



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

Integración de herramientas de conformación y coevaluación de equipos de desarrollo de software

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL EN
COMPUTACIÓN

Cristóbal Ignacio Mesías Durán

Profesor Guía:

Sergio Ochoa Delorenzi

Miembros de la Comisión:

José Piquer Gardner

Éric Tanter

SANTIAGO DE CHILE

2022

Resumen

El trabajo en equipo es algo inherente en nuestra formación y carrera profesional. Para que éste sea provechoso, debe existir un mínimo de cohesión entre los integrantes. Si se quiere cuantificar esta cohesión, es necesario recibir feedback ya que éste generará confianza entre los miembros del equipo. Ahora bien, en el Departamento de Ciencias de la Computación (DCC) de la Universidad de Chile se utilizan dos herramientas, ambas desarrolladas en forma interna varios años atrás, que facilitan el proceso de trabajo en equipo; éstas son: 1) la herramienta de conformación de equipos basados en aptitudes psico-sociales, y 2) la herramienta de coevaluaciones que permite evaluar y enviar feedback entre compañeros de un mismo equipo.

Dado que ambas herramientas se encuentran separadas, pero comparten bastante cosas, se decidió integrarlas con el fin de reducir la carga que el uso de éstas les generaba a los académicos que las utilizaban. Para ello, fue necesario realizar una evaluación y análisis del estado de ambas herramientas para determinar el camino para su integración. El resultado indicó que era necesario realizar una reingeniería de las herramientas, para así robustecer su funcionamiento y facilitar integración de futuras herramientas; además, se requería integrarlas estéticamente y mejorar la usabilidad de ellas.

Por otra parte, este trabajo también abordó el agregado de nuevos criterios de conformación de equipos, tales como el género, el rendimiento académico y la existencia de conflictos entre pares. Para cumplir con los objetivos planteados se tuvieron que realizar diversos cambios a las herramientas, partiendo por migrarlas a versiones actuales de las tecnologías que usaban, pues la mayoría de ellas estaban deprecadas.

En lo que respecta la implementación de la solución propuesta, esta aplica principalmente a la herramienta de Coevaluaciones, dado que el trabajo era de alta complejidad y requería más tiempo del que estaba planteado inicialmente. Como resultado se obtuvo la integración de ambas herramientas a nivel de datos, y la reingeniería del sistema de coevaluaciones, pues éste era el más comprometido en términos de funcionalidad, seguridad y usabilidad. La usabilidad y la utilidad de la solución obtenida fue evaluada con una muestra de los usuarios finales.

Dado que el objetivo de este trabajo de título era, en un principio, la integración de ambas herramientas, se puede decir que se logró la primera etapa y se obtuvo un producto mínimo viable que puede ser usado en la práctica para mitigar el problema planteado. Como trabajo a futuro, se propone implementar la reingeniería a la herramienta de conformación de equipos, y así concluir la labor inicialmente planteada.

Tabla de Contenidos

1. Introducción	1
1.1. Problema Abordado	1
1.2. Objetivos de la Memoria	3
1.3. Resumen de la Solución	4
1.4. Estructura del Documento	4
2. Análisis de los Sistemas Legados	6
2.1 Herramienta de Conformación de Equipos	6
2.1.1 Estructura de la herramienta	6
2.1.2 Tecnologías utilizadas	8
2.1.3 Interfaces de usuario	9
2.2 Herramienta de Coevaluación de Equipos	11
2.2.1 Estructura de la herramienta	11
2.2.2 Tecnologías utilizadas	13
2.2.3 Interfaces de usuario	13
3. Rediseño de la Herramienta de Conformación de Equipos	17
3.1 Criterios para Conformación de Equipos	17
3.1.1. Aspectos de Género	17
3.1.2. Aspectos de Rendimiento Académico	18
3.1.3. Conflictos entre pares	19
3.2. Rediseño de interfaces	19
3.3. Implementación de ajustes al diseño del sistema	22
4. Rediseño del Sistema de Coevaluaciones	23
4.1. Principales requisitos de la solución	23
4.2. Implementación de ajustes al diseño del sistema	24
4.3. Rediseño de interfaces	28

5. Evaluación de la Solución	33
5.1. Evaluación de criterios para la conformación de equipos	33
5.2. Evaluación de la usabilidad de las herramientas	36
5.2.1. Definición de System Usability Scale	36
5.2.2. Resultados de la Encuesta SUS	38
6. Conclusiones y Trabajo a Futuro	39
6.1. Conclusiones sobre aspectos técnicos	39
6.2. Conclusiones Personales	40
6.3. Trabajo Futuro	41
Bibliografía	42

1. Introducción

El trabajo en equipo es una actividad fundamental en el desarrollo de software. Por lo tanto, debe existir un mínimo de cohesión entre los integrantes de los equipos, para que los objetivos planteados en un proyecto de desarrollo puedan cumplirse de buena manera. En este contexto, la cohesión tiene relación con el nivel de comunicación y coordinación de las tareas en el interior del equipo. Los equipos que mayor cohesión poseen se caracterizan por tener una mayor efectividad al momento de desarrollar un proyecto, y por consecuencia, dichos proyectos tienden a ser más exitosos.

Ahora bien, crear un equipo de trabajo donde exista una alta cohesión no es una tarea fácil, pues todas las personas poseen un perfil psico-social distinto, y no necesariamente compatible con otro, ya que a priori, los posibles integrantes no se conocen entre ellos. Si los equipos se conforman de manera correcta, la probabilidad de éxito de un proyecto mejora sustancialmente. En caso contrario, la probabilidad de fracaso será la que aumenta. Esto ocurre ya que si un equipo cuenta con miembros que son psico-socialmente incompatibles, esto afectará negativamente en el estado anímico y/o predisposición de sus integrantes. Si estas situaciones no son detectadas a tiempo pueden generar malos resultados, tanto en el rendimiento del equipo, como en el producto final obtenido.

Por otra parte, es importante medir periódicamente el desempeño de un equipo de trabajo, y entregar *feedback* de buena calidad a sus integrantes. Esto permitirá aumentar notablemente la confianza y cohesión entre sus miembros. Es importante resaltar que para cada integrante del equipo, esta es una etapa de autoconocimiento y aprendizaje, donde se espera que cada uno pueda conocer cómo lo ven y valoran los demás compañeros, y así poder compararse a cómo se ve a sí mismo, para luego proyectar lo aprendido en dicho ciclo.

1.1. Problema Abordado

Esta problemática también se presenta en el ambiente universitario, en particular, existen distintos cursos del Departamento de Ciencias de la Computación (DCC) de la Universidad de Chile, donde el trabajo en equipo es parte de lo que el alumno debe llevar a cabo. Por lo tanto, es pertinente y recomendado conformar equipos de trabajo compuestos por alumnos en base a distintos criterios (principalmente psicosociales), para

que en estos equipos haya buena cohesión, facilitando así el funcionamiento de los mismos.

Actualmente existen dos herramientas de software que se encargan de: (i) apoyar la conformación de los equipos a partir de una encuesta basada en aptitudes psico-sociales [1][3], y (ii) realizar una coevaluación con el fin de dar *feedback* a los alumnos en base a la percepción de sus pares [2].

A pesar de que actualmente las herramientas se encuentran funcionando de buena manera, es necesario añadir ciertas funcionalidades que permitan mejorar su usabilidad y utilidad, con el objetivo de facilitar el trabajo del equipo docente de los cursos que utilizan las herramientas. Dichas funcionalidades son las siguientes:

1. *Mantener la persistencia de la información entregada por los alumnos para la conformación de equipos:* Actualmente los estudiantes deben contestar la encuesta para la conformación de equipo todas las veces que den un curso nuevo que utilice dicha herramienta. Esta funcionalidad debe permitir reutilizar los resultados de las encuestas ya contestadas, dentro de un período de tiempo (se sugiere un plazo 2 años), dando al estudiante la posibilidad de volver a responder la encuesta antes de esa fecha, si lo desea.
2. *Registrar conflictos con otros pares:* Es sabido que si en un equipo de trabajo hay conflictos previos, la cohesión del mismo se verá afectada negativamente. Ahora si un alumno/a posee algún conflicto previo con algún compañero/a, debe poder registrar con quién no le gustaría trabajar, para así prevenir futuros conflictos durante el periodo de desarrollo. Esta información debe ser considerada por el sistema a la hora de conformar los equipos, y deberá mantenerse anónima para todos los usuarios del mismo.
3. *Analizar la inclusión de nuevos criterios para la asignación de personas a equipos:* En la configuración actual, el algoritmo que asigna personas a equipos no considera aspectos como notas de los alumnos, diversidad de género, o el desempeño del alumno en equipos anteriores, entre otros. En el marco de esta memoria se analizó la inclusión de otros criterios a dicho algoritmo, para así tratar de mejorar la estrategia de conformación de equipos.

4. *Permitir analizar la evolución de los alumnos en los diferentes cursos, en términos de su capacidad de trabajo en equipo:* Para ello se debe generar una base de información que permita estudiar la evolución de los alumnos, no sólo en los distintos cursos, sino también a lo largo de su carrera.
5. *Unificar herramientas:* La nueva versión de la herramienta debe permitir que los sistemas legados (el de conformación de equipos y el de co-evaluaciones) funcionen de manera integrada. Es decir, que se unifiquen a nivel de interfaces de usuario y de datos. Esto con el fin de facilitar la configuración y operación de la misma al equipo docente del curso, y así llevar un mayor control y gestión de cualquier curso en el que se necesite conformar y coevaluar equipos.

1.2. Objetivos de la Memoria

Este trabajo de título tiene como objetivo general realizar la reingeniería a las herramientas actuales de conformación y coevaluación de equipos, utilizadas en distintos cursos del DCC, para así facilitar su uso y extender su tiempo de vida esperado. Como parte de ese trabajo se busca, entre otras cosas, migrar a una versión actualizada de los lenguajes de programación utilizados para su construcción. Los objetivos específicos que se derivan del objetivo general son los siguientes:

- Diseñar y validar nuevas interfaces de usuario para las herramientas, que cumplan con criterios de usabilidad actuales.
- Refactorizar código de ambas herramientas, para lograr un escalamiento limpio y mantención del proyecto a largo plazo.
- Migrar ambas herramientas a versiones actualizadas de los lenguajes de programación con los que fue implementada.
- Plantear y evaluar soluciones a los problemas constatados de la herramienta de conformación de equipos.
- Implementar un servicio que brinde persistencia a los resultados de las encuestas psicosociales y coevaluaciones completadas por los alumnos, para que estas puedan ser utilizadas a futuro en otros cursos o como análisis de datos.

1.3. Resumen de la Solución

La solución propuesta pasa por realizar la reingeniería de las dos herramientas legadas. Con esto se busca inicialmente robustecer su funcionamiento y generar estabilidad en ellas, para luego facilitar la integración entre ambas. Para llegar al resultado deseado fue necesario estudiar los estados de las plataformas legadas, analizando su arquitectura, funcionalidades e interfaces. Este análisis permitió encontrar los puntos críticos de cada una de ellas, plantear mejoras y requerimientos de éstas.

Posterior a esto se analizaron las posibles soluciones y modificaciones a los sistemas, donde una de las principales modificaciones realizadas (de nivel estructural), fue establecer un nuevo modelo de interfaces de usuario a seguir. El nuevo modelo debía ser acorde a lo utilizado en otros sistemas del DCC, como también permitir realizar una estructuración a nivel de código, que facilitara la integración de ambas herramientas. Para llevar a cabo esto, fue necesario realizar cambios a nivel de arquitectura física y lógica de los sistemas, con el fin de proceder con la implementación e integración de ambos proyectos.

1.4. Estructura del Documento

A continuación se muestra la estructura del resto del documento, con una pequeña descripción de cada apartado.

- *Capítulo 2 - Análisis de los Sistemas Legados:* En este capítulo se describen y analizan ambas herramientas legadas; es decir, la herramienta de conformación de equipos, y el sistema de coevaluaciones. Aquí se presenta un análisis previo a la modificación de éstas. Este análisis permite indicar el estado actual de los sistemas, incluyendo su arquitectura, funcionalidad y las principales interfaces de estos.
- *Capítulo 3 - Rediseño de la Herramienta de Conformación de Equipos:* Dentro de este apartado se plantean los requisitos con los que debe cumplir la solución para la Herramienta de Conformación de Equipos. Para finalmente presentar cómo se llevó a cabo el desarrollo de la solución planteada.
- *Capítulo 4 - Rediseño de Sistema de Coevaluaciones:* De manera similar al capítulo anterior, se presentan los requisitos que debe cumplir el Sistema de

Coevaluaciones. Para luego presentar la arquitectura y el modelo de datos bajo el cual se desarrolló la solución.

- *Capítulo 5 - Evaluación de la Solución:* En este apartado se describe la evaluación de la solución implementada, validaciones de usuarios y validación de usabilidad del sistema.
- *Capítulo 6 - Conclusiones y Trabajo a Futuro:* En este capítulo se analiza el trabajo realizado y sus resultados finales, comparándolo con el cumplimiento de los objetivos planteados inicialmente. Además, se plantea cómo debe ser el trabajo en siguientes iteraciones del sistema.

2. Análisis de los Sistemas Legados

Para alcanzar los objetivos de este proyecto, se comenzó analizando el contexto a partir del cual surgen las herramientas, así como también su estado actual. Basado en ese conocimiento, se identificaron los componentes a intervenir para cumplir con los objetivos propuestos.

2.1 Herramienta de Conformación de Equipos

Actualmente la herramienta se encuentra operativa en la página web Conformación de Equipos¹. Ésta es utilizada principalmente por el equipo de cátedra de los cursos CC5401: Ingeniería de Software II y CC4402: Formulación, Evaluación y Gestión de Proyectos. Luego de algunos ajustes, la herramienta también podrá utilizarse en los cursos CC4401: Ingeniería de Software I y en CC5402: Proyecto de Software, pues estos usan criterios de agrupamiento distintos a los dos primeros cursos.

Esta herramienta permite que un docente establezca un conjunto de preguntas, que luego serán contestadas por los alumnos de los cursos, para finalmente conformar equipos en base a las respuestas obtenidas. Este software busca ser un apoyo para los docentes de esos cursos, permitiendo generar equipos de trabajo de manera automatizada y balanceada, con la intención de reducir el tiempo que demora esta tarea y minimizando los posibles errores producto del trabajo manual.

Por otro lado, existen tres tipos de roles para los usuarios que utilizan la herramienta: Administradores, Docentes y Alumnos. Estos roles tienen asociados permisos, los que van decreciendo según el rol, donde el primero de estos (es decir, el administrador) puede modificar todo el contenido de la herramienta, y el último (el alumno) solo puede contestar las preguntas de los formularios y visualizar los resultados de sus respuestas.

2.1.1 Estructura de la herramienta

La arquitectura física de la herramienta de conformación de equipos incluye un servidor, que posee una base de datos independiente y se conecta a Internet a través de un firewall. Este servidor se conecta a la *Base de Datos de Docencia DCC*, para obtener la

¹ <https://equipos.dcc.uchile.cl/>

información asociada a los alumnos y cursos que harán uso de la herramienta, según el año y semestre correspondiente. Luego, el usuario se debe autenticar en ella, utilizando la API de U-Pasaporte, para así poder acceder a la funcionalidad del sistema dependiendo del tipo de usuario que sea (Figura 1).

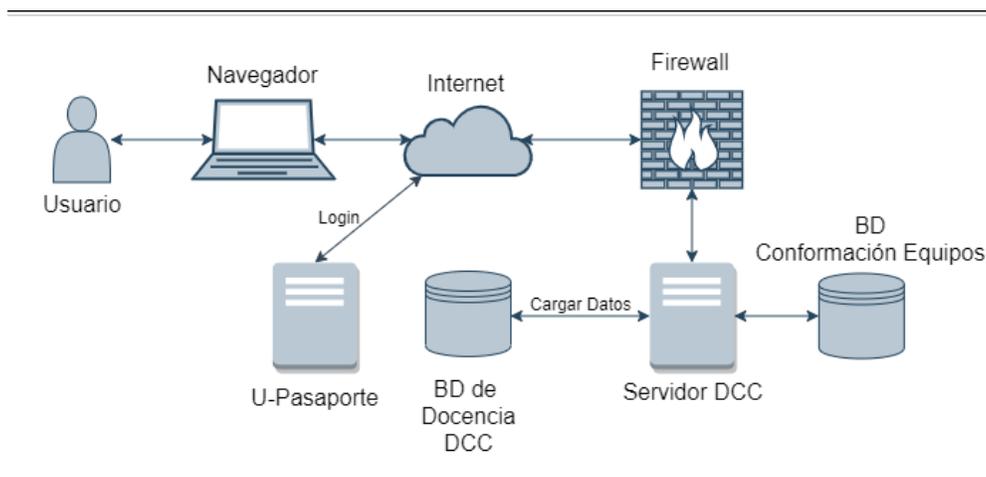


Figura 1: Arquitectura Física de la Herramienta de Conformación de Equipos

Por otro lado, la arquitectura lógica de la herramienta consta de un modelo de tres capas: Presentación, Lógica de Negocio, y Datos. Los usuarios de tipo Docente y Alumno pueden interactuar con la capa de presentación (es decir, con las interfaces de usuario de la aplicación), en cambio el Administrador, puede cambiar todo porque es el encargado de administrar qué cursos son los utilizados por la herramienta directamente en la capa de Datos.

La capa de presentación interactúa directamente con la capa de lógica de negocio, que recibe las peticiones del usuario, ejecuta los programas correspondientes, y luego envía de vuelta las respuestas del proceso a la capa de presentación para que el usuario pueda ver lo solicitado. Ahora bien, si para llevar a cabo dicho proceso es necesario acceder a la base de datos, para esto la capa de negocios es la encargada de solicitar los datos a la capa de datos, que es donde estos residen.

La mayor virtud de este tipo de arquitectura es que permite realizar modificaciones en las distintas capas, sin necesidad de ver afectadas a las otras, facilitando así la escalabilidad o mejoras de la herramienta. Además, permite que la base de datos esté

protegida, ya que solo puede ser modificada de manera directa por el Administrador. En la Figura 2 se muestra la arquitectura lógica de la herramienta, con la capa de Presentación en rojo, la de la Lógica de Negocio en azul y en verde la de Datos.

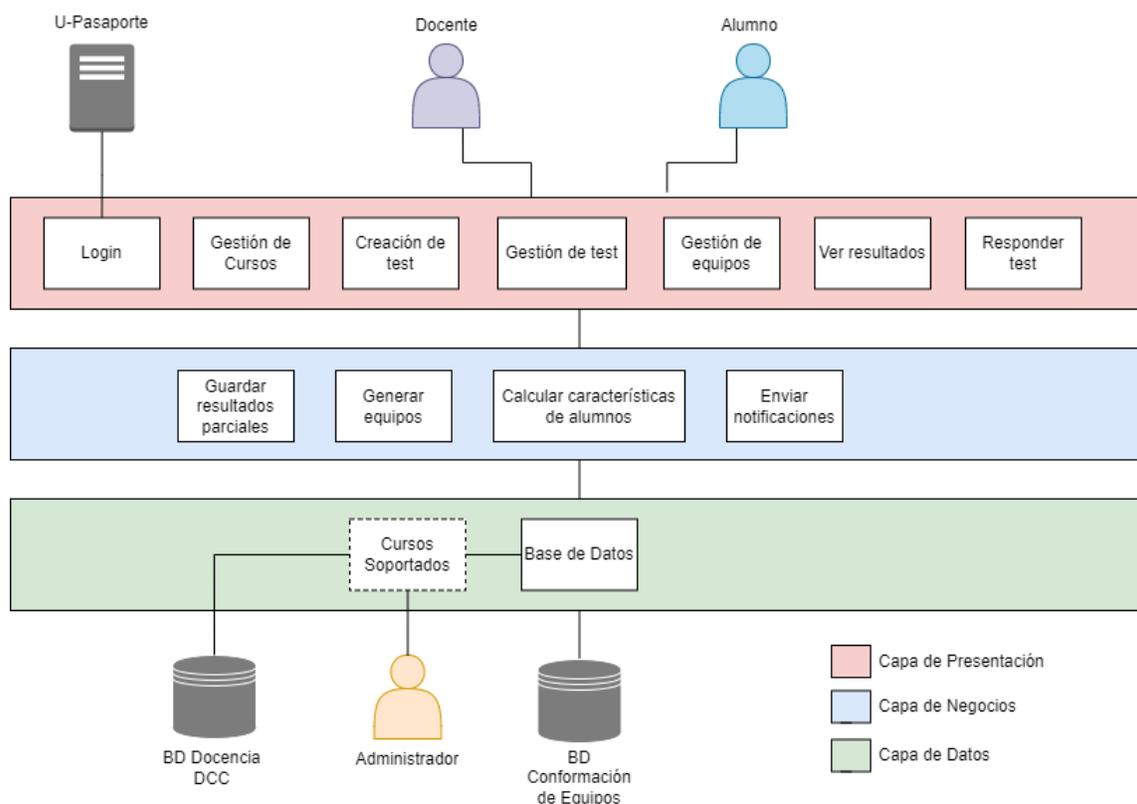


Figura 2: Arquitectura Lógica de la Herramienta de Conformación de Equipos

2.1.2 Tecnologías utilizadas

En el aspecto más técnico, el *Backend* de la herramienta está desarrollado con el framework *Django*, en su versión 1.11. El cual se ejecuta en un entorno con Python 3.4. En cambio, su *Frontend* está creado con el framework *Vue.js* en su versión 2.4. Actualmente, ambos frameworks se encuentran desactualizados, generando un peligro mayoritariamente en lo que es el Backend, ya que no cuentan con soporte de seguridad. Al día de hoy, la versión LTS (*Long Term Support*) de *Django* es la versión 3.2.

2.1.3 Interfaces de usuario

A continuación se muestran las principales interfaces de usuario de la herramienta de Conformación de Equipos, las que corresponden al punto de vista de un Docente; es decir, cuando un docente está logueado en el sistema. Después de hacer login en la página de autenticación, la interfaz principal es la mostrada en la Figura 3.



Figura 3: Página Principal de Herramienta de Conformación de Equipos.

En esta interfaz se pueden ver los cursos donde el usuario participa, ya sea como docente o auxiliar, agrupados en: “Cursos habilitados”, que corresponden a los cursos donde puede conformar equipos, y “Cursos pasados”, que son cursos que utilizaron la herramienta.

Para ver los test asociados a un curso, basta con hacer click en el nombre de uno de estos en la vista principal, donde se redirige a la interfaz mostrada en la Figura 4. La que muestra en detalle los test pertenecientes al curso. Las funcionalidades presentes en la interfaz son: borrar el test, ver la cantidad de alumnos que han respondido el test, cambiar el plazo límite para contestar, y finalmente armar los equipos correspondientes.



Figura 4: Página del detalle de un curso, de la Herramienta de Conformación de Equipos, desde el punto de vista de un Docente o Auxiliar.

Al presionar el botón de “Armar Equipo” te redirige a la interfaz mostrada en la Figura 5. La que permite generar los equipos de manera manual o automática, preguntando primero por la cantidad de equipos que quieres formar, además poder editar de manera manual los equipos una vez creados. Una vez el cuerpo docente esté conforme con los equipos, se puede descargar los resultados en un archivo Excel y publicar los equipos para que los alumnos los conozcan.

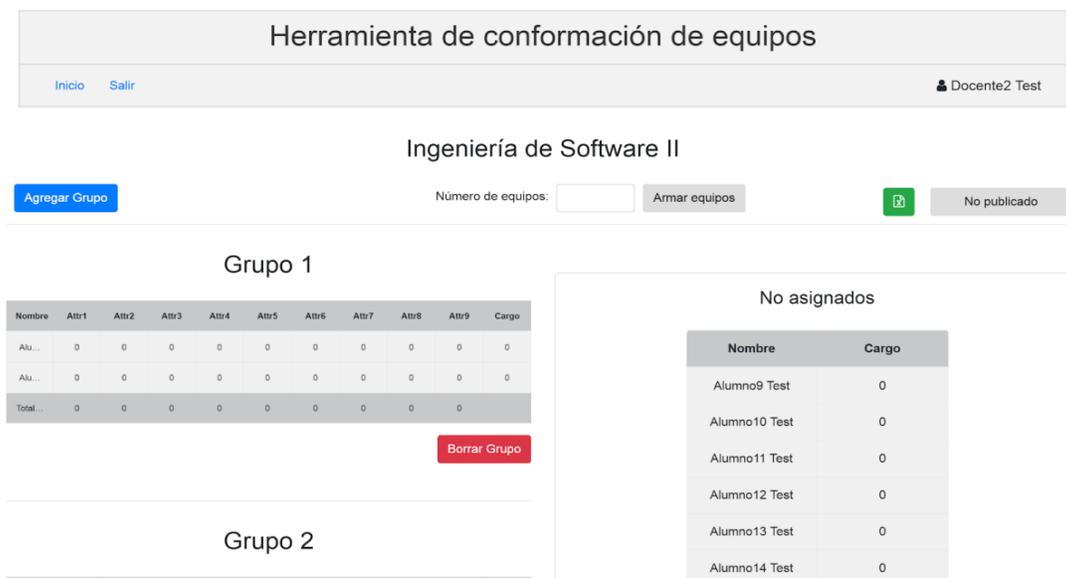


Figura 5: Página del detalle de un curso al momento de armar los equipos, de la Herramienta de Conformación de Equipos desde el punto de vista de un Docente o Auxiliar.

2.2 Herramienta de Coevaluación de Equipos

Una vez conformados los equipos de trabajo, y luego de que estos han trabajado durante el semestre en los proyectos de los distintos cursos, se vuelve necesario realizar una evaluación del desempeño de dichos equipos. Es por esto que se utiliza un sistema de coevaluación² entre pares, como complemento al sistema anterior. La herramienta de coevaluación permite a cada integrante del equipo, evaluar el desempeño de sus compañeros durante el desarrollo del curso. Finalmente, el sistema entrega información relevante al equipo docente sobre el desempeño individual y grupal de los alumnos en base a la coevaluación realizada.

Es importante destacar la importancia de esta herramienta, ya que permite medir el desempeño de un alumno, no solo a lo largo de un semestre, sino que también de todos los cursos que utilicen dicha herramienta a lo largo de su carrera universitaria. Esto permite obtener una retroalimentación mucho más completa, que facilita la definición de virtudes y falencias, además de poder ver el progreso del alumno en lo que refiere al trabajo en equipo.

Por otro lado, existen cuatro roles para los distintos tipos de usuarios del sistema: Administrador, Docente, Auxiliar, Alumno. Al igual que la herramienta de conformación de equipos, los permisos asociados a cada tipo de usuario van decreciendo de manera gradual dependiendo del rol. Particularmente, el Administrador puede tener el control de toda la herramienta, los Docentes pueden crear cursos, agregar a otros usuarios, asignar roles, crear equipos de trabajo, crear coevaluaciones, entre otras tareas. Por su parte, los Auxiliares pueden administrar las coevaluaciones asociadas al curso al que son auxiliares o ayudantes. Finalmente, los Alumnos sólo pueden seleccionar, responder o ver los resultados de las coevaluaciones recibidas.

2.2.1 Estructura de la herramienta

En el caso de la Herramienta de Coevaluaciones de Equipos, la arquitectura física está compuesta por un servidor que administra una base de datos, y se conecta a internet a través de un firewall, similar a la herramienta anterior (Figura 6). La diferencia entre ambos casos es que en la herramienta de coevaluaciones los datos utilizados son cargados de forma manual, o mediante archivos, por los usuarios administradores, y no obtenidos

² <http://coevaluaciones.dcc.uchile.cl/>

de la *Base de Datos de Docencia DCC*. Además, la autenticación en el sistema se hace de manera independiente, y no está asociada a la API de *U-Pasaporte*, como en el caso anterior, generando el problema de tener más de una credencial para sistemas internos del DCC.

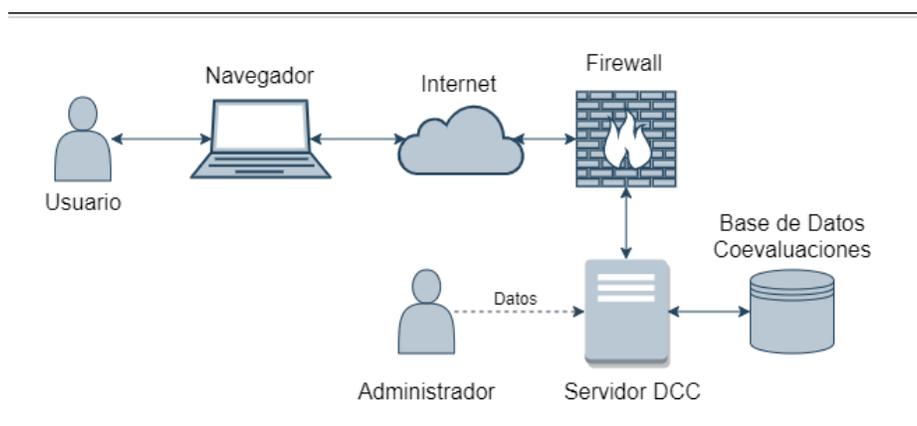


Figura 6: Arquitectura Física de la Herramienta de Coevaluación de Equipos.

La arquitectura lógica de la herramienta viene dada por un modelo de tres capas, similar al de la herramienta anterior: Presentación, Lógica de Negocio y Datos. En la Figura 7 se muestra la arquitectura lógica de la herramienta, con la capa de Presentación en rojo, la de Lógica de Negocio en azul y la de Datos en verde.

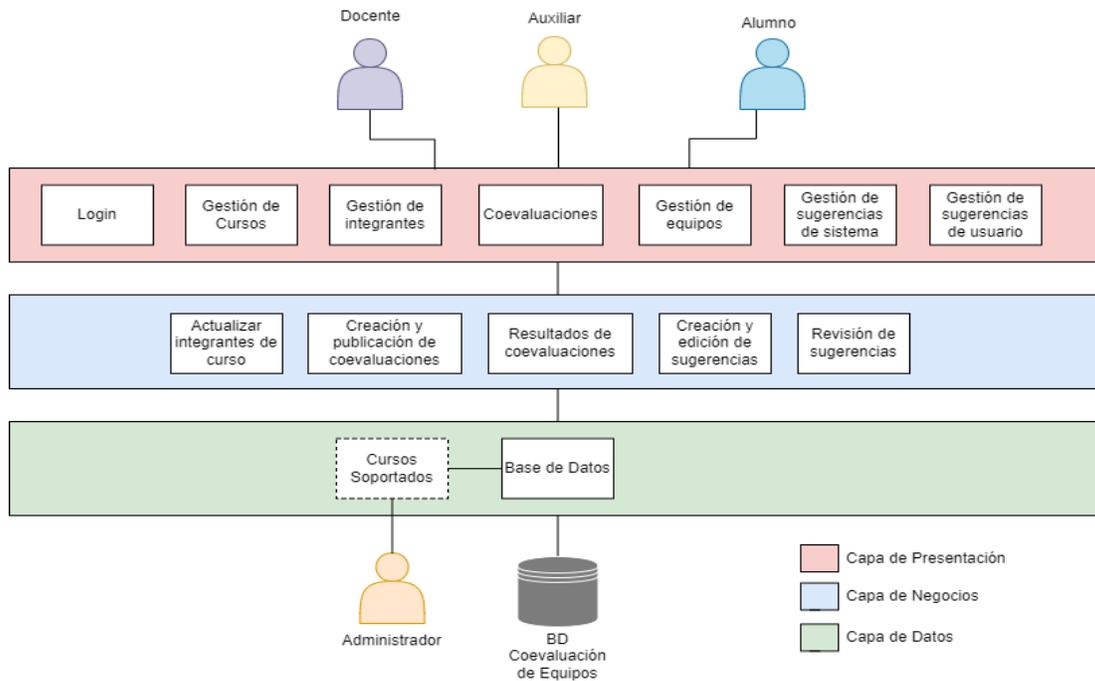


Figura 7: Arquitectura Lógica de la Herramienta de Coevaluación de Equipos.

2.2.2 Tecnologías utilizadas

Esta herramienta fue construida como un monolito de *Django*, esto quiere decir que las tres capas de la arquitectura lógica, se encuentran combinadas bajo una misma aplicación. Tanto el *Frontend* como el *Backend* de la herramienta está desarrollada con el framework *Django*, en su versión 1.5, donde se utilizan los templates nativos de *Django* y la librería Bootstrap 3 para generar las interfaces de usuario.

La herramienta se ejecuta en un entorno con Python 2.7. Al igual que la herramienta anterior, existe el problema que el framework está desactualizado, donde en este caso es aún más crítico ya que está montado sobre Python 2, el cual se encuentra sin soporte hace más de un año, generando graves problemas de compatibilidad y seguridad.

2.2.3 Interfaces de usuario

A continuación se muestran las principales interfaces de usuario de la herramienta de coevaluaciones, las que corresponden al punto de vista de un Docente; es decir, cuando

un docente está logueado en el sistema. Después de hacer login en la página de autenticación, la vista principal es la mostrada en la Figura 8.

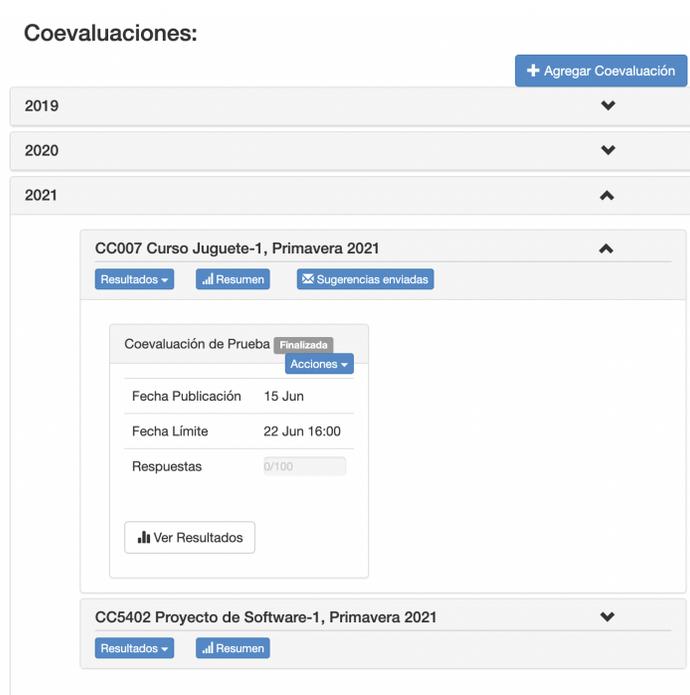


Figura 8: Página Principal de Herramienta de Coevaluación de Equipos.

En esa interfaz se aprecia que las coevaluaciones se encuentran agrupadas por año, para luego ser agrupadas por curso, además incluye un botón que permite crear nuevas coevaluaciones. Dentro de cada curso se pueden realizar distintas acciones, como por ejemplo, ver las sugerencias enviadas a los alumnos, o ver un resumen de las coevaluaciones asociadas al curso.

Para ver los resultados de una coevaluación en detalle, se debe hacer click en el botón de “Ver Resultados” de la vista principal, el cual redirige a una nueva vista que es lo mostrado en la Figura 9. Allí se permite descargar los resultados en un archivo Excel, ver detalles de estos resultados por equipo o por alumno.

Resultados:

Opciones ▾

Coevaluación: Coevaluación de Prueba

Curso: CC007 Curso Juguete-1, Primavera 2021

Profesor: Sergio Ochoa

¡No olvide revisar las sugerencias realizadas por los alumnos a sus compañeros [Aquí](#) !

Respondió coevaluación Respondió coevaluación, pero sin comentarios No respondió coevaluación

1.- Equipo 1 - Portal DCC		Descargar resultados del grupo	Ver resultados
Alumno			Nota final
<input type="checkbox"/> Floted Otino			-
<input type="checkbox"/> Ioltuigto Lodtito			-
<input type="checkbox"/> rotu Pipopo			-
<input type="checkbox"/> TauLote teyed			-
<input type="checkbox"/> Badtíad rutivil			-
<input type="checkbox"/> Guuteo lahuuquil			-
<input type="checkbox"/> Arevino Hepeta			-

Figura 9: Página de resultados de una coevaluación de Herramienta de Coevaluación de Equipos desde el punto de vista de un Docente o Auxiliar.

Finalmente, se añade la vista de los resultados de una coevaluación, desde un punto de vista de un Estudiante (Figura 10). En la interfaz se puede ver la nota final y la nota pertinente a cada pregunta evaluada por los compañeros de equipo del Estudiante logueado.

Coevaluaciones

Nota final obtenida: 7

Tus Compañeros te han evaluado de la siguiente forma:

Pregunta	Nota
Demuestra compromiso con el proyecto.	7
Cumple de manera adecuada con las tareas que le son asignadas.	7
Demuestra iniciativa para lograr el éxito del proyecto.	7
Mantiene buena comunicación con el resto del equipo.	7
Mantiene buena coordinación entre sus tareas y las de sus pares.	7
La calidad de su trabajo es la apropiada para lograr el éxito del proyecto.	7
Ofrece apoyo en las tareas que van más allá del rol asignado.	7
Es capaz de admitir sus equivocaciones y recibir críticas.	7
Fortalezas.	▼
Debilidades.	▼

Figura 10: Página de resultados de una coevaluación de Herramienta de Coevaluación de Equipos, desde el punto de vista de un Estudiante.

3. Rediseño de la Herramienta de Conformación de Equipos

Una vez planteados los objetivos y analizadas las herramientas que están involucradas en este trabajo de título, se procede a describir los rediseños planteados para cada una de ellas. En primer lugar, se indican los cambios que son necesarios realizar en cada herramienta, y la metodología seguida para llevarlos a cabo. Estos cambios son los que permitirán, a futuro, integrarlas de una manera más eficaz.

Dado el contexto analizado en la sección anterior 2.1, la herramienta de conformación de equipos se encuentra con bastantes limitantes respecto a lo que se espera como solución final. Por lo tanto, antes de realizar la integración con la herramienta de coevaluaciones, se deben realizar una serie de cambios para así conseguir una integración limpia y correcta.

El principal cambio a realizar consiste en diseñar y evaluar un nuevo algoritmo de conformación de equipos, el cual debe permitir añadir distintos tipos de criterios o restricciones adicionales al momento de conformar los equipos. Estos criterios son descritos a continuación.

3.1 Criterios para Conformación de Equipos

Actualmente la herramienta de Conformación de Equipos, al momento de ejecutar su algoritmo y generar los equipos, no existen criterios o restricciones que limiten el cómo se forman estos. Por ejemplo, algunas restricciones o criterios son los siguientes: considerar el género de los estudiantes, y su rendimiento en cursos anteriores, entre otros. Es por esto que se vuelve necesario modificar dicho algoritmo para así dar un enfoque donde se permita crear equipos más heterogéneos en todo sentido, intentando a su vez lograr equipos de desarrollo con mayor cohesión. Por esta razón se estudiaron diversos criterios que pudieran ayudar en ese ámbito.

3.1.1. Aspectos de Género

El primer criterio a evaluar es la distribución por género para los equipos, donde según la literatura al respecto [4][5], sugieren que no se pueden establecer correlaciones directas entre equipos conformados por solo mujeres, solo hombres o mixtos, y el

rendimiento que estos tengan. Esto debido a que los distintos estudios han abarcado escenarios opuestos, por ejemplo, en el caso planteado por *Løvold, Lindsjörn y Stray* [5], establecen equipos de dos tipos, (i) equipos conformados por los mismo alumnos (76% de los equipos) y (ii) equipos conformados por el profesor (24% de los equipos). Donde sugieren que los equipos equilibrados en género son irrelevantes, dando más énfasis a la comparativa de rendimiento según quien conformó el equipo, es decir, si el equipo fue conformado por el profesor o por los estudiantes (autoorganización).

Por otro lado, *Dzvonyar, Alperowitz, Henze, y Bruegge* [4] utilizaron equipos conformados por el docente, pero los resultados no fueron claros y se encuentran subrepresentados, ya que solo participaron 12 mujeres en el estudio, de un total de 80 participantes.

Esto representa en particular un hecho común en las carreras de Ciencias de la Computación, donde la cantidad de personas que se identifican con el género femenino es alrededor de un 20-25% del universo de alumnos [7]. Es por esto, que es necesario fomentar la colaboración entre géneros al momento de desarrollar software o incluso al momento de enseñar, ya que se puede acabar con el estereotipo o sesgo que tiene la carrera de Ciencias de la Computación [6].

Para contribuir con lo anterior, se recomienda en general, no dejar equipos de trabajo con solo un miembro de distinto género, es decir, siempre se debe tener al menos una dupla de integrantes que se identifiquen con el mismo género. Esto se debe a que en el primer caso se puede generar una especie de discriminación al momento de la toma de decisiones, al ser la persona de distinto género una minoría dentro del equipo.

3.1.2. Aspectos de Rendimiento Académico

Otro criterio considerado fue el rendimiento académico a lo largo de la carrera universitaria. En este caso, se busca una correlación entre equipos conformados por alumnos con rendimiento sobresaliente, regular o suficiente, y la capacidad de trabajo en equipo que poseen. Para esto fue necesario analizar los datos provenientes del historial académico de cada alumno, y compararlos con las notas de coevaluaciones de cursos anteriores.

El objetivo de este criterio es generar equipos más heterogéneos en lo que respecta a rendimiento académico, ya que actualmente esto no es considerado al momento de conformar equipos, generando en ocasiones equipos potencialmente desbalanceados. Esto es, equipos con gran parte de sus integrantes con rendimiento destacable, y otros equipos con integrantes de bajo rendimiento académico.

3.1.3. Conflictos entre pares

El último criterio a considerar para conformar equipos, es que un alumno pueda seleccionar con qué compañero/a de equipo no le gustaría trabajar, por ejemplo, porque existe un conflicto previo entre ellos. Esto en busca de prevenir futuros conflictos en el periodo de ejecución del proyecto.

Por razones obvias, un alumno sólo se podrá registrar como máximo 5 conflictos. Este número máximo por el momento es arbitrario, y busca evitar que alguien señale tener conflictos con todos sus compañeros, lo cual sería un claro error de diseño. A futuro se determinará este valor con mayor precisión.

Un aspecto a considerar al momento de guardar los registros de conflictos, es que los datos deben ser anonimizados. Esto se hace para salvaguardar futuras filtraciones de información, y también para abordar el aspecto ético en el manejo de la misma.

3.2. Rediseño de interfaces

Con el fin de mejorar la experiencia de usuario y la usabilidad de la herramienta, se realizó un cambio en el diseño de las interfaces de usuario. Para esto se utilizó la herramienta de diseño Figma³, la que permite generar diseños de alta fidelidad y flujos de navegación. Esto fue de gran ayuda al momento de crear nuevas interfaces, ya que facilitó el hacer las pruebas de usabilidad, y validar las interfaces con usuarios antes de implementarlas. Es decir, esto permitió realizar cambios en el diseño de las mismas, en caso de ser requeridos cuando surgieron observaciones durante las pruebas.

Antes de comenzar con el diseño de interfaces, se hizo un estudio para obtener datos y tomar referencias de herramientas usadas (en el ámbito educacional) por los posibles usuarios de esta herramienta; es decir, alumnos, auxiliares y docentes del DCC. El

³ <https://www.figma.com/>

fin de esto fue crear diseños que resulten familiares con los usuarios, y por consecuencia, faciliten la usabilidad de la herramienta.

Es por esto que se usa una paleta de colores idéntica a usada en la página principal del DCC⁴, al igual que algunas de las nuevas herramientas que se están creando. La paleta se muestra en la Figura 11.



Figura 11: Paleta de colores elegida en base a la usada en herramientas del DCC.

Una vez elegida la paleta de colores y analizadas las aplicaciones más usadas por los usuarios de esta herramienta, se comenzó a diseñar la interfaz principal, la cual se muestra en la Figura 12. En esta interfaz se puede observar que es la integración de las interfaces mostradas en las Figuras 3 y 4. Esto permite reducir en uno la cantidad de vistas, mostrando la información de manera directa.



Figura 12: Interfaz principal de la nueva Herramienta de Conformación de Equipos.

⁴ <https://dcc.uchile.cl>

Esta mejora en la usabilidad se consigue en gran medida gracias al uso de un *sidebar*, el cual se encarga de mostrar los cursos en los que el usuario logueado participa. Además, es importante señalar que la información asociada a los cursos es obtenida directamente desde el *Data Lake* de docencia del DCC, el cual no deja espacio para la introducción de errores al cargar a los participantes de los cursos (es decir, profesores, auxiliares, ayudantes y alumnos).

Por otro lado, en el cuerpo de la página se muestran los cuestionarios o test pertenecientes al curso seleccionado, permitiendo al docente crear nuevos test, cambiar los plazos de los existentes y armar los equipos de estos.

En la Figura 13 se muestra la vista de armar los equipos, a la cual se llega desde la vista principal, haciendo click en el botón “Armar Equipos”. Acá se mantienen las mismas funcionalidades de la interfaz antigua (Figura 5), sólo se aplicó la nueva paleta de colores.



Figura 13: Interfaz de conformación de equipos de la Herramienta de Conformación de Equipos

3.3. Implementación de ajustes al diseño del sistema

La implementación de las nuevas interfaces y los ajustes al sistema necesarios para implementar los criterios vistos en la sección anterior 3.1. en conjunto con el profesor guía, se decidió postergarla y realizarla sólo en caso de que alcanzara el tiempo de este trabajo de título. Esto debido a que no era prioritario en el desarrollo del proyecto, dando énfasis en la correcta implementación de la otra herramienta, que era la que se encontraba más obsoleta y con mayores vulnerabilidades.

4. Rediseño del Sistema de Coevaluaciones

En base al análisis realizado en la sección anterior 2.2, se establece lo que es necesario y prioritario para cumplir con los objetivos de este trabajo de título, que implica una actualización y refactorización profunda a esta herramienta. El principal motivo de esto, se debe a que es incompatible con la herramienta de conformación de equipos en lo que respecta a la versión de Python. En particular, se busca migrar desde la versión de Python 2.7, a la versión 3.8. Para esto, será necesario una reconstrucción casi total de la herramienta, donde se debe buscar qué elementos se encuentran deprecados y realizar los cambios correspondientes, con el objetivo de mantener la robustez existente y añadirle la extensibilidad deseada.

4.1. Principales requisitos de la solución

Antes de abordar la reingeniería de la herramienta, es necesario definir cuáles son los principales requisitos que permitirán realizar la labor comprometida; entre los que se destacan los siguientes:

- *Cambio de versión de Python:* Tal como se mencionó con anterioridad, se busca migrar la versión de Python desde la 2.7, que perdió el soporte el año 2020, a una que posea soporte por al menos unos años más, en particular se elige la versión 3.8, la cual cuenta con soporte hasta el año 2024⁵. La importancia de contar con soporte es muy alta, debido a que ante cualquier problema de seguridad se cuenta con parches oficiales que lo solucionen. Al no contar con ese soporte, se tendrá que utilizar parches no oficiales de terceras personas que podrían contener software malicioso o en el peor de los casos, tener que repararlo por sus propios medios. Por otro lado, es importante recalcar que al cambiar la versión de Python, es necesario también actualizar la versión del framework utilizado en la herramienta, Django. Como se señala en la sección 2.2.2, la versión de Django que se usa es la 1.05, la que se encuentra deprecada hace más de 5 años.

⁵ <https://www.python.org/downloads/>

- *Mantenimiento de todas las funcionalidades preexistentes:* De la mano con el requisito anterior, una vez actualizada la versión de Python es necesario asegurar la mantención de todas las funcionalidades preexistentes. Se debe tener en cuenta que, con el cambio de versión de Python, habrá muchos elementos que quedarán obsoletos o deprecados, por lo que será necesario actualizarlos para mantener su funcionalidad.
- *Implementación de un nuevo login para usuarios:* Actualmente en casi todas las herramientas del DCC se está estandarizando el uso del Portal DCC⁶ como “puente” de login, o mejor dicho, como “single-sign-on” (SSO) para acceder a las herramientas de uso interno del Departamento. Dicho portal utiliza las credenciales de la cuenta UChile⁷ del usuario, el cual permite acceder a distintas herramientas con solo unas credenciales, sin necesidad de poseer una credencial distinta por cada herramienta. Dado este beneficio, es necesario cambiar el login antiguo de la herramienta e integrarlo al Portal DCC, facilitando al usuario el ingreso a esta herramienta.
- *Diseño e implementación de nuevas interfaces de usuario:* Si bien uno de los objetivos de este trabajo de título es diseñar e implementar nuevas interfaces de usuario, es necesario señalarlo también como requisito para esta herramienta. Particularmente, se debe mejorar sustancialmente las interfaces de usuario y la usabilidad de la herramienta, pues es uno de sus principales problemas. Además, ésta no ha sido modificada desde su creación, por lo que está bastante desactualizada en cuanto a diseño de interfaces.

4.2. Implementación de ajustes al diseño del sistema

Una vez planteados los requisitos que debe cumplir la solución, se procedió a trabajar en ella. En primer lugar se partió con el requisito de actualizar la versión de Python, ya que es la base para cumplir el objetivo principal de este trabajo de título. Para cumplir con él se utilizaron dos estrategias distintas, las que son detalladas a continuación:

⁶ <https://portal.dcc.uchile.cl/>

⁷ <https://mi.uchile.cl>

1. **Cambiar paulatinamente la versión de Python y Django:** Esta estrategia era la más señalada en diversas fuentes, dado que es avanzar con pequeños cambios hasta llegar a la versión deseada. En particular, consiste en realizar actualizaciones a las versiones estables tanto de Django y Python, avanzando hasta llegar a las versiones deseadas. Al momento de usar esta estrategia se nota la lentitud que ésta provoca, ya que por cada cambio de versión se debe testear todas las funcionalidades de la herramienta mediante *prueba-error* y corregirlas. Este problema podría haber sido resuelto de mejor manera o incluso evitado, si la herramienta legada tuviese un sistema de testing unitario que se pudiera aplicar a las funcionalidades. Al cabo de casi 2 semanas, se decidió no seguir con esta estrategia debido a que aún no se conseguían los cambios deseados y se podría retrasar el avance del proyecto.
2. **Cambiar directamente a la versión deseada y uso de boilerplate DCC:** Luego de no conseguir los avances deseados, se solicita ayuda técnica al encargado del Área de Desarrollo de Software del DCC. Donde se recomienda utilizar un nuevo boilerplate de herramientas internas del DCC⁸, que brinda bastantes facilidades a la hora de desarrollar código. Por ejemplo, posee automatizaciones para ejecutarlo de manera local, integración directa con el SSO del Portal DCC y correctores de código que facilitan las buenas prácticas. Este boilerplate está construido utilizando el framework Django en su versión 4.0 y ejecutado en un ambiente de Python 3.8. Ahora la tarea principal era migrar la herramienta a este boilerplate para conseguir una migración limpia. Esto se realiza copiando el contenido de los archivos de la herramienta, pegando en archivos creados en el boilerplate, probando que todo funcionara de manera correcta, y corrigiendo en caso de que no lo hiciera. Proceso que duró alrededor de 4 semanas, pero el mayor beneficio fue que se completaron rápidamente los 3 primeros requisitos planteados en la sección anterior 4.1. Debido que se mantuvieron todas funcionalidades existentes, se migró a una versión actualizada tanto de Python como Django, y se logró la implementación de un nuevo login usando el SSO del Portal DCC.

Un cambio importante al realizar la migración a Python 3.8 y Django 4.0 fue el modo en que Django realizaba las migraciones de su base de datos,. Con la versión antigua de la herramienta éstas se hacían mediante un software de terceros llamado *South*⁹, pero Django a partir de su versión 1.7 incluye esta funcionalidad en su core. Esto al momento

⁸ <https://github.com/DCC-FCFM-UCHILE/django-boilerplate>

⁹ <https://south.readthedocs.io/en/latest/>

de crear la nueva base de datos generó varios problemas por su incompatibilidad, los que fueron solucionados nuevamente mediante *prueba-error* debido a la falta de test.

Finalmente, para probar que estos cambios funcionaron de manera correcta, se decide poblar con un extracto de la base de datos de la herramienta de producción, lo cual luego de borrar todas las tablas relacionadas a *South*, funcionó sin problemas.

Por otro lado, un cambio relevante en lo que respecta la información asociada a los cursos, es que ahora ésta es obtenida directamente desde el Data Lake de docencia del DCC, y no es necesario que el administrador de la BD o el profesor del curso esté agregando de manera manual la información correspondiente a los cursos. Esto reduce la cantidad de errores al momento de cargar dicha información, ya que se obtiene de manera automática. La información es la oficial de cada curso, considerando el año y semestre, la sección del curso, los alumnos, y profesores, entre otros. Para efectos de este trabajo de título se mantienen activas ambas formas de obtener los datos, ya sea manual a través de un Administrador o Profesor, como también de manera automática. Esto se debe a que la información del Data Lake se debe consumir a través de una API, y ésta aún no está oficialmente en producción, pero sí se puede utilizar.

En la Figura 14, se aprecia en color naranja los cambios realizados en la arquitectura física de la Herramienta de Coevaluaciones, comparado a la antigua arquitectura de la Figura 6. Donde se observa que tanto el Administrador como el Data Lake, proveen de datos el *backend* de la Herramienta de Coevaluación alojado en los servidores del DCC.

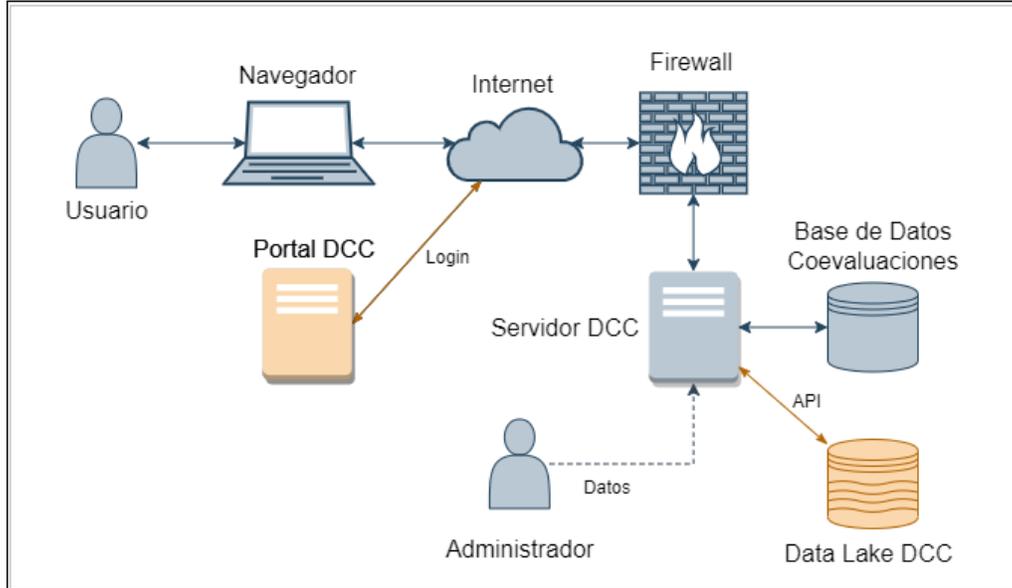


Figura 14: Nueva Arquitectura Física de la Herramienta de Coevaluación de Equipos

En el caso de la arquitectura lógica, tal como se observa en la Figura 15, se añaden los cambios antes mencionados, es decir, la incorporación del Data Lake DCC como proveedor de datos, el Portal DCC para la autenticación de los usuarios en la capa de presentación, y la capa de negocios como un servicio externo.

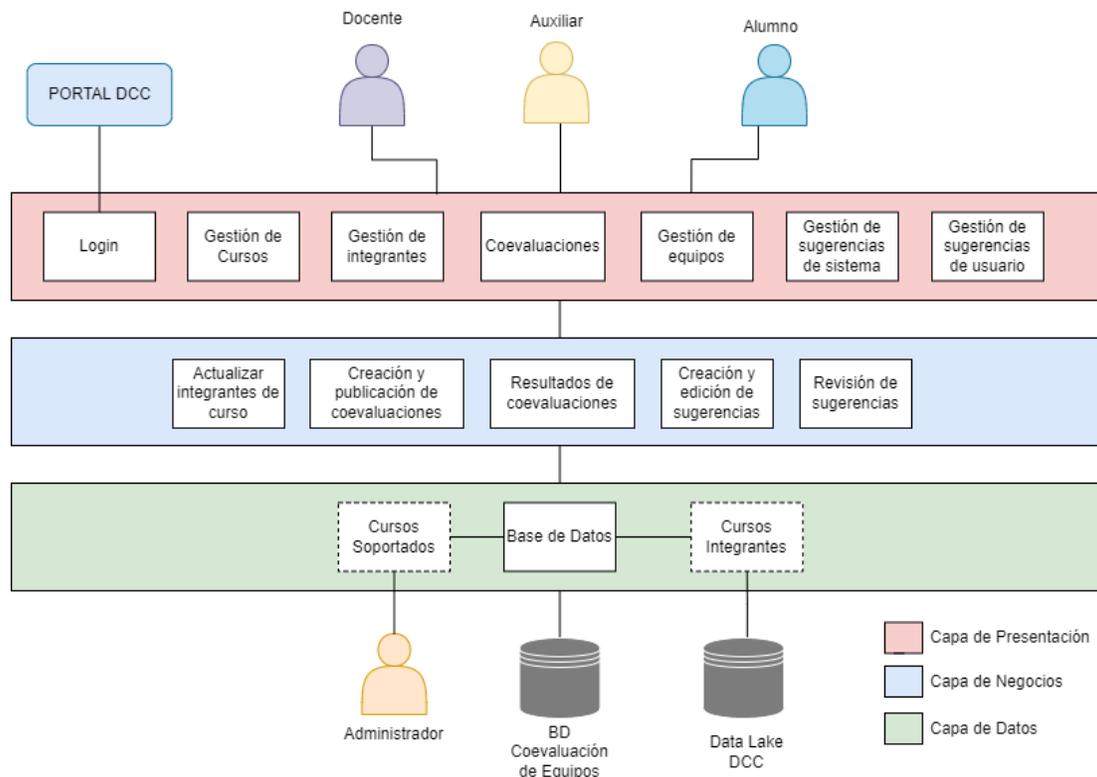


Figura 15: Nueva Arquitectura Lógica de la Herramienta de Coevaluación de Equipos.

4.3. Rediseño de interfaces

El proceso de rediseño de las interfaces de usuario para esta herramienta se realizó en paralelo, y de igual forma que la Herramienta de Conformación de Equipos. Es por esto que comparten como objetivo principal el mejorar la experiencia de usuario y la usabilidad. En esta sección se muestran las propuestas de diseño planteadas para cumplir con el objetivo mencionado con anterioridad.

Luego de analizar las falencias de usabilidad que existían en las interfaces antiguas, se plantean dos propuestas de diseño para la interfaz principal a los usuarios que ayudaron a evaluarlas, las cuales se pueden ver en las Figuras 16 y 17. Como elementos comunes se tiene el *sidebar* al lado izquierdo, el que permite acceder a los distintos cursos

donde el usuario logueado es participante. El otro elemento en común es el *navbar*, que permite navegar entre las dos herramientas cuando éstas se encuentren integradas.

En la primera propuesta (Figura 16) se observa que el cuerpo o wrapper de la herramienta posee una estructura de tarjetas, las cuales se usan para agrupar y mostrar las coevaluaciones asociadas a un curso, además de mostrar una tarjeta invisible para añadir una nueva coevaluación. En particular esta interfaz fue bien evaluada por los usuarios evaluadores, dado que mostraba la información de manera bastante clara y legible.



Figura 16: Primera propuesta de interfaz principal para Herramienta de Coevaluación de Equipos.

La segunda propuesta (Figura 17) consiste en un listado seleccionable de las coevaluaciones asociadas a un curso, donde la información mostrada era igual a la anterior propuesta. Sin embargo, el mayor beneficio observado por los usuarios era que en el caso hipotético de poseer muchas coevaluaciones para un curso, esta vista mostraría la información de manera mucho más ordenada. Además, en el caso de ser una lista seleccionable, ésta permite agrupar coevaluaciones, y por ejemplo, añadir un botón que

permita descargar los resultados seleccionados, mejorando la usabilidad de la herramienta al no ser necesario descargar los resultados uno a uno.



Figura 17: Segunda propuesta de interfaz principal de la herramienta

En el caso de la Figura 18, se muestra una vista similar a la de Conformación de Equipos (Figura 13). Esto debido a que puede ser necesario querer realizar alguna modificación a los equipos antes de iniciar un proceso de coevaluación.



Figura 18: Interfaz de modificación de equipos de la Herramienta de Coevaluación

Finalmente, en la Figura 19 se muestra el detalle de una coevaluación, donde en primer lugar se muestran los resultados de esta, agrupados por equipos. En cada equipo se muestra la nota de cada integrante y si ese integrante respondió o no con comentarios. También se añade una barra de navegación entre “pestañas” o *tabs*, que permite enviar sugerencias, ver las sugerencias enviadas por los alumnos a sus compañeros y un menú para editar la coevaluación, donde se puede modificar los plazos o el nombre de ésta. Además, se añaden botones de acción directa, evitando el uso de botones con *menus dropdown*, tales como el Descargar Resultados por curso o por equipo, Enviar los resultados por mail a los alumnos o ver en detalle los resultados de una coevaluación de un equipo.


Conformación de Equipos Coevaluación de Equipos Profesor

MIS CURSOS
Otoño 2022
 CC5401-1: Ingeniería de Soft ...
 CC5401-2: Ingeniería de Soft ...
 CC5402-1: Proyecto de Soft ...
Primavera 2021
 CC5401-1: Ingeniería de Soft ...
 CC5401-2: Ingeniería de Soft ...
 CC5402-1: Proyecto de Soft ...
 Semestres anteriores

Coevaluación Iteración 1
 CC5401-1: Ingeniería de Software II 2022
 Resultados | Sugerencias de Alumnos | Sugerencias Enviadas | Editar
 Descargar Resultados Curso | Enviar Resultados

Respondió la coevaluación |
 Respondió la coevaluación, pero sin comentarios |
 No Respondió la coevaluación

1. Equipo 1

Nombre	Nota
<input checked="" type="checkbox"/> Nombre Alumno 1	7.0
<input checked="" type="checkbox"/> Nombre Alumno 2	7.0
<input checked="" type="checkbox"/> Nombre Alumno 3	6.3

Figura 19: Interfaz del detalle de una coevaluación

5. Evaluación de la Solución

En este capítulo se describen las evaluaciones hechas a los criterios para conformar equipos, y a los diseños de las herramientas. Para ello se señala la metodología utilizada y los resultados obtenidos.

5.1. Evaluación de criterios para la conformación de equipos

Para obtener datos relevantes en lo que respecta a los criterios para la conformación de equipos, se realizó una encuesta a 22 estudiantes del DCC (14 identificados con el género masculino, 6 con el género femenino y 2 no binario). Esta encuesta busca establecer distintos puntos de vista a los tres criterios que se quieren considerar al momento de conformar equipos, mediante una serie de preguntas, las que son detalladas a continuación:

1. *Género*: Prefiero trabajar en grupos heterogéneos por sobre grupos homogéneos.
2. *Rendimiento académico*: Creo que afecta el rendimiento académico al momento de trabajar en equipo
3. *Conflicto entre pares*: Me gustaría seleccionar con quien no trabajar.

Se utilizó la escala Likert para cuantificar las respuestas a esta encuesta. En dicha escala hay 5 respuestas posibles para cada pregunta, las que van de manera creciente desde 1 - *totalmente en desacuerdo* hasta 5 - *totalmente de acuerdo*.

En el primer criterio "*género*", se observa en la Figura 20 que casi un 86,4% de los encuestados sí tiene preferencia en trabajar en grupos heterogéneos, por sobre grupos homogéneos, donde se entiende que un grupo heterogéneo está compuesto por estudiantes de diferentes géneros. Solo un 13,6% de ellos le causa indiferencia el cómo se conforma el equipo de trabajo, pero no hay nadie en desacuerdo con usar ese criterio para armar los equipos.

Prefiero trabajar en grupos heterogéneos por sobre grupos homogéneos.

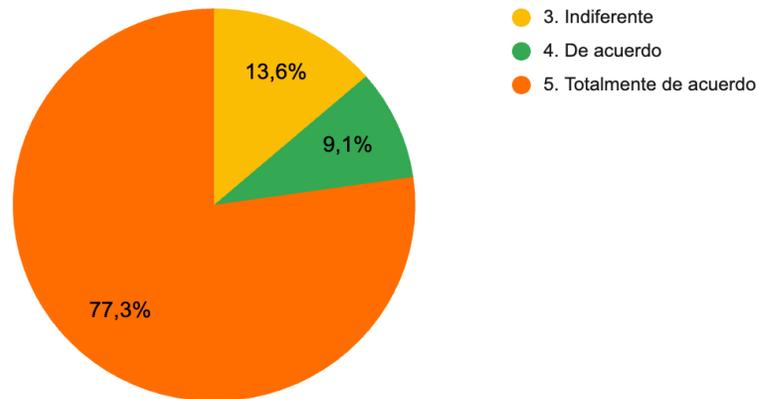


Figura 20: Preferencias respecto a trabajar en grupos heterogéneos en términos de género.

Para el segundo criterio “*rendimiento académico*”, se observa en la Figura 21 que los resultados son más dispersos comparados al criterio anterior. Más de la mitad de los encuestados, mejor dicho sobre el 60%, considera que no es relevante o no está de acuerdo con que el rendimiento académico durante la universidad, sea un criterio que afecte al momento de trabajar en equipo. En conversación con algunos de los encuestados, se plantea que este criterio es bastante arbitrario, debido a que el rendimiento académico puede verse afectado por diversos motivos, lo cual no es un buen parámetro a considerar para la formación de equipos.

Por otro lado, también se plantea que existe la problemática de los equipos conformados con “*gente de rendimiento académico sobresaliente*”. Los equipos donde todos sus integrantes son sobresalientes tienden a tener más problemas, debido a que buscan la perfección en lo que hacen, generando conflictos entre ellos, que a la larga termina afectando de manera significativa la cohesión entre ellos.

Creo que afecta el rendimiento académico al momento de trabajar en equipo

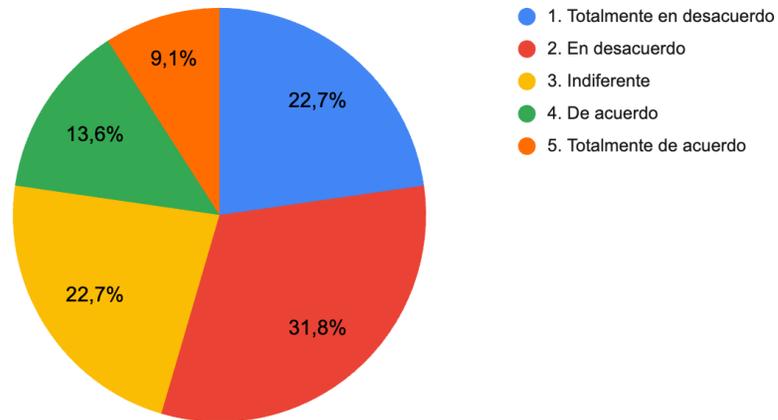


Figura 21: Preferencias respecto a considerar el rendimiento académico en la conformación de equipos

Finalmente, el último criterio a evaluar fue el de “*conflicto entre pares*”, donde solo una persona de los encuestados señala estar en desacuerdo con este criterio, y no le gustaría señalar con quien no trabajar. En cambio, al 27% de los encuestados le es indiferente seleccionar a sus compañeros. Por lo tanto, la amplia mayoría de los encuestados está a favor de incluir este criterio al momento de conformar los equipos.

Me gustaría seleccionar con quien no trabajar.

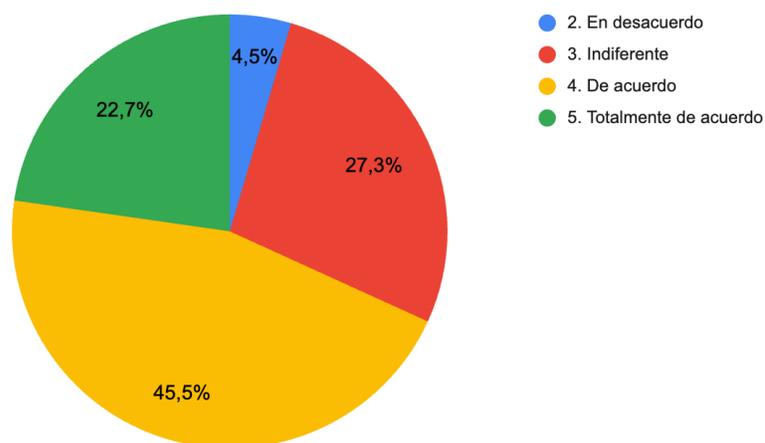


Figura 22: Preferencias respecto a seleccionar con quien no trabajar.

5.2. Evaluación de la usabilidad de las herramientas

A continuación se presentan los instrumentos para evaluar la usabilidad y utilidad de las herramientas, luego de la integración realizada a nivel de datos y el rediseño de sus interfaces.

5.2.1. Definición de System Usability Scale

Para realizar la evaluación de usabilidad de los diseños propuestos para ambas herramientas, se utiliza el “sistema de escala de usabilidad”, o más conocida en inglés como: System Usability Scale (SUS) [8]. Esta escala que permite evaluar la usabilidad de acuerdo a la percepción de los usuarios sobre un sistema, enfocándose en 3 aspectos:

1. Eficacia: ¿los usuarios pueden alcanzar con éxito sus objetivos?
2. Eficiencia: ¿cuánto esfuerzo es necesario para que pueda alcanzar esos objetivos?
3. Satisfacción: ¿el uso del sistema fue satisfactorio?

Posee beneficios bastante prácticos a la hora de utilizarla, en particular, puede ser utilizada en pequeños grupos de usuarios y sus resultados siguen siendo confiables. Además, es una encuesta breve porque para contestarla no es necesario redactar las respuestas ni invertir demasiado tiempo descifrando el enunciado. Está compuesta de 10 enunciados ya definidos y sus respuestas utilizan la escala Likert, con puntajes que van del 1 al 5 según qué tan de acuerdo se encuentra el evaluador respecto al enunciado. A continuación se muestran los enunciados y respuestas realizadas por 5 usuarios:

1. *Creo que me gustaría utilizar este sistema con frecuencia.*
2. *Encontré el sistema innecesariamente complejo.*
3. *Pensé que el sistema era fácil de usar.*
4. *Creo que necesitaría el apoyo de un técnico para poder utilizar este sistema.*
5. *Encontré que las diversas funciones de este sistema estaban bien integradas*
6. *Pensé que había demasiada inconsistencia en este sistema.*

7. *Me imagino que la mayoría de la gente aprendería a utilizar este sistema muy rápidamente.*
8. *Encontré el sistema muy complicado de usar.*
9. *Me sentí muy seguro usando el sistema.*
10. *Necesitaba aprender muchas cosas antes de empezar con este sistema.*

La escala asociada a las respuestas de los evaluadores fue la siguiente:

1. *Totalmente en desacuerdo*
2. *En desacuerdo*
3. *Indiferente*
4. *De acuerdo*
5. *Totalmente de acuerdo*

Los enunciados de número impar como se puede notar son positivos, en cambio los pares son negativos.

$$R_{\text{IMPAR}} = (R1 + R3 + R5 + R7 + R9) - 5 \quad (1)$$

$$R_{\text{PAR}} = 25 - (R2 + R4 + R6 + R8 + R10) \quad (2)$$

$$R = (R_{\text{IMPAR}} + R_{\text{PAR}}) * 2.5 \quad (3)$$

Para calcular el puntaje final de esta encuesta, y que tan usable es la herramienta se utiliza la ecuación (3) cuyo resultado se encuentra en una escala entre 1-100. Las escalas de usabilidad se establecen según los siguiente rangos para el valor de R :

- Menos de 25 puntos, el escenario de usabilidad es “el peor imaginable”.
- Entre 25 y 38, la usabilidad es considerada “pobre”.
- Entre 39 y 52, la usabilidad es considerada “ok” (puede ser considerado un mínimo aceptable).
- Entre 53 y 73, la usabilidad es considerada “buena”.
- Entre 74 y 85, la usabilidad es considerada “excelente”.

- Entre 86 y 100, el escenario es “lo mejor posible”.

5.2.2. Resultados de la Encuesta SUS

En esta sección se muestran los resultados de la encuesta SUS, se aprecian los puntajes que le asigna cada usuario, los resultados finales y un promedio de cada uno de los enunciados, que permitirá obtener conclusiones y posibles puntos de mejoras.

Dados los resultados mostrados en la Tabla 1, la usabilidad de las nuevas versiones de las herramientas caen en la categoría “lo mejor posible” de acuerdo a la escala SUS. Aún así, es posible notar algunos puntos que se podrían mejorar; por ejemplo, el enunciado 5 es el que posee el puntaje más bajo, se refiere a la integración de las funcionalidades de la aplicación. Esto se puede explicar debido a que la encuesta fue realizada durante el proceso de diseño, y no luego de haber implementado la solución. Además, la encuesta fue realizada mediante la técnica Wizard of Oz¹⁰, la que simula la interacción entre el usuario y el sistema, y por lo tanto acota las funcionalidades percibidas por el usuario.

Otro puntaje a considerar es el enunciado 6, ya que es de los más altos y se refiere a la inconsistencia del sistema, donde al igual que el enunciado 5 se puede ver afectado por el uso de Wizard of Oz al realizar la encuesta.

Tabla 1: Resultados de encuesta SUS

Usuario Enunciado	1	2	3	4	5	6	7	8	9	10	Resultados
Usuario 1	4	2	5	1	4	1	5	1	5	1	92,5
Usuario 2	3	1	5	1	4	1	4	2	4	2	82,5
Usuario 3	4	1	5	1	3	2	5	1	5	1	90
Usuario 4	5	2	4	1	4	2	4	1	5	1	87,5
Usuario 5	5	1	4	1	4	2	5	1	5	1	92,5
Promedio	4,2	1,4	4,6	1	3,8	1,6	4,6	1,2	4,8	1,2	89

¹⁰ <https://ux4sight.com/blog/wizard-of-oz-prototyping>

6. Conclusiones y Trabajo a Futuro

Como se puede apreciar en los capítulos anteriores, este trabajo de título se enfocó principalmente en hacer reingeniería a dos sistemas utilizados en el DCC, como lo son la herramienta de Conformación de Equipos y la herramienta de Coevaluación de Equipos. En particular se buscaba realizar una reingeniería que permitiera la integración completa entre ambas herramientas, aprovechando de mejorar la usabilidad de los sistemas, facilitando su uso y extendiendo la vida útil de éstas.

La motivación principal de esto surge de diversos motivos, como por ejemplo, ambas herramientas se encontraban separadas pero los datos que utilizan estaban muy relacionados entre sí. Dado eso, el cuerpo docente de los cursos que las utilizan deben hacer dos trabajos por separado, configurarlas cada semestre y hacer uso de ellas; por lo tanto, su integración facilita estas labores. Otro motivo importante es que las herramientas fueron realizadas con lenguajes de programación ya deprecados. Por lo que es casi una obligación migrarlos a versiones más actualizadas de dichos lenguajes, para un correcto funcionamiento.

6.1. Conclusiones sobre aspectos técnicos

Si se evalúan uno a uno los objetivos planteados en la sección 1.2 se puede establecer que estos fueron cumplidos parcialmente; esto debido a que se decidió, en conjunto al profesor guía, acotar el alcance de la memoria por temas de tiempo. Para ello se priorizó la reingeniería de la herramienta de Coevaluaciones, por temas de uso y porque ésta ya se encuentra con soporte deprecado, siendo mucho más urgente la necesidad de migrarla a versiones más nuevas de Python y Django..

La implementación de la herramienta de Coevaluaciones fue cumplida en su totalidad. Los principales requisitos propuestos en la sección 4.1, como por ejemplo, la migración a una versión actualizada de Python y Django, el nuevo login a utilizar, la mantención de funcionalidades y la implementación de las nuevas interfaces fueron cumplidos a cabalidad. Estos requisitos son de gran importancia ya que facilitan el uso a

los futuros usuarios de manera sustancial, tal como fue observado en la sección 5 donde se evalúa dicha solución.

Si bien aún faltan detalles por mejorar en la implementación, estos no afectan el uso de la herramienta, pero sí aportan valor extra que puede sumar en lo que respecta a su usabilidad. Por ejemplo, se podría implementar la descargar de los resultados de varias coevaluaciones en conjunto, donde en la actualidad solo se puede descargar de uno en uno.

Otro punto importante es agradecer los consejos y su mentoría del encargado del Área de Desarrollo de Software del DCC. Quién fue el que ayudó con la base inicial para el proyecto, con la creación del Boilerplate DCC, lo cual facilitó la creación de herramientas internas y la escritura de código.

6.2. Conclusiones Personales

En lo que respecta a conclusiones personales a partir del trabajo desarrollado, se considera importante la correcta documentación del proceso. En particular, en algunas partes del proceso se llevó a cabo una bitácora detallada del trabajo realizado, mientras que en otras, la documentación fue casi nula. A la hora de escribir el presente documento, la diferencia entre qué partes fueron apropiadamente documentadas y cuáles no, se hizo evidente dada la mayor dificultad para plasmar en este documento el proceso y las decisiones tomadas.

Otro punto a considerar fue el la distribución y uso del tiempo para realizar el trabajo, esto debido a que ambas herramientas fueron legadas y sus respectivos códigos sufrieron cambios hasta casi mediados de Abril de 2022. Esto se hizo en marco de otro trabajo de título realizado por un compañero del DCC, quien fue la única persona que había tocado el código en los últimos años. Debido a la complejidad de la integración planteada, y a lo tarde que estuvo disponible una versión estable del código fuente de ambas aplicaciones, es que se decidió acotar el alcance del proyecto. De todas maneras, el resultado obtenido es usable y útil, y permitirá a los cuerpos docentes de los cursos mitigar el problema planteado.

6.3. Trabajo Futuro

En lo que respecta al trabajo futuro, existen varias tareas por realizar, en particular la primera debiese ser la reimplementación de las interfaces de usuario y migración planteada para la herramienta de Conformación de Equipos. Esta se debe migrar a versiones de Django y Python compatibles con lo realizado con la herramienta de Coevaluaciones.

Una vez realizada esta reingeniería, es posible seguir con el objetivo principal de este trabajo de título que era la integración completa de ambas herramientas. Tal como se mencionó con anterioridad, el modelo de datos de Conformación de Equipos posee bastantes similitudes con el de Coevaluaciones, hay muchas tablas de la base de datos que se comparte, tales como cursos, alumnos o equipos. Por lo que la integración a través de Django sería bastante sencilla, dado que la información de éstas es obtenida desde el Data Lake de Docencia del DCC.

Otro posible trabajo a futuro es transformar todo este sistema actual basado en monolito de Django, en un sistema de microservicios, donde tanto su frontend como su backend se encuentren separados. Esto facilitaría la escalabilidad de cara a otros proyectos que se quieran integrar a futuro, ya que el backend pasaría a ser una API Rest que provee los datos. La separación en microservicios permitiría la utilización de diferentes librerías para la creación del frontend, como por ejemplo React¹¹ o Vue¹², que son muy usadas en la industria de software en la actualidad.

¹¹ <https://reactjs.org/>

¹² <https://vuejs.org/>

Bibliografía

1. Curso CC5401 - Ingeniería de Software II. Documento Histórico del Proyecto Conformación de Equipos de Desarrollo en Escenarios Académicos. Reporte Técnico. DCC, Universidad de Chile. Semestre Primavera 2017 y Otoño 2018.
2. Curso CC5401 - Ingeniería de Software II. Documento Histórico del Proyecto Co-Evaluaciones de Equipos de Desarrollo de Software. Reporte Técnico. DCC, Universidad de Chile. Semestre Primavera 2017.
3. Silvestre, L. Diseño de equipos de desarrollo de software en escenarios universitarios. Tesis de Magíster en Ciencias de la Computación. DCC, FCFM, Universidad de Chile, 2012.
4. Dzvonyar, D., Alperowitz, L., Henze, D., Bruegge, B. Team composition in software engineering project courses. In *2018 IEEE/ACM International Workshop on Software Engineering Education for Millennials (SEEM'18)* (pp. 16-23). IEEE Press. June 2018.
5. Løvold, H. H., Lindsjörn, Y., Stray, V. Forming and assessing student teams in software engineering courses. In *Proceedings of the International Conference on Agile Software Development* (pp. 298-306). Springer, Cham. June 2020.
6. García-Holgado, A., Vázquez-Ingelmo, A., García-Peñalvo, F. J., González-González, C. S. Perspectiva de género y fomento de la diversidad en la docencia de Ingeniería del Software. *Actas de las Jornadas de la Enseñanza Universitaria de la Informática (JENUi)*, 5, pp. 269-276. 2020.
7. Fernandez-Sanz, L., Misra, S. Analysis of cultural and gender influences on teamwork performance for software requirements analysis in multinational environments. *IET Software*, 6, pp. 167-175. DOI: 10.1049/iet-sen.2011.0070. 2012.
8. Brooke, John. (1995). SUS: A quick and dirty usability scale. *Usability Eval. Ind.*. 189.