todos los usuarios de la aplicación.

Comunicaciones DCC

Noticias Eventos Registro Ingresar

---

2017



09 de Junio de 2017

### Programa de Educación Continua del DCC graduó 50 profesionales

Los profesionales cursaron los Diplomas de Postítulo en Gestión de Proyectos Informáticos y Seguridad Computacional



09 de Junio de 2017

### El DCC conmemoró sus 40 años de vida

En una emotiva ceremonia que congregó a egresados, estudiantes, académicos y funcionarios, el Departamento de Ciencias de la Computación (DCC) de la FCFM celebró cuarenta años de historia.



Académicas del DCC organizan primera conferencia latinoamericana de mujeres en Computación

09 de Junio de 2017

### Académicas del DCC organizan primera conferencia latinoamericana de mujeres en Computación

LATiNiTY brindará un espacio para discutir temas de género, crear redes de contacto y fortalecer la comunidad regional de mujeres en Computación.



25 de Mayo de 2017

### DCC tiene un nuevo Doctor en Ciencias mención Computación

Con una tesis que se enmarca en apoyar el diseño de sistemas de computación social, Francisco Gutiérrez obtuvo el grado de Doctor



03 de Mayo de 2017

### NIC Chile celebró el medio millón de dominios .CL



03 de Mayo de 2017

### Alumna de Doctorado del DCC triunfa en ACM Student Research Competition

Maira Marzales obtuvo el primer lugar en la

Figura 4.8: Lista de noticias registradas

Las componentes interactivas que están presentes en la figura 4.8 son las que se explican a continuación:

- La barra de navegación superior se mantiene con el mismo formato y funcionalidad que en los casos anteriores.
- Los vínculos de las noticias del sistema permiten acceder a la vista del detalle de éstas.

## 4.6.5. Detalles de una noticia

En la figura 4.9 se muestra la interfaz de usuario con el detalle de una noticia disponible en el sistema. Esta vista puede ser accedida por todos los usuarios del sitio.

Comunicaciones DCC [Noticias](#) [Eventos](#) [Registro](#) [Ingresar](#)

---





Priscilla Rossi, encargada de Reclutamiento Universitario para América Latina.



Reclutadores en Workshop.



Foto de termino del evento.

### Facebook visitó el DCC para reclutar estudiantes

Con gran asistencia de estudiantes de toda la Facultad se llevó a cabo el workshop organizado por Facebook y el Departamento de Ciencias de la Computación (DCC).

**Tags** 2017 DCC Estudiante Noticias Online

El pasado 6 de abril y por tercer año consecutivo un equipo de reclutamiento de Facebook visitó el **Departamento de Ciencias de la Computación (DCC)**, con el objetivo de resolver dudas y entregar consejos a estudiantes que quieran postular a prácticas y trabajos que ofrece cada año la compañía.

En la actividad realizada en el Auditorio Enrique d'Etigny, de la Facultad de Ciencias Físicas y Matemáticas de la Universidad de Chile, además de orientar y aconsejar a los estudiantes, el equipo de reclutamiento, comandado por Priscilla Rossi, encargada de Reclutamiento universitario para América Latina, resolvió distintas dudas que presentaron los asistentes, con el fin de ayudarlos en las próximas visitas o en las postulaciones que se realizan durante el año. "Para nosotros es fundamental encontrar y contratar gente de toda Latinoamérica y para ello es fundamental el contacto con una de las mejores universidades de la región", señaló Priscilla Rossi.

Priscilla Rossi, en el comienzo del workshop destacó que trabajar en "Facebook es una experiencia dinámica e intensa y la misión es hacer al mundo más abierto y conectado, porque creemos un el mundo conectado es mejor lugar para vivir. Las personas que se conectan se acercan a las cosas que les importan. Queremos conectar los 7 mil millones de personas en el mundo y para ello necesitamos personas brillantes, apasionadas por lo que hacen y en su mayoría que nos muestran una perspectiva diferente sobre los problemas que estamos tratando de resolver", comentó.

Por otro lado, Priscilla enfatizó que "la compañía hoy en día está generamos nuevos puestos de trabajo, comunidades y creamos oportunidades para todos. Facebook recibe antecedentes laborales durante todo el año; en el sitio se van actualizando los cargos específicos que se necesitan en cada área y en el caso de las practicas, que duran tres meses, se realizan en enero septiembre y octubre", destacó.

Finalmente, la visita de Facebook terminó con la invitación de los reclutadores a los estudiantes para que postulen en las diversas ofertas de trabajo que ofrece la red social, ya sea pasantías, prácticas, tutorías y capacitaciones.

--

Comunicaciones DCC

---

© 2017 DCC - Universidad de Chile

Figura 4.9: Detalle de una noticia

A continuación se indican las componentes interactivas que están presentes de la figura 4.9:

- La barra de navegación superior se mantiene igual a las interfaces anteriores.
- El sistema muestra múltiples botones para compartir la noticia en diferentes redes sociales.

## 4.6.6. Registro de usuario

En la figura 4.10 se observa la interfaz de registro de un usuario en el sistema. Esta vista puede ser accedida sólo por los usuarios anónimos.

Comunicaciones DCC

Noticias Eventos Registro Ingresar

Datos personales

Email \*

Nombre \*

Apellido \*

Teléfono

Dirección (primera línea)

Dirección (segunda línea)

Código postal

País \*

Contraseña \*

Repetir contraseña \*

Usuario DCC

Selecciona tus temas de interés. Cuando publiquemos contenido asociado te notificaremos.

2017 ACM Bárbara Poblete DCC

Diplomas de Postítulo Enero 2017

Eric Tanter Estudiante

Gonzalo Navarro B. Inspiratéc

Jocelyn Simmonds Jorge Pérez

José A. Pino José Miguel Piquer

Jérémy Barbay Latinity Maira Marques

Mujeres en computación NIC Chile

Nancy Hirschfeld Kahler Noticias

Online PEC Patricio Poblete

Sergio Ochoa

© 2017 DCC - Universidad de Chile

Figura 4.10: Interfaz de registro de un usuario

Las componentes interactivas que están presentes en la figura 4.10 son las siguientes:

- La barra de navegación superior es igual a las interfaces anteriores.
- El formulario Web muestra múltiples campos, los cuales permiten especificar información detallada del usuario.
- El formulario también muestra múltiples botones para especificar los *tags* de interés del usuario que se está registrando. Esto se utiliza para enviarle después notificaciones automáticas según los temas de preferencia y configuración de privacidad.
- El botón “Registrar” permite crear el usuario en el sistema.

## 4.6.7. Login de usuario

En la figura 4.11 se aprecia la interfaz de autenticación de un usuario al sistema. Esta vista puede ser accedida sólo por los usuarios que no han iniciado sesión en el sitio.

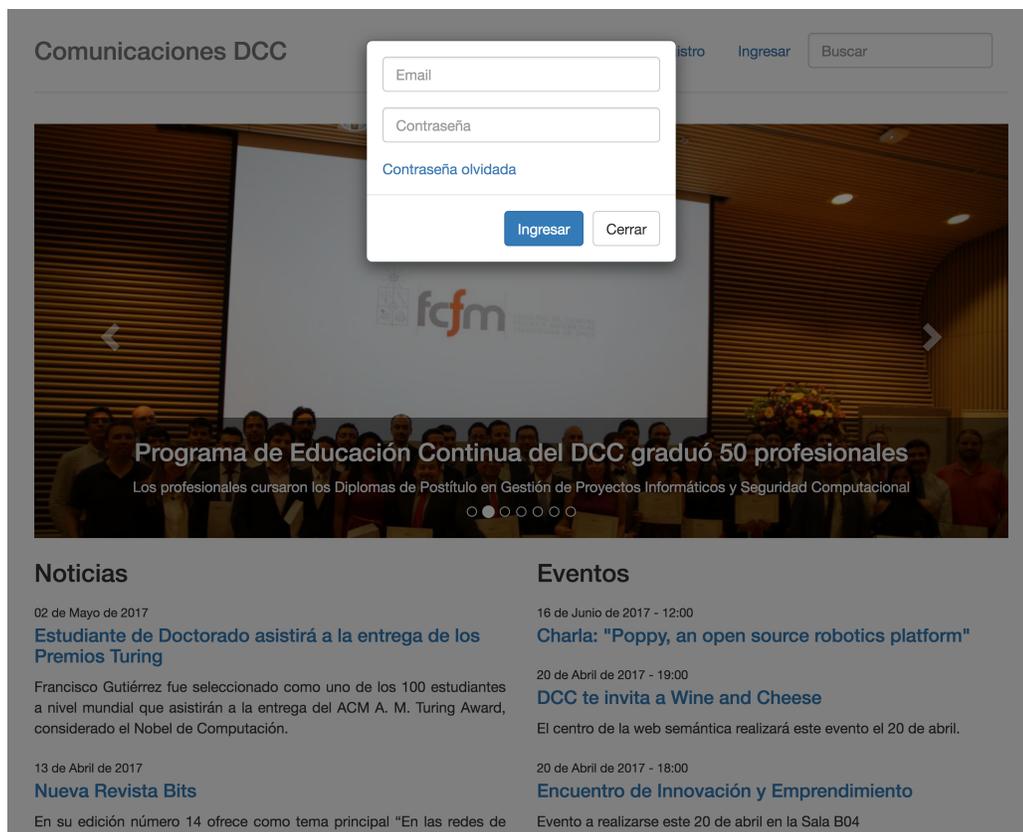


Figura 4.11: Interfaz de login de usuarios

A continuación se describen brevemente las componentes interactivas que se muestran en la figura 4.11:

- La barra de navegación superior se mantiene igual que en los casos anteriores.
- El formulario Web muestra los campos para especificar el correo electrónico (nombre de usuario) y la contraseña del usuario.
- El botón “Ingresar” le permite al usuario acceder al sistema, previa comprobación de las credenciales de conexión.
- El botón “Cerrar” cierra la ventana modal.

## 4.6.8. Interfaz de navegación

En la figura 4.12 se observa la interfaz de navegación del sistema, una vez que el usuario anónimo ha ingresado correctamente a la aplicación.

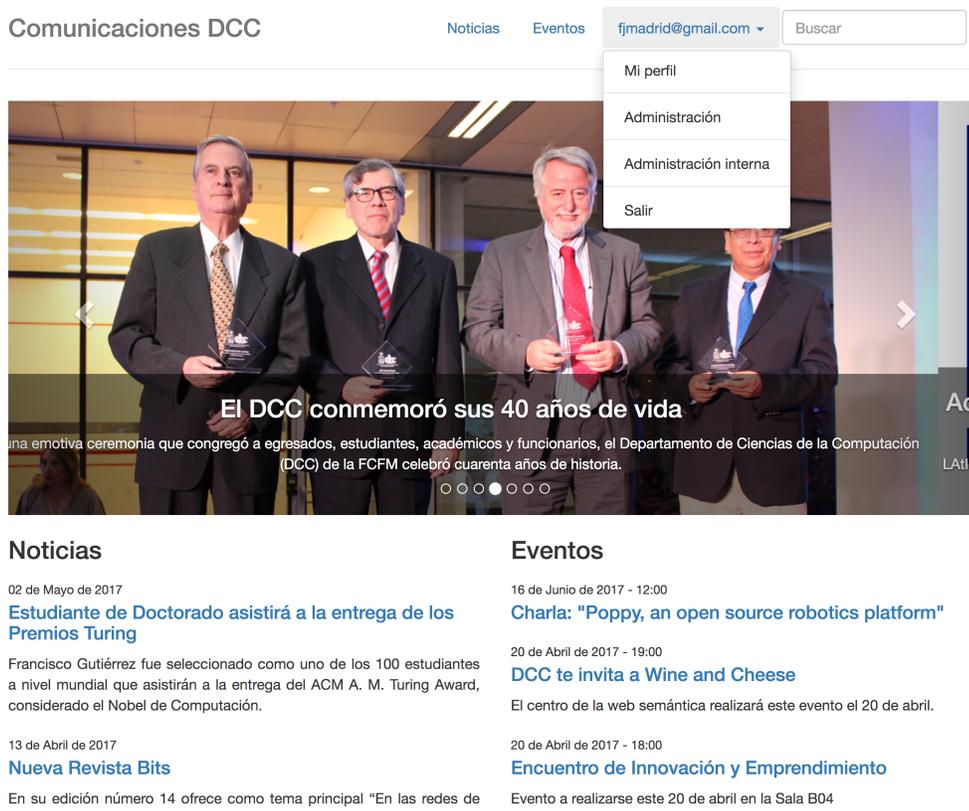


Figura 4.12: Interfaz principal de un usuario logueado

Las componentes interactivas que se muestran en la figura 4.12 son las siguientes:

- Un vínculo que permite a un usuario acceder a su perfil.
- La página también muestra un vínculo al panel de administración (cuando el perfil del usuario lo permite).
- Un vínculo al panel de administración interna de Django (cuando el perfil del usuario lo permite).
- Un vínculo para salir del sistema.

## 4.6.9. Perfil de usuario

En la figura 4.13 se observa la interfaz de usuario que muestra el perfil de un usuario registrado en la aplicación. Esta vista también puede ser accedida por los administradores de contenidos y de sistemas.

Comunicaciones DCC

Noticias Eventos fjmadrid@gmail.com Buscar

Datos personales Mis eventos

Email \* fjmadrid@gmail.com

Nombre \* Francisco

Apellido \* Madrid

Teléfono 12345678

Dirección (primera línea) Beauchef 851

Dirección (segunda línea) Oficina 222

Código postal 8370456

País \* Chile

Contraseña \* Mínimo 8 caracteres

Repetir contraseña \*

Usuario DCC fjmadrid

Selecciona tus temas de interés. Cuando publiquemos contenido asociado te notificaremos.

2017 ACM AJC Bárbara Poblete

DCC Diplomas de Postítulo Enero 2017

Eric Tanter Estudiante

Gonzalo Navarro B. Inspiratec

Jocelyn Simmonds Jorge Pérez

José A. Pino José Miguel Piquer

Jérémy Barbay Latinity Maira Marques

Mujeres en computación NIC Chile

Nancy Hirschfeld Kahler Noticias

Online PEC Patricio Poblete

Sergio Ochoa

Actualizar

© 2017 DCC - Universidad de Chile

Figura 4.13: Perfil de un usuario del SANE

A continuación se introducen brevemente las componentes interactivas mostradas en la figura 4.13:

- La barra de navegación superior es igual que en los casos anteriores.
- La pestaña “Mis Eventos” permite acceder a la información de los eventos registrados por parte del usuario.
- Una serie de campos que permiten especificar detalles propios del perfil del usuario.
- Una serie de botones que le permiten al usuario especificar los *tags* que son de su interés.
- El botón “Actualizar” le permite al usuario guardar la información actual de su perfil en el sistema.

## 4.6.10. Administración de contenidos

En la figura 4.14 se aprecia la interfaz de administración de contenidos del sistema.

Comunicaciones DCC Noticias Eventos fmadrid@gmail.com ▾ Buscar

Noticias **Eventos** Imágenes Documentos Lugares Organizadores Tags Usuarios

Buscar por nombre

ID	Fecha publicación	Nombre		
23	2017-06-09 14:43	Programa de Educación Continua del DCC graduó 50 profesionales	<input type="button" value="✎"/>	<input type="button" value="🗑"/>
22	2017-06-09 14:27	El DCC conmemoró sus 40 años de vida	<input type="button" value="✎"/>	<input type="button" value="🗑"/>
21	2017-06-09 14:04	Académicas del DCC organizan primera conferencia latinoamericana de mujeres en Computación	<input type="button" value="✎"/>	<input type="button" value="🗑"/>
18	2017-05-25 12:33	DCC tiene un nuevo Doctor en Ciencias mención Computación	<input type="button" value="✎"/>	<input type="button" value="🗑"/>
17	2017-05-03 10:32	NIC Chile celebró el medio millón de dominios .CL	<input type="button" value="✎"/>	<input type="button" value="🗑"/>
16	2017-05-03 09:55	Alumna de Doctorado del DCC triunfa en ACM Student Research Competition	<input type="button" value="✎"/>	<input type="button" value="🗑"/>
15	2017-05-02 12:43	Estudiante de Doctorado asistirá a la entrega de los Premios Turing	<input type="button" value="✎"/>	<input type="button" value="🗑"/>
13	2017-04-13 15:42	Nueva Revista Bits	<input type="button" value="✎"/>	<input type="button" value="🗑"/>
10	2017-04-11 09:37	Facebook visitó el DCC para reclutar estudiantes	<input type="button" value="✎"/>	<input type="button" value="🗑"/>
9	2017-03-08 00:00	CaDCC realizó cambio de mando 2017	<input type="button" value="✎"/>	<input type="button" value="🗑"/>

Anterior **1** 2 Siguiete

© 2017 DCC - Universidad de Chile

Figura 4.14: Interfaz de administración del SANE

Las componentes interactivas que se muestran en la figura 4.14 son las siguientes:

- La barra de navegación superior posee vínculos a las vistas antes indicados.
- La interfaz de usuario muestra diversas pestañas con subpaneles de administración de recursos.
- También se muestran los botones con opciones de creación, edición y eliminación de recursos.
- Por su parte el campo para “buscar un recurso por nombre” permite realizar esa tarea ingresando directamente el recurso a buscar.
- También se implementaron los vínculos de acceso directo a recursos.
- Se incluyeron botones para la paginación de los recursos, en caso de que estos sean demasiado. Así se evita el uso del *scroll*.

## 4.6.11. Creación de eventos y noticias

En la figura 4.15 se aprecia la interfaz de usuario que permite la creación de eventos y noticias del SANE. El contenido del evento o noticia puede ser generado a través de widgets *drag and drop*. Además existen contenedores de widgets que dividen el espacio disponible de acuerdo a una proporción en particular, de esa manera se puede generar un diagrama dinámico. Si el contenido corresponde a un evento y no tiene un lugar reservado, entonces se muestra una advertencia.

Comunicaciones DCC

Noticias Eventos fjmadrid@gmail.com Buscar

Este evento no tiene un lugar reservado

Reserva lugar Publicación Guardar Panel de administración

Diagrama Vista previa

Noticia vacía Imagen No se ha seleccionado una imagen

Buscar imagenes +

Tags No se han seleccionado tags

Buscar tags

Elementos

Contenedor 1:1

Contenedor 3:1

Contenedor 1:1:1

Contenedor 1:2:1

Titulo

Bajada

Fecha publicación

Tags

Texto

Foto

Video

Hipervínculo

Documento

Figura 4.15: Interfaz de creación de eventos

A continuación se muestran las componentes interactivas que se presentan en la figura 4.15:

- Campo “Imagen” que permite seleccionar la imagen principal del evento o noticia.
- Campo “Tags” que permite seleccionar los tags asociados al evento o noticia.
- Los elementos contenedores (blancos) y los widgets (azules) son arrastrables, y por lo tanto se los puede colocar en el área “Elementos”.
- El área roja con un ícono de papelerero permite eliminar widgets en el área “Elementos”.
- El botón “Publicación” permite configurar diversos detalles, tales como la fecha, visibilidad, presencia en el carrusel del *home*, entre otros.
- El botón “Reserva lugar”<sup>4</sup> permite configurar la reserva de un lugar para realizar el evento en las dependencias del DCC. Si el evento se realiza fuera del DCC, por ejemplo

<sup>4</sup>Disponible en la edición de eventos

en el auditorio Enrique d’Etigny, entonces el sistema permite ingresar un campo de texto con la ubicación del lugar.

- La pestaña ‘Vista previa’ permite visualizar los cambios del contenido sin necesidad de guardar.
- El botón “Guardar” almacena la información en el sistema.

#### 4.6.12. Administración interna

En la figura 4.16 se aprecia la interfaz de administración interna de la aplicación.

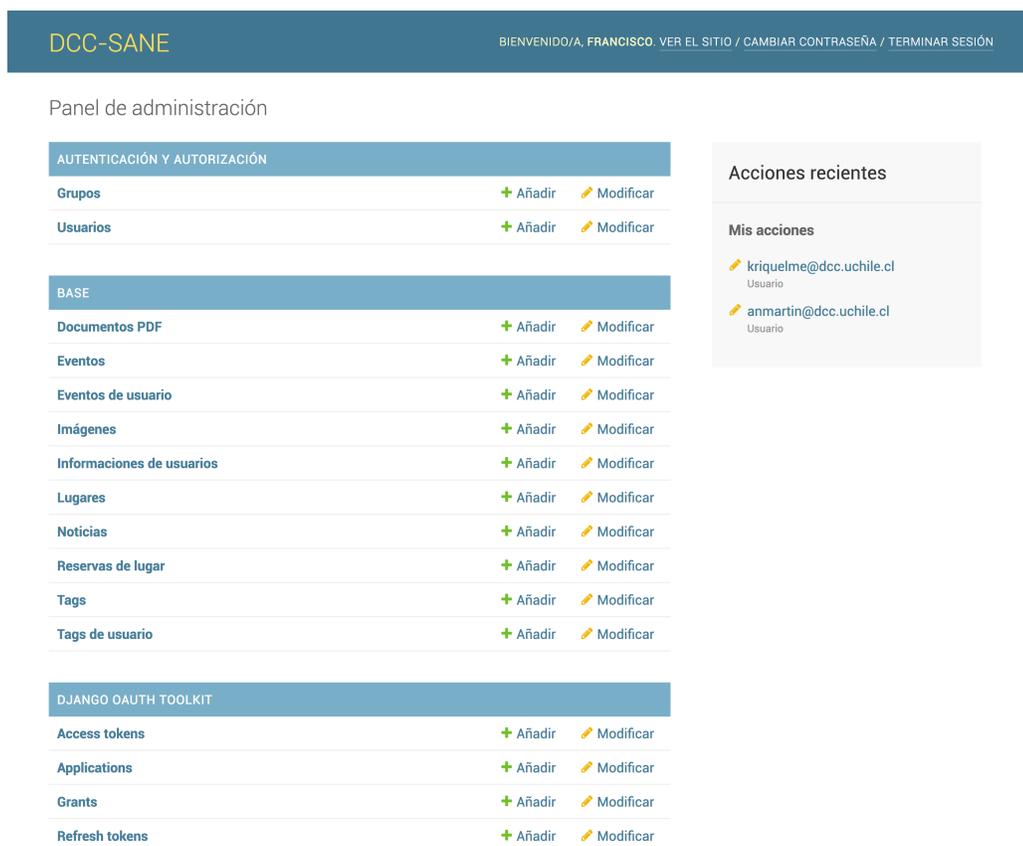


Figura 4.16: Interfaz de administración interna del SANE

Esta vista permite administrar todos los registros del sistema mediante el panel incorporado en el framework Django. Este formulario Web funciona de la misma manera que el ya explicado para el sistema SAR.

# Capítulo 5

## Evaluación de las soluciones

En este capítulo se muestran en detalle las evaluaciones realizadas a los sistemas SAR y SANE, tanto a nivel técnico como de usuarios.

### 5.1. Evaluación técnica

Las modificaciones realizadas al SAR cumplen con la especificación de la arquitectura de microservicios propuesta. Además, los tests unitarios y de integración añadidos garantizan que los servicios implementados cumplen el comportamiento esperado para las nuevas funcionalidades.

Por su parte el desarrollo del SANE cumple con la especificación propuesta, y con todos los estándares actuales, tanto en metodologías ágiles como en métricas de calidad de software. En la tabla 5.1 se muestra el test coverage del *front-end* y del *back-end* de la aplicación. Por otro lado, en la tabla 5.2 se aprecia la complejidad ciclomática [16] [9] y el índice de mantenibilidad de la aplicación [6] [21].

La primera métrica se calcula utilizando el grafo de control de flujo del código, teniendo en cuenta sus vértices, aristas y componentes conexas. La segunda métrica se obtiene usando las métricas de complejidad de Halstead [22], la complejidad ciclomática, el número de líneas y el porcentaje de comentarios del código.

Se espera que el software posea al menos 80 % de cobertura de tests, tanto de *statements* como de *branches*. En cambio para la complejidad ciclomática se espera un valor entre 1 y 5 (ambos inclusive), mientras que para el índice de mantenibilidad el valor debe ser mayor a 20.

Las métricas de la aplicación son obtenidas con las librerías Radon<sup>1</sup> de Python y Plato<sup>2</sup> de JavaScript.

---

<sup>1</sup>Radon <http://radon.readthedocs.io/>

<sup>2</sup>Plato <https://github.com/es-analysis/plato>

	Statements	% de statements	Branches	% de branches	Cantidad de tests
<b>Front-end</b>	1129/1168	96,66	252/280	90,00	258
<b>Back-end</b>	1821/1821	100,00	421/421	100,00	359

Tabla 5.1: Tests coverage del front-end y back-end junto con cantidad de tests del SANE

	Complejidad ciclomática	Índice de mantenibilidad
<b>Front-end</b>	1,19	76,81
<b>Back-end</b>	2,68	89,47

Tabla 5.2: Complejidad ciclomática promedio e índice de mantenibilidad promedio del SANE

Los tests de *front-end* son ejecutados en los browsers Google Chrome versión 59, Mozilla Firefox versión 54 y Microsoft Edge<sup>3</sup> versión 40; usando los sistemas operativos Apple macOS Sierra y Microsoft Windows 10.

Tanto el SAR como el SANE poseen un sistema de *deployment* automatizado, el cual usa una implementación de scripts basada en la librería Fabric<sup>4</sup> de Python. Esta capacidad facilita el paso a producción de las nuevas versiones de estos sistemas, y además permite realizar *rollback* a una versión anterior en caso de ser necesario. Estos scripts también permiten la implementación de *Continuous Deployment* [17] de los proyectos.

El sistema SANE cuenta con 27 tests end-to-end (E2E) que comprueban el funcionamiento de toda la aplicación (*front-end* y *back-end*) utilizando la librería Selenium<sup>5</sup> en conjunto con *bindings* de Python. Sólo se verifican las funcionalidades críticas de interacción de usuarios, tales como la creación de una nueva cuenta, el login y el registro a un evento siguiendo recomendaciones de la industria [20].

## 5.2. Evaluación de usuarios

El día miércoles 1 de marzo de 2017 se pasó a producción<sup>6</sup> la versión 2.1 del SAR, manteniendo las reservas de salas ya incluidas en la versión SAR 1.0. Posteriormente el día miércoles 8 de marzo de 2017 se realizó una capacitación para los académicos y funcionarios sobre la nueva aplicación, y se recogieron sugerencias para mejorar las interfaces de usuario. Desde ese día sólo se han solucionado pequeños bugs de uso, y además se han introducido cambios mínimos a las vistas. Los principales usuarios del servicio acogieron de buena forma la solución, dado que las interfaces propuestas en la nueva versión son una actualización de las implementadas en el SAR v1.0, lo cual disminuyó el tiempo de aprendizaje de la nueva plataforma.

El día viernes 2 de junio de 2017 se realizó el traslado del SANE desde el servidor de stage, al servidor de producción<sup>7</sup> que se encuentra en dependencias del Departamento. Posterior-

<sup>3</sup>Sólo disponible en Microsoft Windows 10

<sup>4</sup>Fabric <http://www.fabfile.org/>

<sup>5</sup>Selenium <http://www.seleniumhq.org/>

<sup>6</sup>SAR <https://sar.dcc.uchile.cl/>

<sup>7</sup>SANE <https://comunicaciones.dcc.uchile.cl/>

2017



19 de Julio de 2017

### Estudiantes de la FCFM finalizan exitosamente Taller de Proyecto enfocado en robótica

Duckietown Engineering Chile, es el Taller de Proyecto que estudiantes la FCFM dictaron para alumnos de Plan Común este semestre.



(a) Lista de noticias del SANE



## Estudiantes de la FCFM finalizan exitosamente Taller de Proyecto enfocado en robótica

Duckietown Engineering Chile, es el Taller de Proyecto que estudiantes la FCFM dictaron para alumnos de Plan Común este semestre.

A través del programa “Estímulo para proyectos académicos de estudiantes de postgrado de la universidad de Chile”, estudiantes del Departamento de Ciencias de la Computación (DCC), de Ingeniería Eléctrica (DIE) y de Ingeniería Mecánica (DIMEC),

(b) Detalles de una noticia del SANE

Figura 5.1: Vistas de noticias del SANE renderizadas en un dispositivo Apple iPhone 6

mente, el personal de la oficina de Comunicaciones del DCC comenzó a completar la nueva plataforma con contenidos similares a los del sitio Web del Departamento. Los funcionarios se encuentran satisfechos con la solución entregada, ya que posee una buena usabilidad, no requiere completar información redundante y ambigua, y además disminuye el tiempo de creación de contenidos en un 10 % aproximadamente (medido a través de pruebas en la página Web oficial del DCC, y en el SANE). Además, el tiempo de edición posterior a la publicación de noticias y eventos es mucho menor, ya que los cambios efectuados se despliegan automáticamente, en comparación a la página Web del Departamento. Otro punto a destacar, como se puede ver en las figuras 5.1a y 5.1b, es la posibilidad de desplegar correctamente la información en computadores personales, tablets, celulares, y en un futuro próximo, en las pantallas públicas del DCC. Esto último dará fin a la actual dependencia que estas pantallas tienen respecto del sistema TV Filler, que es provisto por una empresa externa al Departamento.

# Capítulo 6

## Conclusiones y trabajo a futuro

Este trabajo de memoria buscó integrar dos sistemas de apoyo a la operación del DCC. A uno de ellos se le hizo reingeniería (SAR) y el otro se desarrolló desde cero (SANE). Más allá de las ventajas propias de contar con nueva y mejor funcionalidad para apoyar las actividades del Departamento, se buscó determinar un camino que permitiera integrar de forma razonable los diversos sistemas del DCC (es decir, integrarlos sin necesidad de hacer grandes cambios al código fuente de estos), ya sean estos nuevos o legados. Dicha integración debería evitar que los sistemas sigan funcionando como islas, es decir, como compartimentos que mantienen localmente datos y servicios replicados sin ninguna coordinación (o sincronización) entre ellos.

Buscando alcanzar tanto el desarrollo de la nueva funcionalidad de los sistemas SAR y SANE, como la integración de los mismos, se realizó este trabajo de memoria el cual representa un proyecto piloto que ayuda al DCC a decidir el camino a seguir para integrar el resto de sus sistemas. La integración realizada en este trabajo ha sido a nivel de datos y servicios, permitiendo el reuso de estos últimos, así como mantener la coherencia y consistencia de la información compartida por estos sistemas. Esta integración también permite, por ejemplo, mantener una única fuente de verdad para la información, dependiendo del ámbito a la cual ésta pertenece. Desde el punto de vista de la integración planteada inicialmente (al comienzo de esta memoria), el objetivo fue alcanzado con éxito a través del uso de una arquitectura de microservicios, la cual permite la integración de sistemas a través de una API que debe ser agregada a cada sistema a integrar.

Respecto a los sistemas integrados, el SAR y el SANE, ambos representan un avance para el DCC. La reingeniería del SAR permitió que el sistema (además de volverse integrable con otros sistemas) se volviera más mantenible y evolucionable de cara al futuro. Los servicios agregados a este sistema (por ejemplo, la administración de puestos de trabajo de alumnos de postgrado, postdoctorandos y profesores visitantes) agilizó varias de las tareas de los funcionarios del DCC, y permitió además llevar un control del uso del espacio físico, cosa que en el pasado era casi imposible de realizar.

Respecto al sistema SANE, se logró centralizar toda la información de noticias y eventos del DCC, a través de un sistema que permite administrar de forma fácil este tipo de información por parte de la oficina de comunicaciones. Ahora se cuenta con una única fuente de verdad

para dicha información, la cual además permite que otros sistemas se conecten a él para consumir la información de manera segura y automática. El hecho de contar con este sistema permitirá que una buena parte de las notificaciones que realiza diariamente la oficina de Comunicaciones se realice en forma automática. Esto no sólo bajará la carga sobre dicha oficina, sino que además el proceso de difusión ya no dependerá de la disponibilidad o de la memoria (para enviar los recordatorios) de los funcionarios encargados.

Como se mostró en el capítulo 5, ambos sistemas fueron evaluados a nivel técnico y a nivel de usuario. Los resultados obtenidos fueron altamente satisfactorios, por lo que ambos sistemas se encuentran actualmente en producción. En el caso del SANE, sus ventajas aún no son muy visibles para el público en general, puesto que las herramientas que consumen información de él para luego generar notificaciones aún están en fase de desarrollo; por ejemplo, el alimentador de la página Web, el generador de noticias para las pantallas públicas y el notificador de usuarios suscritos (principalmente por email).

La realización de este trabajo de memoria ha sido tremendamente enriquecedora desde el punto de vista ingenieril. Entre las cosas aprendidas en este proceso está el hecho de que el uso de herramientas y metodologías apropiadas para un desarrollo de software particular, tiene estrecha relación con el resultado del producto obtenido. Por ejemplo, en el caso de esta memoria el uso de metodologías ágiles permitió crear rápidamente una solución funcional, comprendiendo de mejor manera las necesidades de los usuarios, y evitando esfuerzos innecesarios de desarrollo debido a que el software fue creado a la medida para los clientes. También se pudo constatar que el uso de pruebas automatizadas es fundamental en la creación de aplicaciones; particularmente para garantizar su calidad, mantenibilidad y robustez.

Como trabajo a futuro se contempla mejorar la estética del SAR y SANE, en conjunto con un diseñador gráfico. Además, el autor de esta memoria ya está trabajando en el desarrollo de un sistema que se conecte al SANE, y genere automáticamente los avisos correspondientes en las pantallas públicas del DCC. Esto evitará la duplicación de esfuerzos de publicación por parte de los funcionarios de la oficina de Comunicaciones, y además permitirá eliminar la actual dependencia que el Departamento tiene respecto de la empresa externa que provee el servicio del TV Filler (software propietario actualmente usado en el DCC para tal fin). También se debe comprobar la correcta interacción con el Sistema de Información de Noticias y Eventos (SINE), el cual se encuentra actualmente en desarrollo, y que se encarga de realizar el envío automático de notificaciones a usuarios a través de diversos canales de comunicación. Otro punto importante a implementar como trabajo a futuro en esta iniciativa, es el Continuous Deployment de los proyectos, como una forma de disminuir el tiempo de entrega de las nuevas versiones de estos.

# Bibliografía

- [1] Austin, Daniel y Barbir, Abbie y Ferris Christopher y Garg Sharad: *Web services architecture requirements*. W3C Working Group Note, 2004.
- [2] Beck, Kent: *Extreme programming explained: embrace change*. Addison-Wesley Professional, 2000.
- [3] Beck, Kent: *Test-driven development: by example*. Addison-Wesley Professional, 2003.
- [4] Booth, David y Haas, Hugo y McCabe Francis y Newcomer Eric y Champion Michael y Ferris Chris y Orchard David: *Web services architecture*. W3C Working Group Note, 2004.
- [5] Brown, Alan y Johnston, Simon y Kelly Kevin: *Using service-oriented architecture y component-based development to build web service applications*. Rational Software Corporation, 6, 2002.
- [6] Coleman, Don y Ash, Dan y Lowther Bruce y Oman Paul: *Using metrics to evaluate software system maintainability*. Computer, 27(8):44–49, 1994.
- [7] Dragoni, Nicola y Giallorenzo, Saverio y Lafuente Alberto Lluch y Mazzara Manuel y Montesi Fabrizio y Mustafin Ruslan y Safina Larisa: *Microservices: yesterday, today, and tomorrow*. arXiv preprint arXiv:1606.04036, 2016.
- [8] Driessen, Vincent: *A successful Git branching model*, 2010.
- [9] Gill, Geoffrey K. y Kemerer, Chris F.: *Cyclomatic complexity density y software maintenance productivity*. IEEE transactions on software engineering, 17(12):1284–1288, 1991.
- [10] Google: *Search Engine Optimization Starter Guide*, 2010. <https://static.googleusercontent.com/media/www.google.com/en//webmasters/docs/search-engine-optimization-starter-guide.pdf>.
- [11] Haas, Hugo y Brown, Allen: *Web services glossary*. W3C Working Group Note, 2004.
- [12] Hardt, Dick: *The OAuth 2.0 authorization framework*, 2012.
- [13] He, Hao: *What Is Service-Oriented Architecture*, 2003. <http://www.xml.com/pub/a/ws/2003/09/30/soa.html>.

- [14] Jones, Michael y Bradley, John y Sakimura Nat: *JSON Web Token (JWT)*, 2015.
- [15] Llorens, M. y Madrid, F. y Merino P. y Pavez M. y Perez I. y Ormeño S. y Saez T.: *Documento histórico del proyecto Administración de Salas, Ingeniería de Software II, Universidad de Chile*, 2016.
- [16] McCabe, Thomas J: *A complexity measure*. IEEE Transactions on software Engineering, 1(4):308–320, 1976.
- [17] Olsson, Helena Holmström y Alahyari, Hiva y Bosch Jan: *Climbing the "Stairway to Heaven-A Multiple-Case Study Exploring Barriers in the Transition from Agile Development towards Continuous Deployment of Software*. En *Software Engineering y Advanced Applications (SEAA), 2012 38th EUROMICRO Conference on*, páginas 392–399. IEEE, 2012.
- [18] Rosenberg, Linda H y Hyatt, Lawrence E: *Software re-engineering*. Software Assurance Technology Center, páginas 2–3, 1996.
- [19] W3C, Oficina Española: *Guía breve de servicios web*, 2005. <http://www.w3c.es/Divulgacion/GuiasBreves/ServiciosWeb>.
- [20] Wacker, Mike: *Just Say No to More End-to-End Tests*, 2015. <https://testing.googleblog.com/2015/04/just-say-no-to-more-end-to-end-tests.html>.
- [21] Welker, Kurt D: *The software maintainability index revisited*. CrossTalk, 14:18–21, 2001.
- [22] Weyuker, Elaine J.: *Evaluating software complexity measures*. IEEE transactions on Software Engineering, 14(9):1357–1365, 1988.

# Apéndice A

## Interfaces

A continuación se muestran las interfaces secundarias del SANE, que corresponden al panel de administración y a la documentación del código de la aplicación.

### A.1. Interfaces del SANE

En la figura A.1 se aprecia la interfaz de administración de eventos del SANE. A través de ella se puede crear, editar o eliminar un evento. Además se puede generar una lista de usuarios registrados a un evento en particular. También se puede realizar una búsqueda de usuarios registrados a través del campo “Buscar por nombre”.

Comunicaciones DCC

Noticias Eventos fjmadr@gmail.com ▾ Buscar

Noticias **Eventos** Imágenes Documentos Lugares Organizadores Tags Usuarios

Buscar por nombre ✕ Nuevo evento

ID	Fecha publicación	Fecha	Hora inicio	Hora término	Nombre			
5	2017-06-22 13:13	2017-06-28	12:00:00	15:00:00	Tech Sessions por Arkano Software	☰	✎	🗑
4	2017-06-13 15:54	2017-06-16	12:00:00	15:00:00	Charla: "Poppy, an open source robotics platform"	☰	✎	🗑
3	2017-06-08 15:42	2017-06-12	10:00:00	12:30:00	Coloquio Cloud Computing y Derecho	☰	✎	🗑
1	2017-06-04 14:08	2017-04-20	19:00:00	21:00:00	DCC te invita a Wine and Cheese	☰	✎	🗑
2	2017-06-04 14:10	2017-04-20	18:00:00	21:00:00	Encuentro de Innovación y Emprendimiento	☰	✎	🗑

Anterior 1 Siguiente

© 2017 DCC - Universidad de Chile

Figura A.1: Interfaz de administración de eventos del SANE

[Noticias](#) [Eventos](#) [Imágenes](#) [Documentos](#) [Lugares](#) [Organizadores](#) [Tags](#) [Usuarios](#)

Buscar por nombre



Nueva imagen

ID	Nombre		
74	ArkanoTech-Sessions_.png		
67	educacion.png		
66	graduacionPECmarzo2017-7		
65	graduacionPECmarzo2017-6		
64	graduacionPECmarzo2017-5		
63	graduacionPECmarzo2017-4		
62	graduacionPECmarzo2017-3		
61	graduacionPECmarzo2017-2		
60	graduacionPECmarzo2017-1		
59	Aniversario DCC4		

Anterior **1** 2 3 4 5 6 7 Siguiete

Figura A.2: Interfaz de administración de imágenes del SANE

En la figura A.2 se observa la interfaz de administración de imágenes del SANE. Esta interfaz permite crear, editar o eliminar una imagen. Además se pueden buscar imágenes mediante el campo “Buscar por nombre”.

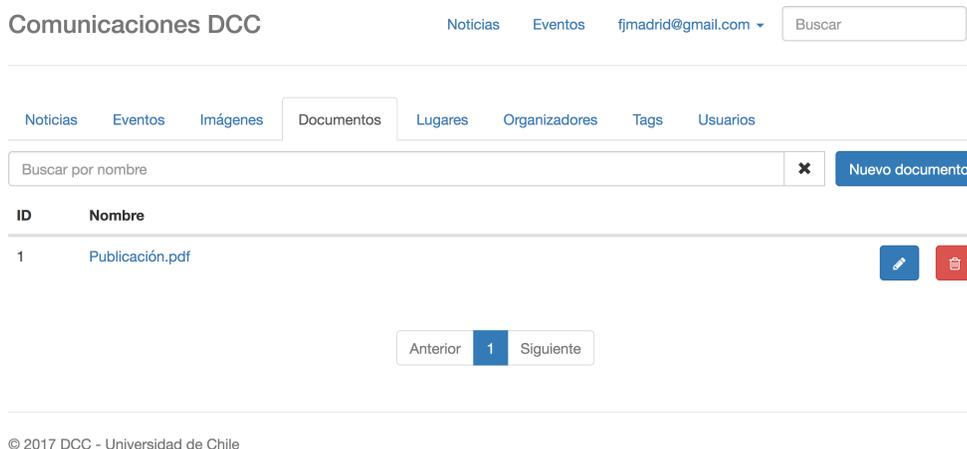


Figura A.3: Interfaz de administración de documentos del SANE

En la figura A.3 se aprecia la interfaz de administración de documentos del SANE. En ella se puede crear, editar o eliminar un documento PDF. También se pueden buscar a través del campo “Buscar por nombre”.

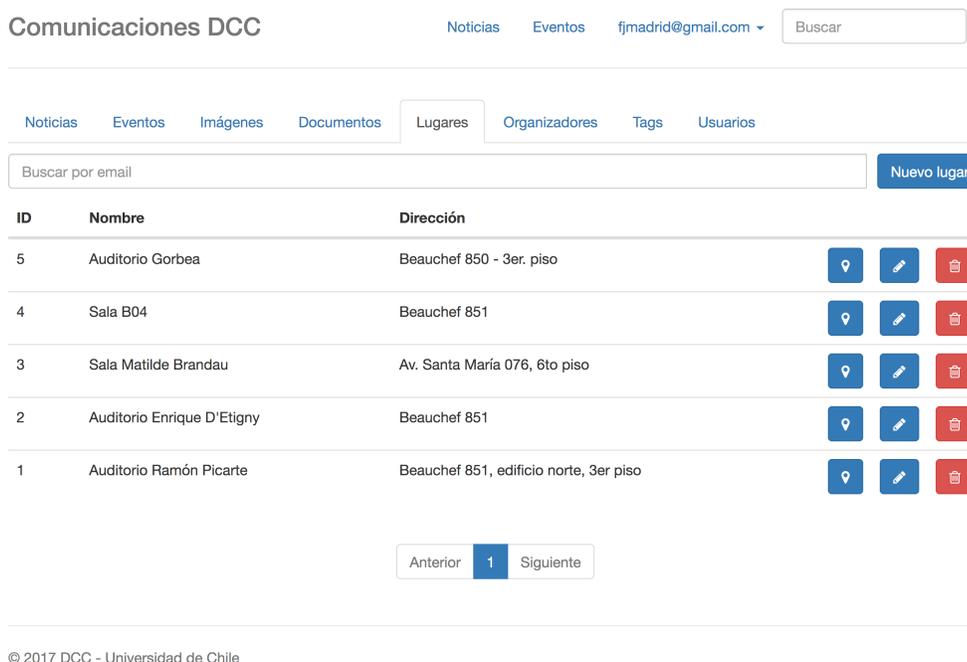


Figura A.4: Interfaz de administración de lugares del SANE

En la figura A.4 se muestra la interfaz de administración de lugares del SANE. Similar a los casos anteriores, a través de esta interfaz se puede crear, editar o eliminar el lugar del evento. Además se puede visualizar un mapa con la ubicación de dicho lugar. También se pueden buscar mediante el campo “Buscar por nombre”.

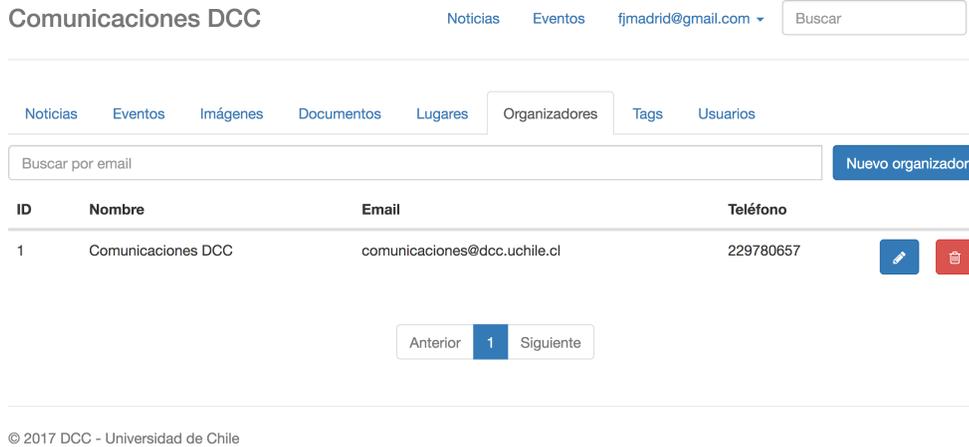


Figura A.5: Interfaz de administración de organizadores del SANE

En la figura A.5 se observa la interfaz de administración de organizadores del SANE. Usando esta vista se puede crear, editar o eliminar un organizador de un evento. Al igual que en los casos anteriores, se pueden buscar a través del campo “Buscar por nombre”.

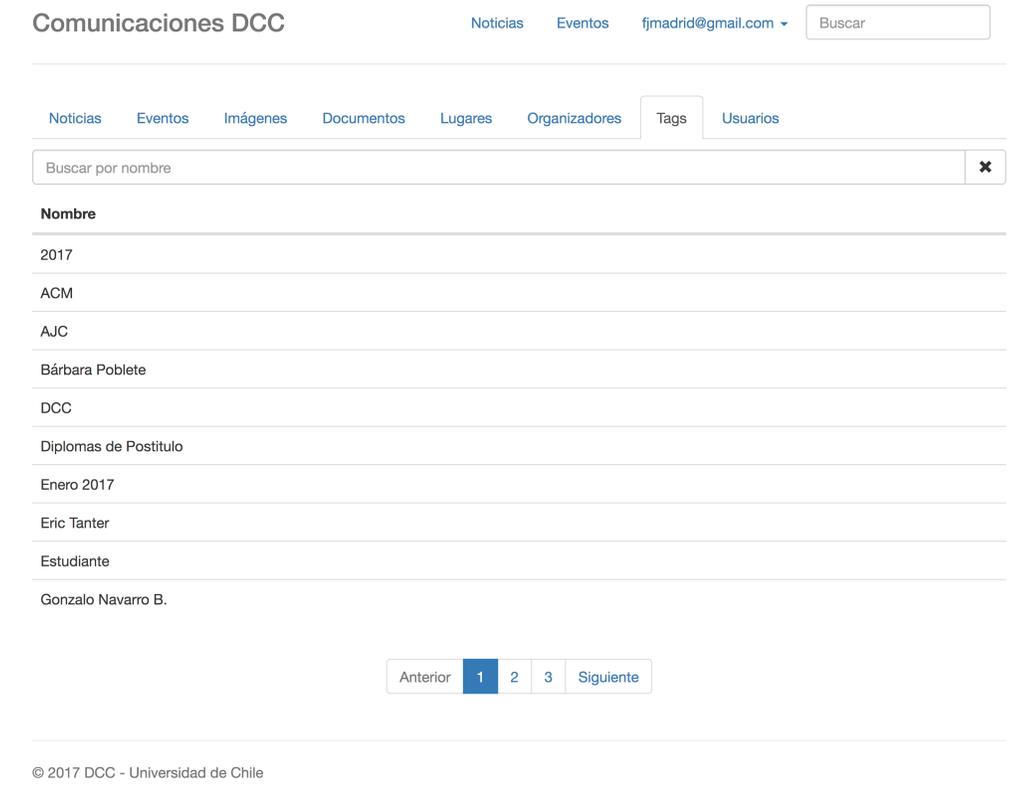


Figura A.6: Interfaz de administración de tags del SANE

En la figura A.6 se aprecia la interfaz de administración de tags del SANE. Estos tags se pueden buscar mediante el campo “Buscar por nombre”.

Buscar por email



ID	Email	Nombre	Apellido		
2	fjmadrid@gmail.com	Francisco	Madrid		
3	kriquelme@dcc.uchile.cl	Karín	Riquelme Díaz		
4	anmartin@dcc.uchile.cl	Ana	Martinez		
5	sochoa@dcc.uchile.cl	Sergio	Ochoa		
6	dperovic@dcc.uchile.cl	Daniel	Perovich		

Anterior

1

Siguiente

Figura A.7: Interfaz de administración de usuarios del SANE

En la figura A.4 se muestra la interfaz de administración de usuarios del SANE. La interfaz permite editar o eliminar un usuario. También se pueden buscar los usuarios registrados través del campo “Buscar por nombre”.

## A.2. Documentación del SANE

En la figura A.8 se observa la interfaz de la documentación del SANE, generada con la librería Sphinx de Python.

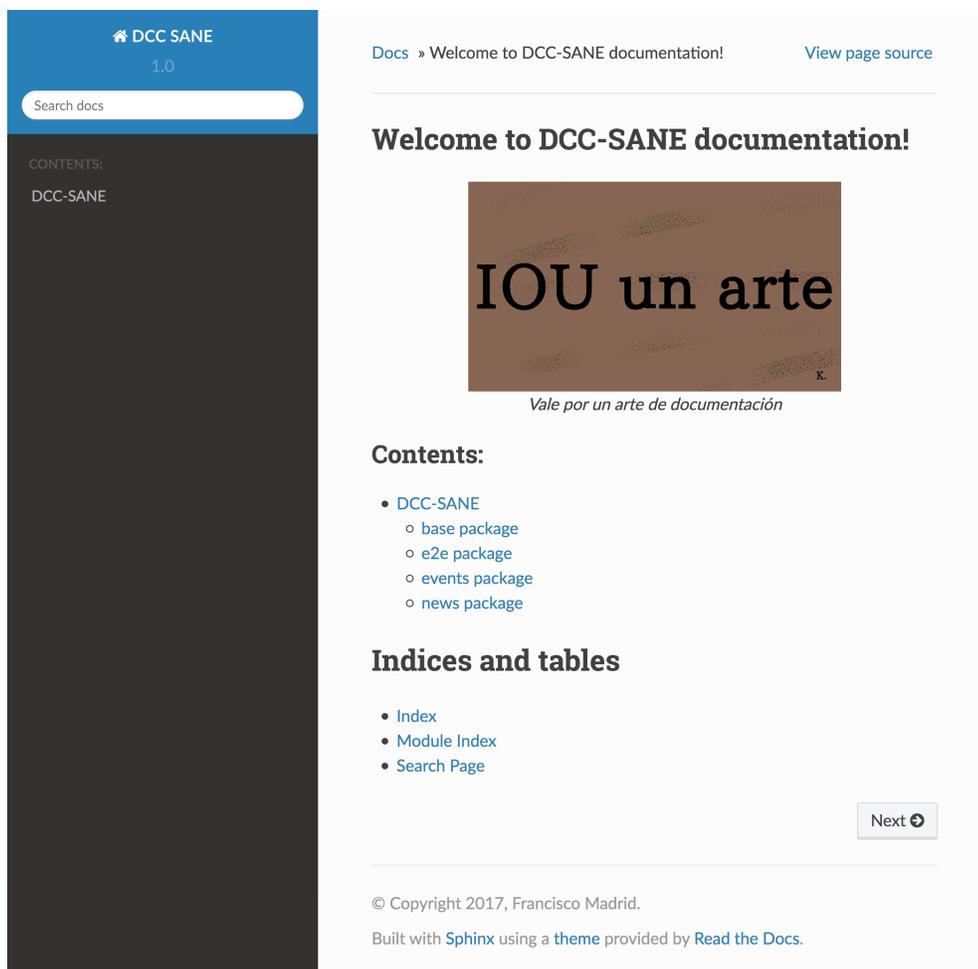


Figura A.8: Interfaz de documentación del SANE en HTML



