



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

REINGENIERÍA DEL SISTEMA DE CONFORMACIÓN DE EQUIPOS

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL EN COMPUTACIÓN

ESTEBAN IGNACIO ZÚÑIGA SALAMANCA

PROFESOR GUÍA:
SERGIO OCHOA DELORENZI

PROFESOR CO-GUÍA:
CRISTÓBAL MESÍAS DURÁN

MIEMBROS DE LA COMISIÓN:
MARÍA CECILIA RIVARA ZÚÑIGA
PABLO GONZÁLEZ JURE

SANTIAGO DE CHILE
2024

Resumen

El trabajo en equipo implica la colaboración de dos o más personas que aportan conocimientos, ideas y habilidades para alcanzar un objetivo común. En el contexto del desarrollo de software, la habilidad para trabajar en equipo es esencial; donde la comunicación efectiva, el respeto y la responsabilidad son claves para lograr el éxito.

En la carrera de Ingeniería Civil en Computación del Departamento de Ciencias de la Computación de la Universidad de Chile hay cursos donde los estudiantes deben trabajar en equipo, y para facilitar la conformación de éstos en los cursos “Ingeniería de Software II” y “Formulación, Evaluación y Gestión de Proyectos”, se desarrolló una herramienta que permite armar equipos balanceados en términos de capacidad técnica y de gestión, y donde sus miembros son psicosocialmente compatibles.

La herramienta de conformación de equipos actual, alojada en un servidor del DCC, enfrenta diversas limitaciones que requieren una reingeniería inmediata. En primer lugar, utiliza una tecnología deprecada, lo que impide su evolución y mantención. Además, la carga manual de datos de integrantes de cursos podría simplificarse integrando una API del DCC. Importantes aspectos, como la perspectiva de género y la consideración de conflictos previos entre integrantes, no se tienen en cuenta en la actual heurística de conformación de equipos.

El proceso de abordar las limitaciones identificadas requería, en primera instancia, familiarizarse con el código y el funcionamiento del sistema legado. Posteriormente, se emprendió la tarea de establecer nuevos requisitos de usuario y diseñar un flujo actualizado para mejorar la eficiencia y funcionalidades del sistema.

La solución comprendió la creación de un nuevo repositorio que permite mejorar la gobernabilidad del sistema, la integración de la API para automatizar procesos manuales, así como la incorporación de la perspectiva de género y la consideración de conflictos previos en la heurística de conformación de equipos. La solución implicó la actualización del modelo de datos, así como de la arquitectura física y lógica del sistema. Además, se implementaron mejoras significativas en los templates para enriquecer la experiencia del usuario.

La evaluación de la solución implementada reveló una alta usabilidad y una percepción positiva de utilidad. Por otra parte, el nuevo algoritmo demostró consistencia en diversidad psicosocial en comparación con el sistema legado.

A Carlos, Lorena, Karla y Lucas

Agradecimientos

Al Esteban de 17 años, gracias por tu admirable madurez al tomar la decisión acertada.

Al Esteban de 23 años, gracias por haber entregado lo mejor a lo largo de estos seis años.

A Dios, mis padres, hermanos y abuelas, por apoyarme en todo momento desde que tomé la difícil decisión de dejar el nido para mudarme a la capital.

A mis amigos, tanto aquellos que han estado conmigo desde el inicio de esta etapa como a quienes se sumaron en el camino. Gracias por ser mi compañía durante estos años.

A mis profesores guía y co-guía, gracias por su importante y constante ayuda para llevar a cabo este trabajo de título.

A ti, compañera, gracias por estar en las buenas y en las malas, y por darme ánimo para sacar adelante este trabajo.

A todas las personas que, de una u otra forma, me han apoyado durante estos seis años, les estoy profundamente agradecido.

Tabla de Contenido

1. Introducción	1
1.1. Situación actual	2
1.2. Objetivo de la memoria	3
1.3. Estrategia de evaluación de la solución	3
2. Marco Teórico	5
2.1. Estrategias de conformación de equipos	5
2.2. Familiarización con el sistema legado	6
2.2.1. Arquitectura del sistema legado	7
2.2.2. Modelo de datos legado	8
2.2.3. Tecnologías utilizadas por el sistema legado	9
2.2.4. Funcionalidad principal del sistema	10
2.3. Interfaces de usuario	11
2.3.1. Interfaces de usuario para el equipo docente	11
2.3.2. Interfaces de usuario para estudiantes	12
2.4. Algoritmo de conformación de equipos	13
3. Concepción de la Solución	14
3.1. Nuevos requisitos de usuario	14
3.1.1. Requisitos del usuario docente	14
3.1.2. Requisitos del usuario estudiante	15
3.2. Flujo del sistema	15

4. Diseño de la Solución	17
4.1. Creación del nuevo repositorio de código refactorizado	17
4.2. Separación de componentes	18
4.3. Integración del Importador de Data de Docencia	19
4.4. Inclusión de la perspectiva de género y conflictos previos	19
4.5. Servicio para descargar resultados de coevaluaciones	20
4.6. Actualización del modelo de datos	21
4.7. Descripción de la nueva heurística de conformación de equipos	22
4.8. Arquitectura física	26
4.9. Arquitectura lógica	27
5. Implementación de la Solución	28
5.1. Interfaces comunes entre los usuarios	29
5.2. Interfaces de usuario docente	30
5.3. Interfaces de usuario estudiante	34
5.4. Resumen de la Solución	37
6. Evaluación de la Solución	38
6.1. Evaluación de la usabilidad percibida	38
6.2. Evaluación de la utilidad percibida	40
6.3. Evaluación de la consistencia en la conformación de los equipos	41
6.4. Resumen del proceso de evaluación	42
7. Conclusiones y Trabajo a Futuro	44
Bibliografía	47
Anexo A: Módulos del cuestionario basado en aptitudes psicosociales	48
Anexo B: Ejemplo de uso de <i>models.TextChoices</i>	53

Anexo C: Enunciados de la encuesta SUS	54
Anexo D: Enunciados de la encuesta TAM	55

Índice de Tablas

4.1. Resultado de la asignación de macrogrupos (ejemplo de conformación de equipos)	24
4.2. Equipos después de la asignación de los primeros cuatro macrogrupos (ejemplo de conformación de equipos)	25
4.3. Equipos después de la asignación de los segundos cuatro macrogrupos (ejemplo de conformación de equipos)	25
4.4. Equipos finales conformados por el Sistema de Conformación de Equipos (ejemplo de conformación de equipos)	26
5.1. Resumen de la solución	37
6.1. Resultados de asignación de macrogrupos (evaluación de consistencia)	42
6.2. Cantidad de estudiantes por macrogrupo en los equipos conformados por el sistema legado (evaluación de consistencia)	42
6.3. Cantidad de estudiantes por macrogrupo en los equipos conformados por el nuevo sistema (evaluación de consistencia)	42

Índice de Ilustraciones

2.1. Arquitectura física del sistema legado	7
2.2. Modelo de datos del sistema legado	9
2.3. Casos de uso del sistema de conformación de equipos	10
2.4. Resumen de cursos de un profesor en los que se ha usado la herramienta . .	11
2.5. Resumen del proceso de conformación de equipos del curso CC4402	12
2.6. Resumen de cursos de un estudiante en los que se ha usado la herramienta .	12
2.7. Resumen de cursos de un estudiante en los que se ha usado la herramienta .	13
3.1. Nuevo flujo del sistema	15
4.1. Nuevo modelo de datos del sistema	22
4.2. Arquitectura física del Sistema de Conformación de Equipos	27
5.1. Vista principal del Sistema de Conformación de Equipos	29
5.2. Vista de los cursos de un usuario	30
5.3. Vista de los cuestionarios de un curso para un usuario docente	31
5.4. Ventana modal de creación de un cuestionario	31
5.5. Cuestionario en modo lectura	32
5.6. Número de estudiantes que aún no han contestado el cuestionario	32
5.7. Extender plazo del período de respuestas de un cuestionario	33
5.8. Vista para la conformación de equipos	33
5.9. Ejemplo de archivo CSV con los equipos conformados	34
5.10. Vista de los cuestionarios de un curso para un usuario estudiante	35

5.11. Formulario de información personal	36
5.12. Vista inicial del cuestionario basado en aptitudes psicosociales	36
5.13. Botones de navegación dentro del cuestionario	37
6.1. Resultados de la evaluación de usabilidad percibida	39
6.2. Resultados de la evaluación de utilidad percibida	40
7.1. Módulo 1 del cuestionario de aptitudes psicosociales	48
7.2. Módulo 2 del cuestionario de aptitudes psicosociales	49
7.3. Módulo 3 del cuestionario de aptitudes psicosociales	49
7.4. Módulo 4 del cuestionario de aptitudes psicosociales	50
7.5. Módulo 5 del cuestionario de aptitudes psicosociales	50
7.6. Módulo 6 del cuestionario de aptitudes psicosociales	51
7.7. Módulo 7 del cuestionario de aptitudes psicosociales	51
7.8. Módulo 8 del cuestionario de aptitudes psicosociales	52
7.9. Módulo 9 del cuestionario de aptitudes psicosociales	52
7.10. Clase “Pregunta” del modelo de datos	53

Capítulo 1

Introducción

El trabajo en equipo es una forma de trabajo donde dos o más personas aportan sus conocimientos, ideas y habilidades para lograr un objetivo en común, por ejemplo, la resolución de un problema, la toma de una decisión o el logro de una meta. En el desarrollo de software, saber trabajar en equipo es vital para el cumplimiento de los proyectos [1]. Algunas de las habilidades necesarias para llevar adelante un buen trabajo en equipo son la comunicación efectiva para decir y escuchar ideas, el respeto para mantener un buen ambiente de trabajo, y la responsabilidad para cumplir con las tareas asignadas, entre muchas otras.

Conformar equipos de trabajo efectivos, donde la cohesión de sus integrantes sea alta, es una tarea complicada, pues usualmente las personas no se conocen entre sí, y cada persona tiene un perfil psicosocial distinto. El perfil psicosocial es un término utilizado para explicar las características psicológicas y sociales de una persona. La conjunción de perfiles psicosociales en un equipo de trabajo incluye fortalezas, debilidades, habilidades sociales, valores aspiraciones y muchas otras cualidades. Este perfil es importante porque influye en cómo cada individuo se relaciona con sus pares y, en el caso específico de un equipo de desarrollo de software, en la comunicación y manejo de conflictos y emociones.

Dentro de la carrera de Ingeniería Civil en Computación, impartida en el Departamento de Ciencias de la Computación (DCC) de la Facultad de Ciencias Físicas y Matemáticas (FCFM) de la Universidad de Chile, un estudiante debe aprobar varios cursos donde debe trabajar en equipo para desarrollar un producto de software. Algunos de estos cursos son: Ingeniería de Software (CC4401), Ingeniería de Software II (CC5401), Formulación, Evaluación y Gestión de Proyectos (CC4402) y Proyecto de Software (CC5402). En estos casos, los estudiantes son agrupados en equipos, y deben desarrollar un software para un cliente real o ficticio, según sea el curso. Por lo tanto, al inicio de cada semestre el equipo docente de cada curso se encarga de formar los equipos de trabajo.

La conformación de equipos en estos cursos siguen criterios distintos, debido a que la formación previa de los estudiantes y las condiciones de trabajo en los proyectos son diferentes [2]. Sin embargo, en los cursos CC4402 y CC5401 el contexto de trabajo es similar, razón por la cual ambos cursos utilizan la misma heurística para conformar los equipos. Éste es particularmente el ámbito en el que se desarrolló el trabajo de título aquí detallado.

Llevar a cabo la conformación de equipos en estos dos cursos puede ser una tarea muy tediosa para el equipo docente. Para facilitar dicha labor, Luis Silvestre [3] desarrolló una herramienta web llamada “Herramienta de Conformación de Equipos” en su tesis de Magíster en Ciencias, Mención Computación. Esta herramienta permite a los estudiantes de un curso responder una encuesta, la cual determina su perfil psicosocial y técnico. Luego, estos resultados son utilizados por un algoritmo que crea automáticamente los distintos equipos de trabajo, buscando formar equipos balanceados en términos de capacidad técnica y de gestión, y donde sus miembros son psicosocialmente compatibles. En el resto de este documento se le llamará “sistema legado” a dicha herramienta.

1.1. Situación actual

La herramienta de conformación de equipos se encuentra alojada en un servidor del DCC, y en producción hasta el día de hoy [4]. Sin embargo, ésta cuenta con diversas limitaciones que hacen urgente realizar una reingeniería de la misma.

- **Limitación 1: Uso de tecnología deprecada.** La aplicación está construida con el framework Django; una tecnología muy utilizada, con buena documentación y simple de implementar. Sin embargo, la versión utilizada en este proyecto es Django 1.11.29, que dejó de tener soporte en abril de 2020. Esto limita su capacidad de evolución, y mantiene abierto los problemas de seguridad de las versiones pasadas de dicho framework. Utilizar una versión actualizada de Django es muy beneficioso, ya que con cada nueva versión se agregan nuevas características y mejoras, además, se arreglan *bugs* y se brinda una mayor seguridad a las aplicaciones. Por lo anterior, ha sido una prioridad en este trabajo de título migrar todo el sistema a una versión actualizada del framework.
- **Limitación 2: No se aprovecha la disponibilidad de datos de cursos y sus integrantes.** En el sistema actual, la información de los integrantes de un curso debe ser subida manualmente por el/la docente al sistema de conformación de equipos; típicamente, a través de un archivo Excel con dicha información. Sin embargo, estos datos se pueden extraer automáticamente desde una API del DCC, evitando así el esfuerzo de carga y todos los errores que pueden ocurrir como producto de la actividad manual. Una API (*Application Programming Interface*) es un fragmento de código que facilita la comunicación entre distintas aplicaciones, y particularmente, el intercambio de información entre ellas. Por lo tanto, esta memoria también busca automatizar la carga y el mantenimiento de información, facilitando el trabajo del docente al momento de armar los equipos.
- **Limitación 3: La actual heurística de conformación de equipos no considera la perspectiva de género.** Según conversaciones con miembros del Área de Género y Diversidad del Centro de Alumnos del DCC (CaDCC), las personas identificadas como mujeres o personas no binarias del DCC, se sienten más cómodas para opinar o mostrar un punto de vista distinto cuando hay otra persona que se identifique como mujer o persona no binaria en el mismo equipo. Debido a esto, se debe realizar una mejora al sistema para lograr que todos los integrantes de un equipo se sientan cómodos y seguros al momento de trabajar con sus pares.

- **Limitación 4: La actual heurística de conformación de equipos no considera los conflictos previos entre algunos de sus integrantes.** Juntar en un mismo equipo a personas que han tenido conflictos en el pasado puede conducir a que no logren trabajar de manera apropiada, dañando la convivencia y comunicación entre sus miembros, contribuyendo a un posible fracaso del proyecto.
- **Limitación 5: Los resultados de las coevaluaciones de un curso son difíciles de recuperar.** Las coevaluaciones, usadas regularmente en los cursos de proyectos, contienen información valiosa para el equipo docente. Sin embargo, actualmente no es posible descargar dichos resultados desde el sistema de conformación de equipos, para luego ser analizados, pues ese servicio no existe en la plataforma. Por lo tanto, se requiere desarrollar un servicio para descargarlos de manera anonimizada, para proteger la identidad de los integrantes del grupo de trabajo.

1.2. Objetivo de la memoria

El objetivo general de este trabajo de título es realizar una reingeniería a la actual herramienta de conformación de equipos utilizada en los cursos CC4402 y CC5401, actualizando la versión del framework Django y todas las dependencias utilizadas en el proyecto. Además, se debe extender su funcionalidad para dar cabida a nuevos requerimientos que han surgido para esta aplicación.

Para dar cumplimiento al objetivo general, se han definido los siguientes objetivos específicos:

1. Refactorizar código en Python de una versión deprecada de Django a una actual.
2. Mejorar el mecanismo de conformación de equipos, para que estos se conformen con perspectiva de género, y considerando los conflictos previos (declarados) entre los miembros del curso.
3. Automatizar la carga de datos de los cursos y sus integrantes.
4. Integrar el Sistema de Coevaluaciones para permitir la descarga de los resultados de éstas, protegiendo la identidad de los estudiantes.

1.3. Estrategia de evaluación de la solución

Se evaluó la usabilidad y utilidad percibida por los usuarios, para la mejora realizada al sistema, utilizando instrumentos conocidos. En el caso de la usabilidad se utilizó la encuesta SUS (*System Usability Scale*) [5] y en el caso de la utilidad se usó una versión reducida de la encuesta definida en TAM (*Technology Acceptance Model*) [6].

La encuesta SUS permite evaluar la eficiencia, eficacia y satisfacción que genera el sistema por medio de 10 preguntas, cuyas respuestas van desde “Totalmente en desacuerdo” a “Totalmente de acuerdo”. Por otra parte, el subconjunto de ítems de la encuesta TAM utilizado en

la evaluación, evalúa sólo la utilidad percibida, pues la evaluación de la usabilidad se realiza con la encuesta SUS.

Para la evaluación, típicamente se toma una muestra de los distintos perfiles de usuario soportados por el sistema, y se les pide realizar algunas actividades comunes en la aplicación. Luego, se realiza la encuesta de usabilidad y utilidad antes mencionadas. Según los valores obtenidos, se determina el nivel de usabilidad y utilidad de la solución. Estos valores están separados en rangos para cada uno de estos atributos de calidad de la solución.

Capítulo 2

Marco Teórico

En este capítulo se presentan otras soluciones para apoyar la tarea de conformar equipos en otros cursos del DCC. También se realiza un análisis detallado de la estructura y funcionalidad del sistema legado.

2.1. Estrategias de conformación de equipos

La conformación de equipos de trabajo es un proceso que puede llevarse a cabo mediante diversas modalidades. Dentro del ámbito universitario, cada curso presenta su propia forma de conformar equipos; algunos docentes optan por crearlos siguiendo criterios predefinidos, mientras que otros permiten a los estudiantes organizar sus propios equipos.

Cuando los docentes arman los equipos de trabajo, pueden hacerlo de forma totalmente aleatoria o conformarlos según un criterio simple, como el agrupamiento por género u orden alfabético. También se pueden usar criterios más complejos como, por ejemplo, los roles, disponibilidad horaria, características personales, etc. Por el contrario, cuando estos equipos son armados por los propios estudiantes, se busca lograr confianza y comodidad entre los miembros.

Una investigación realizada por C. A. Wellington, T. Briggs y C. D. Girard demuestra que tener equipos con personalidades diversas es un factor relevante para que un equipo logre con éxito un objetivo [7]. Esta hipótesis también es apoyada por los resultados de un estudio realizado por A. Law y R. Charron, donde se concluye que para conformar un equipo con alta cohesión, se deben considerar las características psicosociales de los individuos [8]. A partir de estas dos conclusiones, se logra dimensionar la importancia de considerar estas características psicológicas y sociales, para poder formar equipos cohesivos.

En el ámbito de la conformación de equipos en cursos del DCC, se han realizado algunos proyectos. Por ejemplo, en el curso “Proyecto de Software” (CC5402), que es parte de la malla curricular de Ingeniería Civil en Computación en la Facultad de Ciencias Físicas y Matemáticas de la Universidad de Chile, se diseñó un sistema de conformación de equipos en [9]. Dicho sistema implementa un algoritmo existente [10] que arma equipos teniendo en

cuenta que sus integrantes compartan bloques de horario disponibles. De esa manera, ningún integrante deberá trabajar solo en un cierto período. Para establecer la disponibilidad horaria de los estudiantes, se les solicita indicar su disponibilidad de lunes a viernes en bloques de 4 horas.

El algoritmo de conformación de equipos comienza seleccionando aleatoriamente una cantidad de personas determinadas por el equipo docente, el cual depende de la cantidad de proyectos disponibles para un semestre. Del grupo seleccionado, se escoge de forma aleatoria un estudiante, al cual se le llama pivote. Luego, se compara la disponibilidad horaria del pivote con cada uno de los demás estudiantes del grupo seleccionado. Si ambos tienen disponible el mismo bloque horario, se registra la combinación. En caso de no compartir disponibilidad horaria, se ingresa al estudiante en un grupo de intentos fallidos, de donde se toma a otro estudiante para comparar disponibilidades con el pivote. Este proceso se repite hasta conseguir armar todos los equipos.

Después que el sistema arma los equipos, es necesario calificar el resultado de la conformación. Para esto se generó un algoritmo de armonía local, que mide qué tanto los equipos armados se acercan al óptimo; es decir, a la situación en que todos los miembros de un equipo puedan trabajar en simultáneo.

Para calcular la armonía local de un equipo se aplica una fórmula matemática que considera dos factores: la capacidad de los alumnos para trabajar juntos en un mismo bloque horario, y la cantidad de interacciones posibles entre ellos a lo largo de la semana.

Después de calcular la armonía local de cada equipo, se ejecuta un algoritmo de armonía global, el cual pretende lograr que los equipos conformados por el sistema sean lo más balanceados posible. Es decir, se quiere minimizar las diferencias entre los valores de armonía local de cada equipo, para lograr que todos los equipos tengan un valor similar. De esa manera, se hace una conformación de equipos más justa.

El algoritmo de armonía global toma los equipos con mayor y menor armonía local. Luego intercambia al alumno con mayor cantidad de bloques disponibles del equipo con mayor armonía local con el alumno que indicó menos bloques disponibles del equipo con menor armonía local. Esto afecta directamente el valor de la armonía local de ambos equipos. Este algoritmo se ejecuta hasta que se alcance una desviación estándar mínima entre los valores de armonía local de los equipos.

Si bien este algoritmo ha demostrado ser un aporte para armar equipos en el curso CC5402, no es aplicable a la realidad de los cursos CC5401 y CC4402, puesto que estos usan criterios de agrupación diferentes.

2.2. Familiarización con el sistema legado

Para iniciar el trabajo de título, resultaba fundamental adquirir un conocimiento total del sistema legado. Esto permitiría llevar a cabo las modificaciones de código de manera más eficiente, y sin generar repercusiones negativas en otros componentes que utilicen dicho

código.

En primer lugar, se obtuvieron los permisos necesarios para acceder al repositorio en GitHub de la “Herramienta de conformación de equipos”. Una vez que se obtuvo el acceso, se clonó el repositorio de forma local y se procedió a revisar minuciosamente cada uno de los archivos existentes, analizando desde el modelo de datos hasta las interfaces de usuarios del sistema. No obstante, aún quedaba por conocer cómo funcionaba el sistema en su conjunto.

Con este fin, se realizó una solicitud al Área de Sistemas del DCC para obtener un ambiente de prueba, que permitió identificar los diversos flujos de la herramienta y comprender, en su mayoría, las funcionalidades que ésta ofrece. Las últimas modificaciones al sistema legado habían sido realizadas por equipos del curso “Ingeniería de Software II” durante los años 2017 y 2018 [11].

2.2.1. Arquitectura del sistema legado

La arquitectura física del sistema legado está compuesta por un servidor web de Django, que posee una base de datos PostgreSQL donde se almacenan los datos de los cursos y sus integrantes, las respuestas de los cuestionarios respondidos por estudiantes y los equipos conformados.

Para acceder al sistema, el usuario debe autenticarse en la plataforma utilizando la API de U-Pasaporte, lo que le permite ingresar en función del perfil de usuario correspondiente. La Figura 2.1 muestra la arquitectura física del sistema legado. Ésta se obtuvo del documento histórico del proyecto [11].

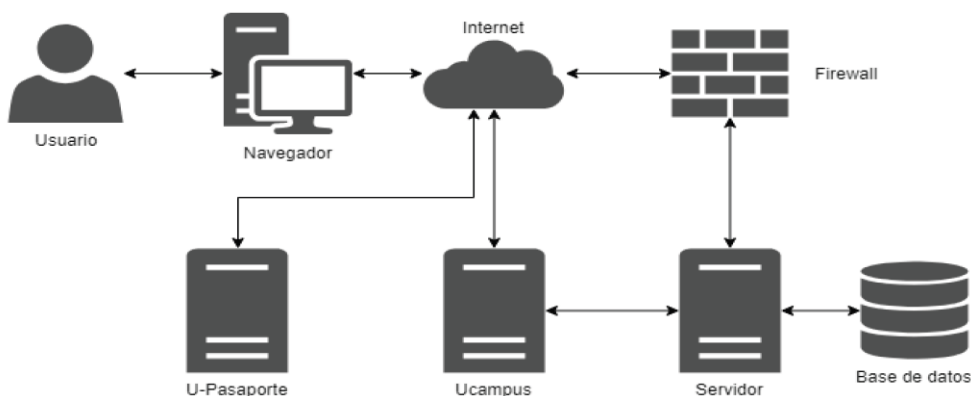


Figura 2.1: Arquitectura física del sistema legado

La arquitectura lógica del sistema legado consta de tres capas: capa de presentación, capa de negocios y capa de datos.

- La capa de presentación corresponde a las vistas de la plataforma, es la capa que permite la interacción de los usuarios con el sistema.

- La capa de datos almacena y entrega los datos necesarios para llevar a cabo las acciones requeridas por el usuario.
- La capa de negocios es la que recibe las acciones y peticiones del usuario realizadas en la capa de presentación. Además, solicita a la capa de datos que entregue la información necesaria para enviar la respuesta al usuario.

2.2.2. Modelo de datos legado

Las principales entidades del modelo de datos de la herramienta de conformación de equipos son las siguientes:

- *CourseUser*: Modela a un usuario del sistema y permite diferenciar entre dos perfiles: administrador (que pertenece al equipo docente de un curso) y estudiante.
- *Question*: Corresponde a una pregunta de un cuestionario.
- *Answer*: Representa una posible respuesta a una instancia del modelo *Question*.
- *Test*: Modela un cuestionario, el cual está compuesto de varias instancias del modelo *Question*.
- *Team*: Corresponde a un equipo formado por el sistema. Éste se compone de cierta cantidad de instancias del modelo *CourseUser*.
- *Course*: Modela un curso del DCC, donde se utiliza la herramienta de conformación de equipos.

El modelo de datos se completa con distintas relaciones entre las entidades ya mencionadas, que permiten el correcto funcionamiento del sistema. La relación entre los modelos del sistema se muestra en la Figura 2.2.

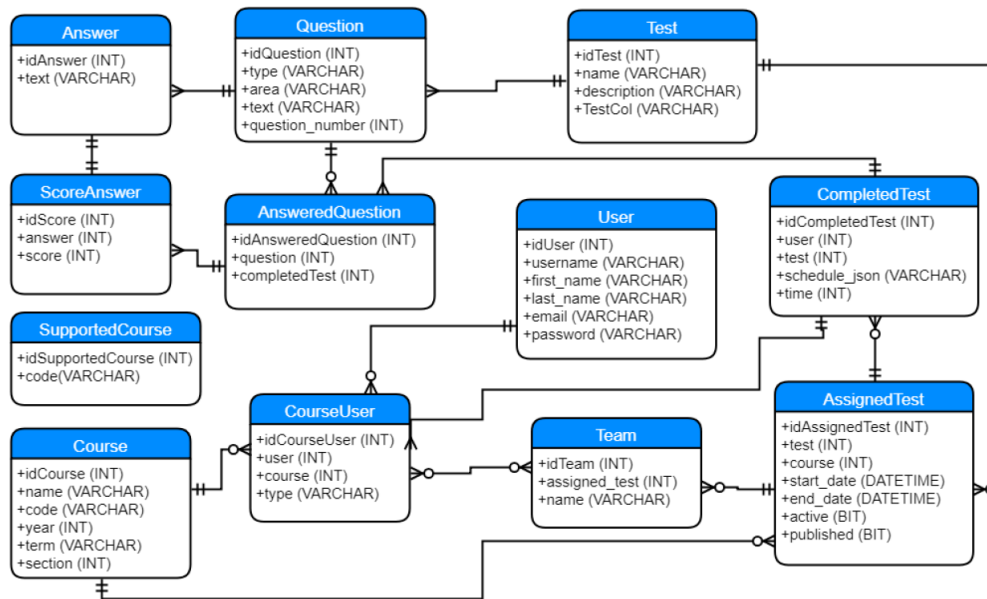


Figura 2.2: Modelo de datos del sistema legado

Cabe destacar que en el sistema legado hay dos tipos de usuarios, a continuación se detallan las funciones de cada uno.

- **Administrador:** Corresponde a un miembro del equipo docente, usualmente el profesor o profesora de cátedra. Éste es el encargado de cargar los estudiantes al sistema, de confirmar o reorganizar los equipos que la aplicación genera, y de exportar los resultados en un archivo CSV para posteriormente publicarlos.
- **Alumno:** Corresponde a un estudiante de un curso de los soportados por el sistema. Éste ingresa a la plataforma utilizando las credenciales de U-Pasaporte, y responde el cuestionario que está disponible en la plataforma.

2.2.3. Tecnologías utilizadas por el sistema legado

En cuanto al lado más técnico del sistema, se tienen dos partes esenciales, el backend y el frontend. Por el lado del backend, la herramienta está desarrollada con el framework Django en su versión 1.11.29 y se ejecuta en un entorno con Python 3.4. Por otra parte, el frontend utiliza templates de Django y el framework Vue.js en su versión 2.4; además, utiliza librerías como *JQuery* y *Bootstrap*. Es importante destacar que las versiones de los frameworks se encuentran sin soporte, es decir, deprecadas, debido a la evolución que éstas han tenido en los últimos años.

2.2.4. Funcionalidad principal del sistema

En la Figura 2.3, obtenida del documento histórico del proyecto [11], se muestran los casos de uso y actores involucrados en la interacción con el sistema de conformación de equipos.

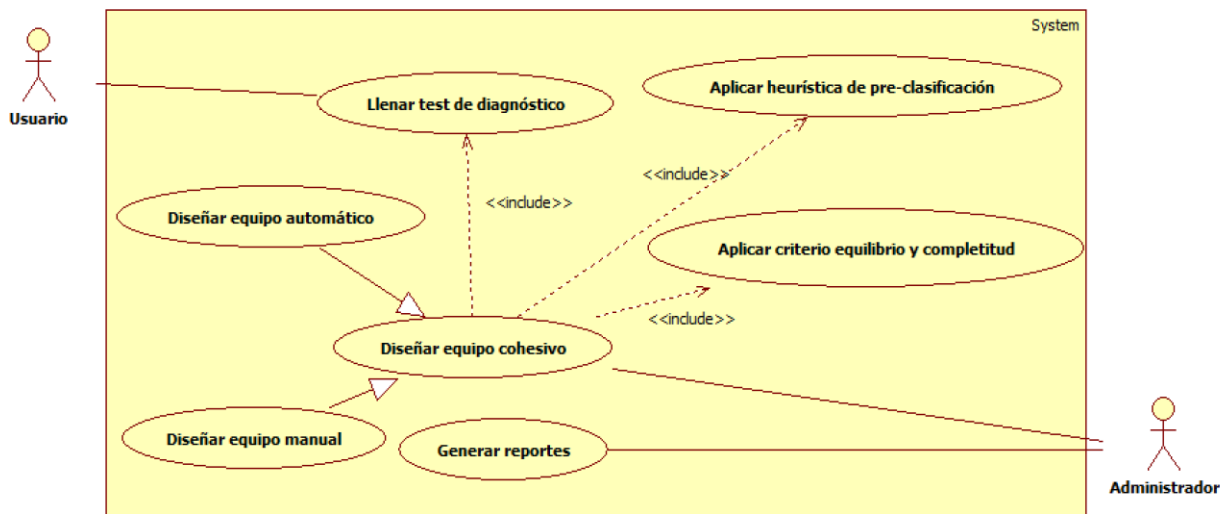


Figura 2.3: Casos de uso del sistema de conformación de equipos

El flujo del proceso tiene tres etapas principales: 1) la carga de estudiantes y configuración de equipos, 2) la respuesta de los estudiantes al cuestionario basado en aptitudes psicosociales, y 3) la conformación de los equipos y posterior notificación a los estudiantes. A continuación se explica cada una de las etapas.

1. **La carga de estudiantes y configuración de equipos:** Esta etapa es ejecutada en su totalidad por el equipo docente, es decir, por usuarios administradores. En primer lugar, se debe cargar al sistema la planilla con los integrantes del curso en cuestión. Luego se debe personalizar la configuración con la cual se conformarán los equipos, esto se hace creando un cuestionario con fecha de inicio y término que los estudiantes del curso deberán responder dentro de ese período.
2. **La respuesta de los estudiantes al cuestionario basado en aptitudes psicosociales:** En la segunda etapa, los estudiantes de cada curso responden el cuestionario de forma individual, y el sistema registra los resultados de cada encuesta respondida. El diseño de este cuestionario fue reportado en la tesis de Luis Silvestre [3]. Los módulos que componen el cuestionario se pueden ver en el Anexo A.
3. **La conformación de los equipos y la notificación a los estudiantes:** El flujo concluye con la conformación de los equipos acorde a los resultados de la encuesta psicosocial. Una vez que el software ha creado los equipos de trabajo, el equipo docente puede modificarlos, y posteriormente notificar a los estudiantes acerca de la composición de sus respectivos equipos.

2.3. Interfaces de usuario

2.3.1. Interfaces de usuario para el equipo docente

Para el caso del equipo docente a cargo de un curso, el sistema ordena por año los cursos en los que se utiliza la herramienta para conformar los equipos de trabajo. En la Figura 2.4 se ven los cursos dictados por un académico, y se observa que, por ejemplo, en el semestre otoño 2023 el usuario conectado (profesor de cátedra) realizó la conformación de equipos en los cursos CC4402 y CC5401 utilizando esta herramienta.

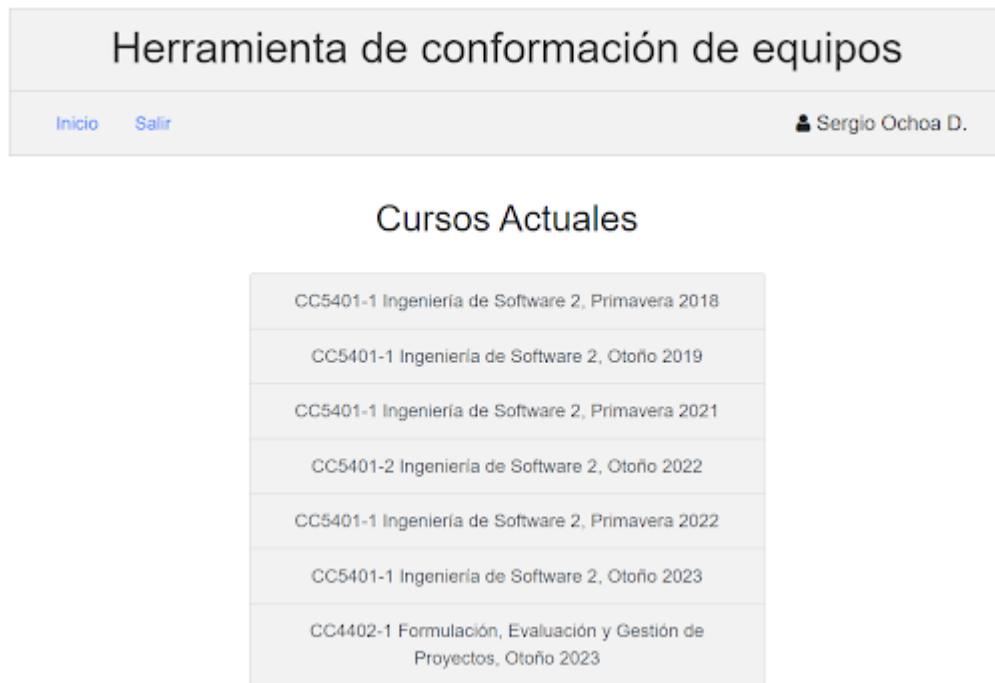


Figura 2.4: Resumen de cursos de un profesor en los que se ha usado la herramienta

Cuando se escoge un curso en particular, por ejemplo el CC4402, el académico a cargo puede ver cuántas conformaciones de equipo se realizaron ese semestre (podría ser más de una), en qué fechas, y cuáles fueron los equipos resultantes. La Figura 2.5 muestra un resumen de la información disponible para cada instancia de conformación de equipos para ese curso, año y semestre.



Figura 2.5: Resumen del proceso de conformación de equipos del curso CC4402

2.3.2. Interfaces de usuario para estudiantes

Por otra parte, los estudiantes que ingresan a la plataforma pueden ver todos los cursos que han cursado o están cursando, en los cuales se utiliza el sistema. En el caso mostrado en la Figura 2.6, se ven los cursos inscritos por un estudiante y se observa que en el semestre de otoño 2022 cursó “CC5401 - Ingeniería de Software II”, y los equipos de dicho semestre fueron conformados usando el sistema.

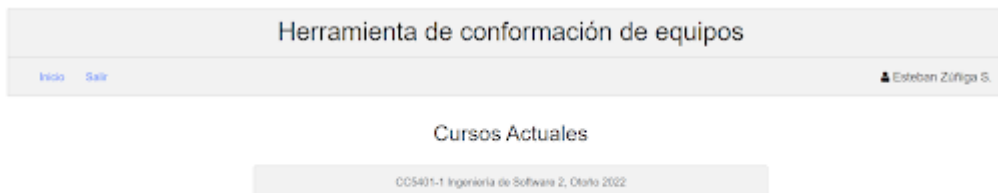


Figura 2.6: Resumen de cursos de un estudiante en los que se ha usado la herramienta

Al hacer click en el curso, se muestran los cuestionarios asignados. En este caso, el estudiante debía responder el cuestionario basado en aptitudes psicosociales para el curso Ingeniería de Software II (Figura 2.7).



Figura 2.7: Resumen de cursos de un estudiante en los que se ha usado la herramienta

2.4. Algoritmo de conformación de equipos

El algoritmo de conformación de equipos se diseñó con el objetivo de agrupar a estudiantes de manera efectiva y variada, teniendo en cuenta su perfil psicosocial. A continuación, se presenta una explicación detallada de cada paso del proceso.

- Paso 1: Recolección de respuestas. El algoritmo inicia con la recolección de respuestas de los estudiantes al cuestionario de aptitudes psicosociales. Este cuestionario consta con varias preguntas, y cada una de ellas pertenece a un área. Por ejemplo, las preguntas relacionadas con habilidades técnicas contribuyen al área técnica, mientras que las preguntas sobre habilidades interpersonales influyen en el área de habilidades sociales. Las áreas de interés en este cuestionario son: habilidades técnicas y sociales, liderazgo, personalidad, trabajo en equipo, comunicación, resolución de conflictos, capacidad de gestión y planificación. El cuestionario concluye con el Test de Belbin [12], el cual se utiliza para identificar y analizar los roles naturales de los individuos en situaciones de trabajo en equipo.
- Paso 2: Cómputo de puntajes por área. Después de recopilar las respuestas, el sistema procede a computar los resultados por área. Los resultados obtenidos permiten asignar un puntaje a cada estudiante en relación con su desempeño específico en cada área evaluada. Este puntaje refleja la afinidad del estudiante en cada área específica, proporcionando una visión detallada de sus habilidades y preferencias.
- Paso 3: Asignación a macrogrupos. Luego, se procede a dividir a los estudiantes en cuatro macrogrupos según los puntajes calculados. Cada macrogrupo agrupa a estudiantes con perfiles psicosociales similares, de acuerdo a sus respuestas.
- Paso 4: Conformación de equipos. Finalmente, se procede a la creación de equipos mediante la iteración por los macrogrupos. Se forma cada equipo incorporando a un estudiante de cada macrogrupo. Esta estrategia busca diversificar los equipos, intentando que cada uno cuente con una combinación equilibrada de habilidades y condiciones.

Capítulo 3

Concepción de la Solución

En esta sección se aborda la planificación de la nueva solución, introduciendo nuevos requisitos de usuario que buscan optimizar la experiencia, tanto para docentes como para estudiantes. Además, se presenta el nuevo flujo del sistema a través de un diagrama que da una visión global del proceso de conformación de equipos.

3.1. Nuevos requisitos de usuario

A continuación se listan los nuevos requisitos de usuario a tener en cuenta para la implementación de la nueva solución.

3.1.1. Requisitos del usuario docente

- **Agregar cursos automáticamente**

Un usuario que ingrese como docente podrá ver sus cursos actuales, sin necesidad de agregarlos por la vista de administrador de Django.

- **Agregar integrantes automáticamente**

Un usuario que ingrese como docente podrá ver los integrantes de sus cursos, sin necesidad de cargarlos manualmente a la plataforma.

- **Ver el cuestionario que los estudiantes deben responder**

Un usuario que ingrese como docente podrá ver, en modo lectura, el cuestionario que los estudiantes del curso están respondiendo.

- **Conformar equipos con perspectiva de género**

Un usuario que ingrese como docente podrá conformar equipos que cumplan con los nuevos lineamientos planteados respecto a una perspectiva de género.

- **Conformar equipos evitando juntar estudiantes con conflictos previos**

Un usuario que ingrese como docente podrá conformar equipos donde no existan conflictos previos entre los integrantes.

- **Enviar un correo automáticamente con los equipos generados**

Un usuario que ingrese como docente podrá publicar los equipos generados por el sistema enviando un correo a cada estudiante informando su equipo.

3.1.2. Requisitos del usuario estudiante

- **Responder formulario con información personal**

Un usuario que ingrese como estudiante podrá llenar un formulario indicando su correo electrónico, género y conflictos con otros estudiantes.

- **Editar o ver respuestas**

Un usuario que ingrese como estudiante podrá editar sus respuestas, siempre que no haya vencido el plazo de respuesta del cuestionario. Además, este usuario podrá ver, en modo lectura, sus respuestas si dicho plazo ya se cumplió.

3.2. Flujo del sistema

Luego de comprender el flujo de la herramienta de conformación de equipos original, fue necesario crear un nuevo diagrama de flujo incluyendo las modificaciones planificadas. En la Figura 3.1 se muestra el nuevo flujo del sistema.

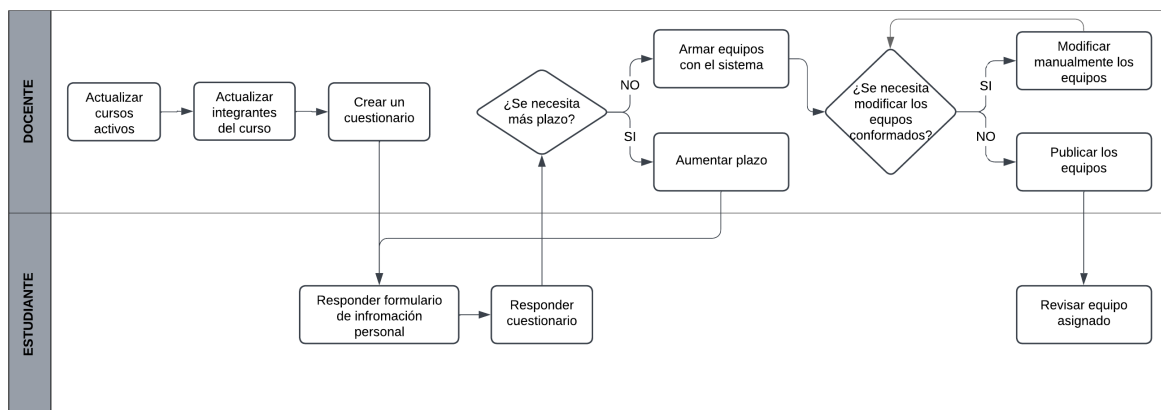


Figura 3.1: Nuevo flujo del sistema

El flujo comienza cuando un usuario docente ingresa al sistema para comenzar la conformación de equipos de un curso. Para esto, lo primero que debe realizar es actualizar los cursos presionando un botón, acción que cargará en la plataforma los nuevos cursos del semestre actual. Una vez seleccionado un curso, el usuario docente deberá presionar nuevamente un botón que cargará automáticamente los integrantes de dicho curso. Esto permitirá a los estudiantes tener acceso al sistema y ver los procesos de conformación de equipos activos. Actualizados los integrantes de un curso, se procederá a crear un nuevo cuestionario, el cual deberá ser respondido por los estudiantes dentro de un plazo específico.

El cuestionario comenzará con un formulario donde cada estudiante deberá responder tres secciones: la primera corresponde al correo electrónico donde desea recibir los resultados de la conformación de equipos, la segunda al género con el que se identifica, y por último deberá seleccionar (opcionalmente) un máximo de dos estudiantes con los que tiene un conflicto previo y desean no tener en su equipo.

Cuando el plazo termina, el usuario docente puede decidir extender el plazo de respuestas al cuestionario en caso de ser necesario. Si no es necesario, entonces se procede a la creación de los equipos. Para esto se requiere especificar la cantidad de equipos que se desea generar y presionar el botón para generar los equipos.

Con los equipos creados, el usuario docente puede modificar a su criterio los equipos conformados por el sistema hasta tener la conformación final, la cual deberá guardar, y publicar en caso de ser los equipos definitivos. Una vez publicados los equipos, se enviará un correo automático a cada estudiante con la conformación de su equipo.

Capítulo 4

Diseño de la Solución

En esta sección se abordan los aspectos más importantes para la modernización y mejora del sistema existente. Se menciona cómo se logró implementar una infraestructura más actualizada y compatible con versiones recientes de Django. Además, se explora la integración de herramientas clave para automatizar la carga de información de cursos y participantes. También se introducen cambios en el modelo de datos, así como nuevas heurísticas para la conformación de equipos, considerando aspectos de género y conflictos previos entre los potenciales compañeros de equipo. Por último, se describe la arquitectura física y lógica del sistema, resaltando la adopción del patrón MVT (Model-View-Template) en Django, como una mejora estructural fundamental.

4.1. Creación del nuevo repositorio de código refactorizado

Tal como se mencionó anteriormente, la versión de Django con la que está implementado el sistema corresponde a la versión 1.11.29, que no tiene soporte desde el año 2020. Es por esto que era necesario actualizar la aplicación a una versión *Long-Term Support* (LTS), es decir, una versión con soporte por los próximos años (versión mayor a la 4.2).

Después de analizar el código y estructura del sistema legado, se tomó la decisión, en colaboración con el encargado del Área de Desarrollo del DCC, de crear un nuevo repositorio basado en un *boilerplate* de Django desarrollado por su área, llamado “django_utils” [13]. Esta decisión se basó en el hecho de que sería extremadamente complicado refactorizar el código en el repositorio existente, debido a la desactualización tan abultada de versiones de Django. Además, se recomendó esta opción para mejorar la gobernabilidad y facilitar la posterior mantención del sistema, ya que la mayoría de las nuevas aplicaciones desarrolladas por el DCC se basan en este repositorio. El boilerplate “django_utils” es un conjunto de aplicaciones para Django desarrollado por el Área de Desarrollo del DCC. Este repositorio está construido utilizando el framework Django en su versión 4.2, ejecutado en un ambiente de Docker con Python 3.8, el cual entrega funcionalidades comunes para los distintos productos de software basados en dicho framework, y que son desarrollados en el departamento. En este

trabajo de título se utilizaron dos aplicaciones entregadas por esta herramienta:

1. *Single Sign-On (SSO)*: Es un conjunto de tecnologías que permiten a un usuario autenticarse una vez en un sistema, y luego acceder a múltiples aplicaciones o servicios sin tener que volver a autenticarse cada vez. En este caso, el SSO permite ingresar a las distintas aplicaciones del Departamento con las credenciales institucionales.
2. *Users*: Esta aplicación provee un usuario personalizado, llamado *CustomUser*, que otorga al sistema variada información sobre un integrante del Departamento. La información entregada incluye nombres de la persona, su correo electrónico, y roles que tiene dentro del departamento, entre otros.

Adicionalmente, el boilerplate cuenta con correctores de código que promueven el uso de buenas prácticas, facilitando aún más el proceso de implementación.

Una vez creado y configurado el nuevo repositorio, se inició el proceso de migración, refactorización y optimización del código fuente. Estos tres procesos fueron realizados de forma simultánea, es decir, a medida que se hacían migraciones de código de un repositorio a otro, éste se refactorizaba con funcionalidades de Django más recientes y se optimizaba en los casos que se pudiera. Algunos bloques de código estaban escritos en inglés y otros en español. Debido a que otras aplicaciones desarrolladas por el DCC tienen su código en español, se decidió unificar todo el código a este idioma.

4.2. Separación de componentes

Django es un framework que sigue el patrón de diseño Modelo-Vista-Controlador (MVC), específicamente su variante Modelo-Vista-Template (MVT) debido a que separa el código en tres capas: modelos, vistas y templates. A continuación se detalla brevemente cada una de las capas.

- **Modelos**: Son los responsables de definir la estructura que tendrá la base de datos del proyecto y cómo interactúa con el resto del sistema. Cada modelo representa una entidad o una relación en la base de datos.
- **Vistas**: Manejan la interacción entre el usuario y la base de datos.
- **Templates**: Corresponden a las distintas interfaces que utiliza el usuario. Permiten la inclusión de variables entregadas por las vistas en el código HTML que es enviado al navegador.

Django también presenta muchas características que facilitan el desarrollo web, una de ellas es la modularización del código en aplicaciones. Una aplicación en Django es un módulo independiente que encapsula una funcionalidad específica dentro de un proyecto de desarrollo web. Separar el proyecto de Django en aplicaciones entrega beneficios, entre los cuales se encuentran:

1. Reutilización de código: Se pueden crear aplicaciones con funcionalidades específicas que pueden utilizarse en otros proyectos.
2. Mayor mantenibilidad: Facilita el mantenimiento del código, ya que cada aplicación tiene una lógica propia de modelos, vistas y plantillas.
3. Desacoplamiento entre componentes: Mejora la cohesión del código a través de los distintos componentes, y permite identificar errores fácilmente.

El repositorio original no tenía una estructura separada en aplicaciones, por ende, todos los modelos, vistas y templates estaban en un solo directorio con archivos de código extensos. Por este motivo, se decidió organizar el nuevo repositorio en aplicaciones, teniendo una aplicación para los cursos, otra para los integrantes, una tercera para todo lo relacionado a los cuestionarios y, por último, una aplicación para los equipos. Cada una de estas aplicaciones tiene sus respectivos modelos, vistas y templates.

4.3. Integración del Importador de Data de Docencia

Para la automatización de la carga de datos de los cursos y sus integrantes, se utilizó el “Importador de Data de Docencia” [14]. Esta herramienta es otro desarrollo llevado a cabo por el Área de Desarrollo del DCC, el cual provee una API. La información extraída a través de la API corresponde principalmente a los cursos de los ramos soportados por el sistema, es decir, las distintas secciones de los cursos CC4402 y CC5401. Dentro de esta información se encuentra la lista de integrantes de dichos cursos, con su información personal y de contacto.

Para conectarse a la API es necesario estar conectado a una red de internet de la Universidad de Chile o a la VPN corporativa de esta casa de estudios, y tener un *token* de acceso proporcionado por el encargado del Área de Desarrollo del DCC. De esta forma, ya no se depende de que el equipo docente cargue correctamente esos datos, sino que ahora estos son recuperados por un servicio del sistema.

4.4. Inclusión de la perspectiva de género y conflictos previos

En la búsqueda de soluciones para crear equipos con una perspectiva de género inclusiva, se tomaron en cuenta las conclusiones reportadas por Bastarrica y Simmonds [15], en un estudio realizado en el curso CC5402 (Proyecto de software). Dicha investigación sugiere que al formar un equipo de trabajo, es preferible que éste tenga al menos 2 mujeres, o bien ninguna.

Tras analizar este tema con el Área de Género y Diversidad del Centro de Alumnos del DCC (CaDCC), se llegó a la conclusión de que esta idea también debe extenderse a personas que no se identifican como hombre ni mujer. En consecuencia, en este trabajo de título se

propuso que en cada equipo en el que haya una persona que se identifique como mujer o como persona no binaria, deberían haber al menos dos miembros que se identifiquen de esta manera.

Por otra parte, reportar los conflictos previos permite mejorar la calidad y comodidad del equipo a lo largo del semestre. Según De Wit, Jehn y Scheepers [16] la presencia de conflictos interpersonales afecta la motivación y la capacidad de los miembros de un grupo para abordar puntos de vista divergentes relacionados con una tarea; además, provoca rigidez y procesamiento sesgado de información. Para evitar estos problemas es que se planteó la idea de permitir a los estudiantes indicar, opcionalmente, dos estudiantes del curso con los que han tenido un conflicto previo a la realización del curso, y desearan no compartir equipo con tales personas.

Para registrar el género de los estudiantes y los conflictos previos que tengan con otros, se creó un formulario web que el estudiante debe completar antes de comenzar a responder el cuestionario basado en aptitudes psicosociales. La implementación del formulario web se llevó a cabo en dos fases. En la primera se diseñó la interfaz, y luego se validó con potenciales usuarios. En la segunda se acogió el feedback obtenido, y se comenzó con el desarrollo de la funcionalidad y la integración con el resto del sistema.

4.5. Servicio para descargar resultados de coevaluaciones

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 coevaluación entre pares. Para esto existe una herramienta desarrollada en el DCC llamada “Herramienta de Coevaluación de Equipos” [17]. Esta herramienta permite a cada integrante del equipo, evaluar el desempeño de cada uno 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.

Un objetivo específico de este trabajo de título aborda la oportunidad de integrar este servicio de coevaluaciones utilizado en los distintos cursos donde existen equipos de trabajo. Aprovechar esta oportunidad implica que un usuario docente debe poder descargar los resultados de las coevaluaciones de un curso, y utilizarlos con fines científicos o para la evaluación y mejora de las distintas técnicas de conformación de equipos. Esta descarga de información no debe significar una vulneración a la privacidad de los estudiantes.

Durante la realización de este trabajo de título, se exploraron distintos escenarios y posibilidades para agregar un servicio que permita comunicarse con el sistema de coevaluaciones. A pesar de la relevancia de esta funcionalidad, la complejidad de su integración y las limitaciones de tiempo disponible para la conclusión del trabajo de título impidieron llevar a cabo una implementación. Como resultado, se tomó la decisión de no abordar este objetivo específico en la presente memoria, principalmente por razones de tiempo y complejidad de la

labor requerida.

4.6. Actualización del modelo de datos

La primera modificación al modelo de datos del sistema legado fue el ajuste de los nombres de los modelos y sus respectivos atributos, debido al mencionado cambio de idioma en el código de la aplicación.

Una mejora al modelo de datos se obtuvo gracias a la actualización de la versión de Django, puesto que en la versión utilizada en el Django boilerplate se incluyen los *TextChoices*, que permiten declarar en la misma clase del modelo distintas opciones que puede tomar un atributo, sin necesidad de utilizar malas prácticas. Por ejemplo, en el modelo “Pregunta”, cada pregunta corresponde a sólo un área. Es por esto que se define una clase “Area”, que se extiende de la clase *TextChoices*, y disponibiliza todas las áreas posibles que puede tener una pregunta. Así solo basta asignarle una de estas opciones a la instancia de “Pregunta”. Esto hace al código más extensible, puesto que, de aparecer una nueva área, solo sería necesario agregarla a la clase “Area” (ver Anexo B).

Para la automatización de la carga de estudiantes al sistema fue necesario crear una nueva entidad llamada “Ramo”. Esta nueva entidad permite guardar en la base de datos los ramos soportados por el sistema; es decir, los ramos “CC4402 - Formulación, Evaluación y Gestión de Proyectos” y “CC5401 - Ingeniería de Software II”. Un atributo relevante en esta nueva entidad es el `id_api`, que corresponde al identificador único del ramo en la API del “Importador de Data de Docencia”, de esta forma es posible actualizar los cursos de estos ramos en cada nuevo semestre.

Para integrar las nuevas funcionalidades, es decir, la inclusión de perspectiva de género y un registro de conflictos, fue necesario modificar profundamente la entidad “Integrante”, que modela a un integrante del curso. En resumen, los nuevos atributos de esta clase son:

- usuario: Una llave foránea a *CustomUser* provisto por el “django_utils”. Este guarda el nombre del usuario e información personal.
- rut: Corresponde al rut del usuario y permite buscar al usuario en la API provista por el Importador de Data de Docencia.
- curso: Llave foránea al curso del individuo.
- genero: Corresponde al género del individuo, solo se define para los estudiantes y permite hacer una conformación de equipos con perspectiva de género.
- email: El correo electrónico del estudiante donde se le enviarán los resultados de la conformación de equipos.
- tipo: Corresponde al tipo de usuario, es decir, docente o estudiante.
- informacion_personal_respondida: Es un booleano que muestra si un estudiante respondió el formulario de información personal, donde indica su correo electrónico, género y estudiantes con conflicto.

- integrantes_conflicto: Llave foránea a la misma clase Integrante, y corresponde a los otros estudiantes con los cuales un individuo tiene conflictos.

El modelo de datos definitivo se muestra en la Figura 4.1.

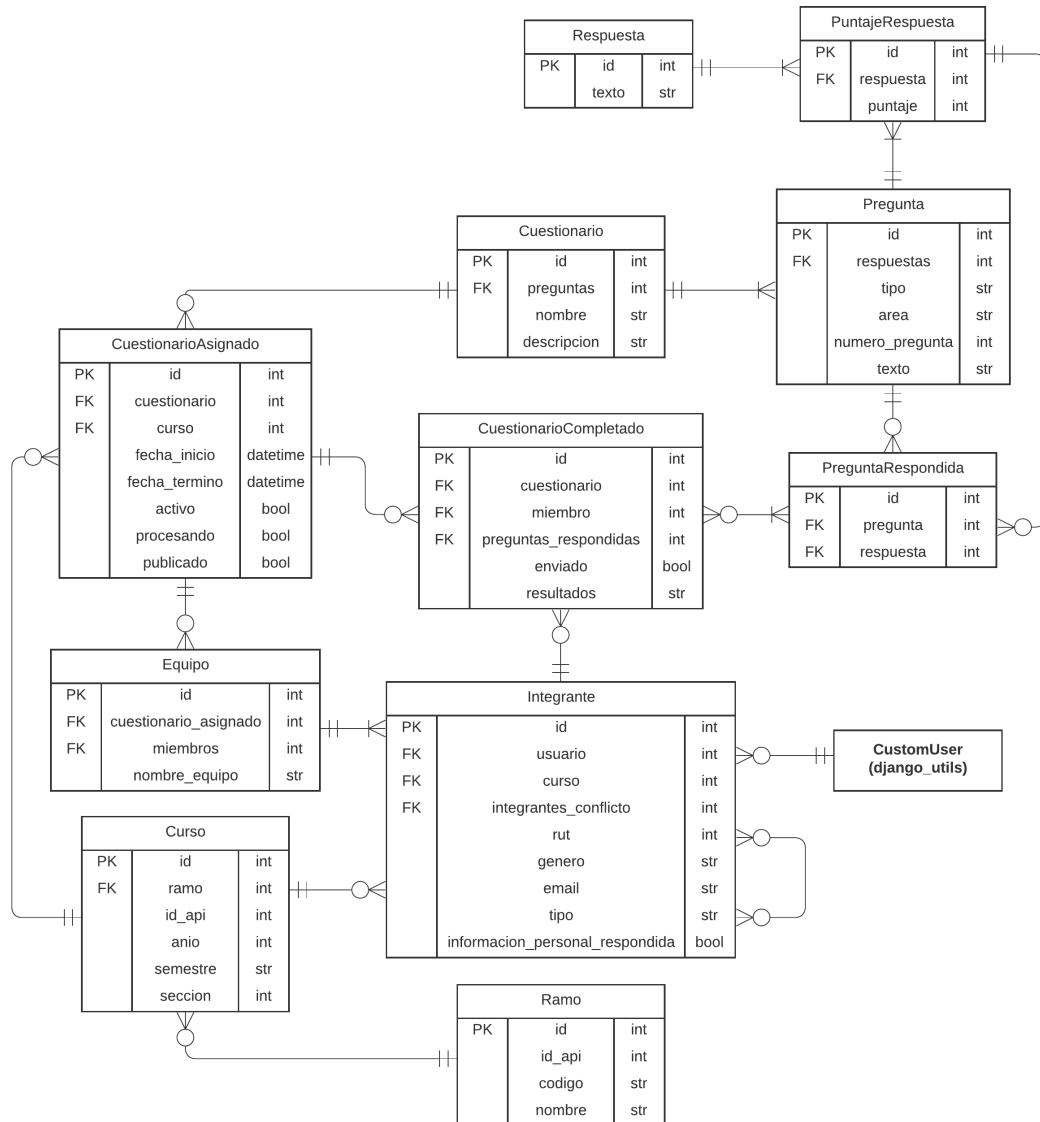


Figura 4.1: Nuevo modelo de datos del sistema

4.7. Descripción de la nueva heurística de conformación de equipos

Realizados los cambios necesarios en el modelo de datos para implementar la perspectiva de género y los conflictos previos, se modificó la heurística de conformación de equipos, y por ende, el algoritmo correspondiente.

Los dos primeros pasos del algoritmo, es decir, la recolección de respuestas y el cómputo de puntajes por área, no sufrieron cambios. Sin embargo, los dos pasos siguientes fueron modificados. A continuación se detallan las nuevas mecánicas para asignar estudiantes a los macrogrupos, y conformar equipos; estos corresponden a los pasos 3 y 4 del algoritmo legado (descrito en la sección 2.4).

- **Paso 3: Asignación a macrogrupos.** Luego de computar los puntajes por área se procede a dividir a los estudiantes en macrogrupos. Cada uno de los macrogrupos agrupa a estudiantes con perfiles psicosociales similares, de acuerdo a sus respuestas al cuestionario.

Esta vez se realizaron dos asignaciones de cuatro macrogrupos, es decir, ocho macrogrupos en total. Los primeros cuatro agrupan a estudiantes que mencionaron su género como femenino, o que no se sentían representados con femenino ni masculino. Los siguientes cuatro macrogrupos tienen las mismas características que los anteriores, pero agrupan al resto de estudiantes, es decir, los que identificaron su género como masculino o que prefirieron no decirlo.

- **Paso 4: Conformación de equipos.** Esta vez la conformación de los equipos comenzó asignando a las personas de los primeros cuatro macrogrupos. Esta asignación comenzó agregando a un estudiante del primer macrogrupo al primer equipo, luego se selecciona una persona del segundo macrogrupo. En caso de que la persona que ya está en el equipo no registre conflictos con esta segunda persona, y viceversa, entonces se agrega la persona seleccionada al equipo. Por el contrario, si se registró un conflicto entre estas personas, se selecciona a otra persona del segundo macrogrupo y se aplica el mismo razonamiento. En caso de no haber más personas en dicho macrogrupo, se pasa al tercer macrogrupo, y así sucesivamente. Luego se aplica este proceso a los siguientes equipos.

Lo anterior se repite hasta que cada grupo esté integrado por dos personas pertenecientes a los primeros cuatro macrogrupos, o hasta que ya no queden personas por repartir en estos. Esto se realiza para respaldar el requerimiento de que siempre debe haber dos personas que se identifiquen como mujer o no binario en un equipo en el que haya una persona identificada como tal.

Si quedan personas sin equipo en los primeros macrogrupos, éstas se reparten secuencialmente en los equipos.

Finalmente, se toman los siguientes cuatro macrogrupos y se reparten sus estudiantes de forma secuencial, siempre aplicando el razonamiento mencionado anteriormente de evitar conflictos. Primero se agregan los individuos del quinto macrogrupo en orden por los distintos equipos, luego los del sexto macrogrupo, terminando con el séptimo y octavo.

El algoritmo anteriormente descrito no evita completamente los conflictos, es por esto que al finalizar la conformación de equipos se realiza una búsqueda extra de conflictos. Esto se logra iterando por cada equipo, y revisando si hay algún conflicto entre cualquier par de integrantes. En caso de detectarse un conflicto, se toma al estudiante y se obtiene su macrogrupo. Luego se busca otro estudiante de su mismo macrogrupo (que pertenezca a otro equipo), y se valida si es viable hacer un intercambio de participantes (*swap*) entre los dos equipos analizados. Si es viable el intercambio, entonces se aplica,

si no lo es, entonces se prueba hacer intercambio con otro estudiante del mismo macrogrupo que esté en otro equipo. Así se procede hasta encontrar el candidato apropiado para realizar el intercambio, o bien se itera hasta llegar al peor caso, donde no es posible eludir el conflicto entre esos pares.

A continuación se muestra un ejemplo simplificado del algoritmo de conformación de equipos. En este ejemplo se muestra el caso de un curso de 8 estudiantes, donde se deben armar dos equipos de 4 integrantes cada uno. En la Tabla 4.1 se muestra cómo quedaron los macrogrupos después de computar los resultados.

Macrogrupo 1	Macrogrupo 2	Macrogrupo 3	Macrogrupo 4
Estudiante 1: - Género femenino - Conflicto con Estudiante 3 Estudiante 2: - Género no binario	Estudiante 3: - Género femenino		Estudiante 4: - Género femenino
Macrogrupo 5	Macrogrupo 6	Macrogrupo 7	Macrogrupo 8
Estudiante 5: - Género masculino	Estudiante 6: - Género masculino - Conflicto con Estudiante 8 Estudiante 7: - Género no especificado	Estudiante 8: - Género masculino	

Tabla 4.1: Resultado de la asignación de macrogrupos (ejemplo de conformación de equipos)

La conformación de los equipos comienza agregando el Estudiante 1 al Equipo 1, luego se selecciona el primer estudiante del segundo macrogrupo (es decir, el Estudiante 3) y se verifica que no existan conflictos entre el Estudiante 1 y el 3. En este caso si existe un conflicto, entonces se pasa al siguiente estudiante del macrogrupo 2, pero ya no quedan, por lo que se buscan estudiantes en el macrogrupo 3, pero está vacío. Se pasa al macrogrupo 4 y se selecciona al estudiante 4. Después de validar que no existen conflictos entre el estudiante 1 y 4, se agrega el Estudiante 4 al Equipo 1.

Continuando con el algoritmo, se agrega al Equipo 2 el siguiente estudiante del primer macrogrupo, el Estudiante 2. Luego, se selecciona el siguiente estudiante disponible, el cual se encuentra en el macrogrupo 4, se verifica que no tengan conflictos y se agrega al Equipo 2. Los equipos quedan de la siguiente forma momentáneamente:

Equipo 1	Equipo 2
Estudiante 1	Estudiante 2
Estudiante 4	Estudiante 3

Tabla 4.2: Equipos después de la asignación de los primeros cuatro macrogrupos (ejemplo de conformación de equipos)

Ahora se deben agregar los estudiantes de los siguientes cuatro macrogrupos. Por lo tanto, se selecciona el primer estudiante del macrogrupo 5 (Estudiante 5), se verifica que no tenga conflictos en el Equipo 1 y se agrega, luego se selecciona al siguiente estudiante del macrogrupo, pero ya no quedan disponibles, por ende, se sigue con el primer estudiante del macrogrupo 6 (Estudiante 6), se verifica que no tenga conflictos con el Equipo 2 y se agrega. Se selecciona al siguiente estudiante del macrogrupo 6 (Estudiante 7), se verifica que no haya conflictos con el Equipo 1 y se agrega. Finalmente, se selecciona el primer integrante del macrogrupo 7 (Estudiante 8), al verificar que no hayan conflictos con el Equipo 2 se puede notar que existen conflictos, específicamente con el Estudiante 6, pero no quedan opciones, se debe agregar al Equipo 2. Los equipos quedan de la siguiente forma:

Equipo 1	Equipo 2
Estudiante 1	Estudiante 2
Estudiante 4	Estudiante 3
Estudiante 5	Estudiante 6
Estudiante 7	Estudiante 8

Tabla 4.3: Equipos después de la asignación de los segundos cuatro macrogrupos (ejemplo de conformación de equipos)

Para finalizar, el algoritmo realiza una búsqueda de conflictos por equipo. En el primer equipo no hay conflictos internos, pero en el segundo sí; se identifica un conflicto entre los estudiantes 6 y 8. Siguiendo el algoritmo, se busca un estudiante del mismo macrogrupo que el Estudiante 6, en este caso, el Estudiante 7. Se analiza si al realizar el intercambio entre estos estudiantes se generan nuevos conflictos en los equipos. La respuesta es no, ya que el Estudiante 7 no tiene conflictos con estudiantes del Equipo 2, y el Estudiante 6 no tiene conflictos con estudiantes del Equipo 1. En conclusión, se realiza el intercambio y finaliza el algoritmo.

Los equipos finales creados por el algoritmo se muestran a continuación. Notar que se cumplen las restricciones de la nueva perspectiva de género y de los conflictos.

Equipo 1	Equipo 2
Estudiante 1	Estudiante 2
Estudiante 4	Estudiante 3
Estudiante 5	Estudiante 6
Estudiante 8	Estudiante 7

Tabla 4.4: Equipos finales conformados por el Sistema de Conformación de Equipos (ejemplo de conformación de equipos)

4.8. Arquitectura física

La nueva arquitectura física mantiene el servidor con su respectiva base de datos, la cual almacena los datos de los cursos, información personal, respuestas de estudiantes y equipos conformados.

El nuevo repositorio se basó en el “django_utils” que se ejecuta en un entorno de Docker. Como se mencionó anteriormente, éste es un conjunto de aplicaciones para Django y en este proyecto se utilizaron dos: el SSO que permite la autenticación de los usuarios con credenciales institucionales y el modelo de usuarios *CustomUser*, que entrega información personal y académica de los usuarios.

Para cargar la información de los cursos y estudiantes se utilizó la API del “Importador de Data de Docencia”. En la Figura 10 se muestra la nueva arquitectura física del sistema.

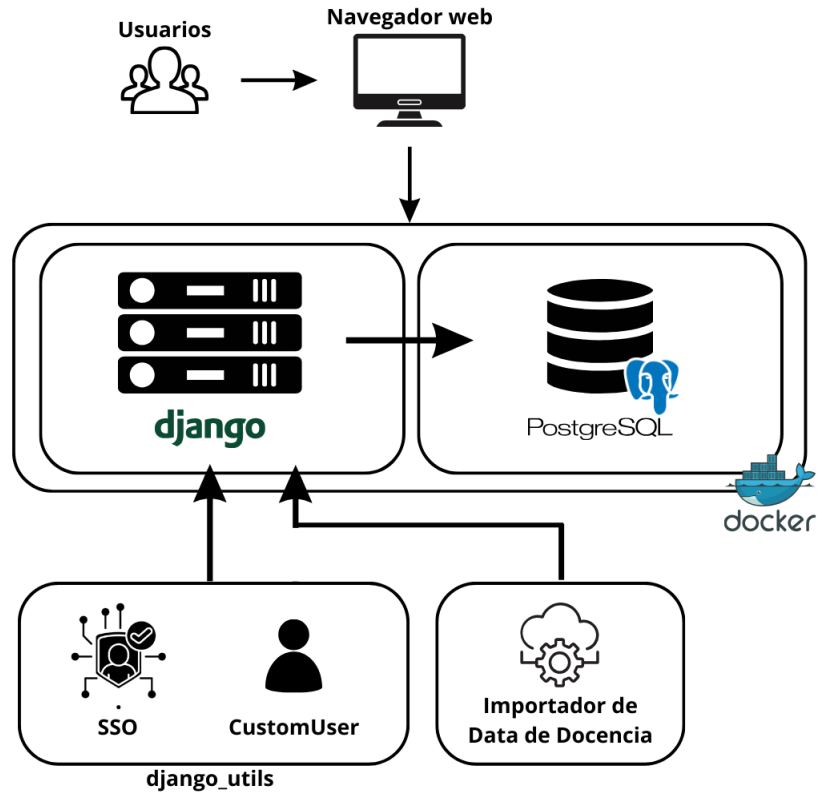


Figura 4.2: Arquitectura física del Sistema de Conformación de Equipos

4.9. Arquitectura lógica

La arquitectura lógica del sistema mantuvo su estructura en capas mencionada en el análisis del sistema legado. Sin embargo, en esta ocasión, se ha implementado de manera más efectiva al adoptar el patrón de diseño de Django, específicamente, el Modelo-Vista-Controlador (MVC), en su variante conocida como Modelo-Vista-Template (MVT). El modelo es el encargado de almacenar y entregar los datos, la vista es la encargada recibir las peticiones de las acciones del usuario y entregar la respuesta correcta, y por último, el template es el que permite la interacción del usuario con el sistema.

Capítulo 5

Implementación de la Solución

En esta sección se detalla el resultado de la implementación de la solución mencionada anteriormente. Este apartado presenta las nuevas interfaces de usuario desarrolladas, destacando su diseño y mejoras para optimizar la experiencia del usuario.

A continuación, se describen las interfaces de usuario del sistema, detallando las funcionalidades disponibles en cada una. Todas las interfaces se crearon usando los templates de Django. Estos templates son archivos que contienen código HTML con incrustaciones de expresiones y etiquetas de Django. Están diseñados para permitir la integración de lógica de presentación dinámica en las aplicaciones web construidas con el framework.

Algunas características clave de los templates de Django son la inclusión de bloques de código predefinidos, la existencia de etiquetas de control de flujo, que permiten incorporar una lógica condicional y por último, la herencia de templates. Esta última característica permite utilizar el *header* (encabezado) y *footer* (pie de página) típicos de las aplicaciones desarrolladas por el Departamento de Ciencias de la Computación.

Tres aspectos relevantes a tener en consideración son los siguientes:

1. Se anonimizaron las identidades de los estudiantes y miembros del equipo docente. Los nombres por defecto serán “Docente X” o “Estudiante X”, donde X es un correlativo que permite diferenciar entre los integrantes del curso.
2. No hay una vista para el inicio de sesión, ya que al utilizar el SSO provisto por el DCC se utilizan las credenciales institucionales de la Universidad de Chile, por ende, son vistas ajenas a esta aplicación.
3. Hay algunos templates completamente nuevos, no obstante, la mayoría de interfaces de usuario se reutilizaron del sistema legado aplicando nuevas funcionalidades y notables mejoras de diseño para mejorar la experiencia de usuario.

5.1. Interfaces comunes entre los usuarios

En general, los usuarios docentes y estudiantes no tienen acceso a la misma interfaz, no obstante, hay dos vistas que comparten. Éstas se indican a continuación.

Vista principal del sistema

La vista principal del sistema muestra los cursos soportados por el mismo, y el nombre de la aplicación web (“Sistema de Conformación de Equipos”). Esta vista también tiene un botón que redirige al usuario a la autenticación utilizando sus credenciales institucionales.

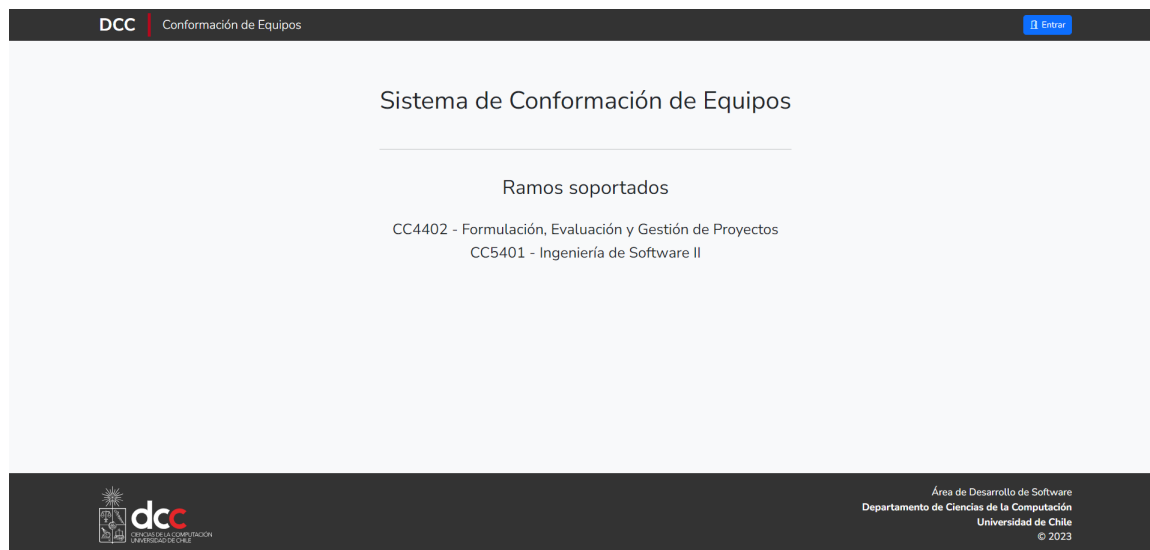


Figura 5.1: Vista principal del Sistema de Conformación de Equipos

Cursos de un usuario

La segunda vista que comparten todos los usuarios es la que lista los cursos de los que forma parte el usuario. En la figura 5.2 se puede ver que el usuario forma parte de los cursos “Formulación, Evaluación y Gestión de Proyectos” e “Ingeniería de Software II” en el semestre de primavera del año 2022. Dicho usuario debe presionar el curso que desea revisar.

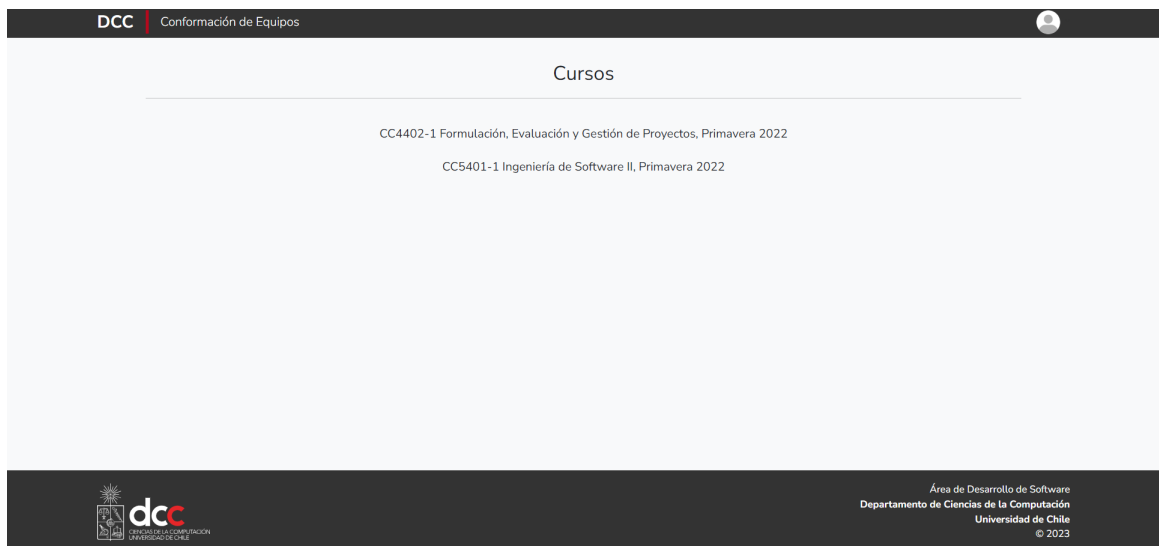


Figura 5.2: Vista de los cursos de un usuario

5.2. Interfaces de usuario docente

Un usuario de tipo docente tiene acceso a múltiples interfaces, ya que es el usuario que tiene más interacción con la plataforma. A continuación se detalla cada una de éstas.

Cuestionarios de un curso

Cuando el usuario de tipo docente ingresa a un curso aparecen los cuestionarios asociados a dicho curso. Cada cuestionario que se muestra corresponde a un proceso de conformación de equipos distinto.

Los cuestionarios se muestran en forma de tabla, donde cada fila corresponde a un cuestionario. El color de la fila indica su estado, si los equipos no han sido publicados, entonces la fila es de color amarillo. Por el contrario, si los equipos fueron publicados, el color de la fila será verde.

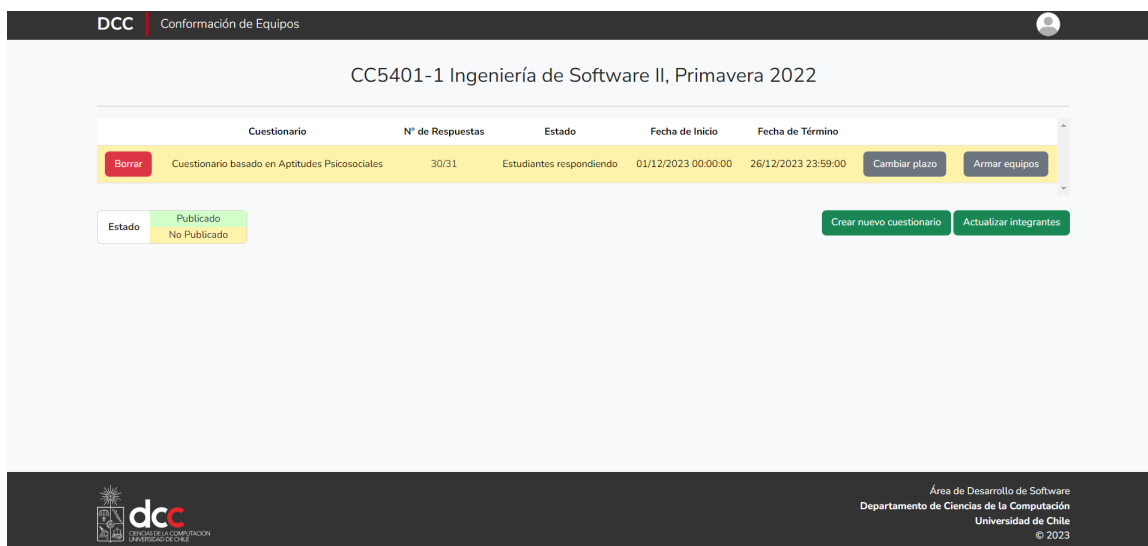


Figura 5.3: Vista de los cuestionarios de un curso para un usuario docente

Para crear un cuestionario se debe presionar el botón “Crear nuevo cuestionario”. Esta acción abrirá una ventana modal con un formulario breve, donde se debe indicar el tipo de cuestionario a utilizar y las fechas de inicio y término del período de respuestas. Actualmente, el único cuestionario implementado es el basado en aptitudes psicosociales.

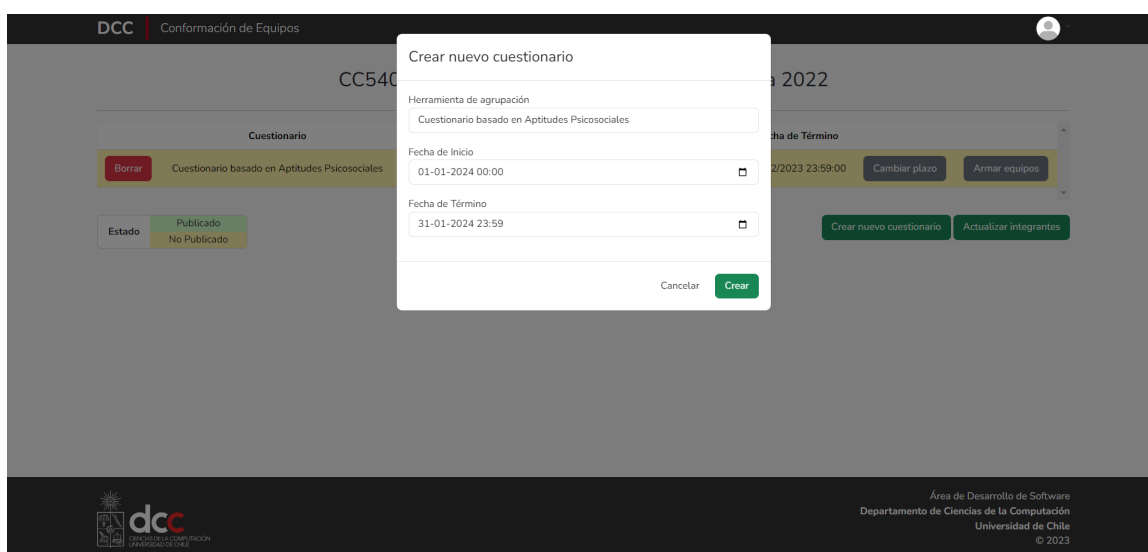


Figura 5.4: Ventana modal de creación de un cuestionario

Cada cuestionario tiene distintos atributos, entre ellos se encuentra su nombre. Al hacer click sobre éste despliega el formulario en modo lectura para el usuario docente (Figura 5.5). A continuación se muestra el número de respuestas que hay actualmente y el número de estudiantes totales del curso. Al presionar el número de respuestas se despliega una ventana que muestra los estudiantes que faltan por contestar el cuestionario (Figura 5.6).

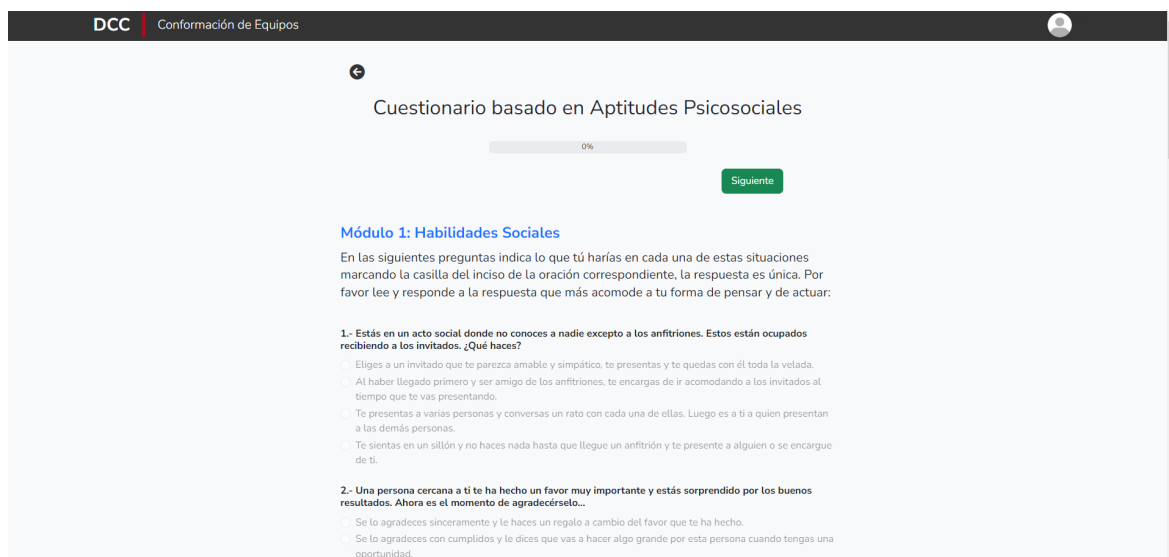


Figura 5.5: Cuestionario en modo lectura

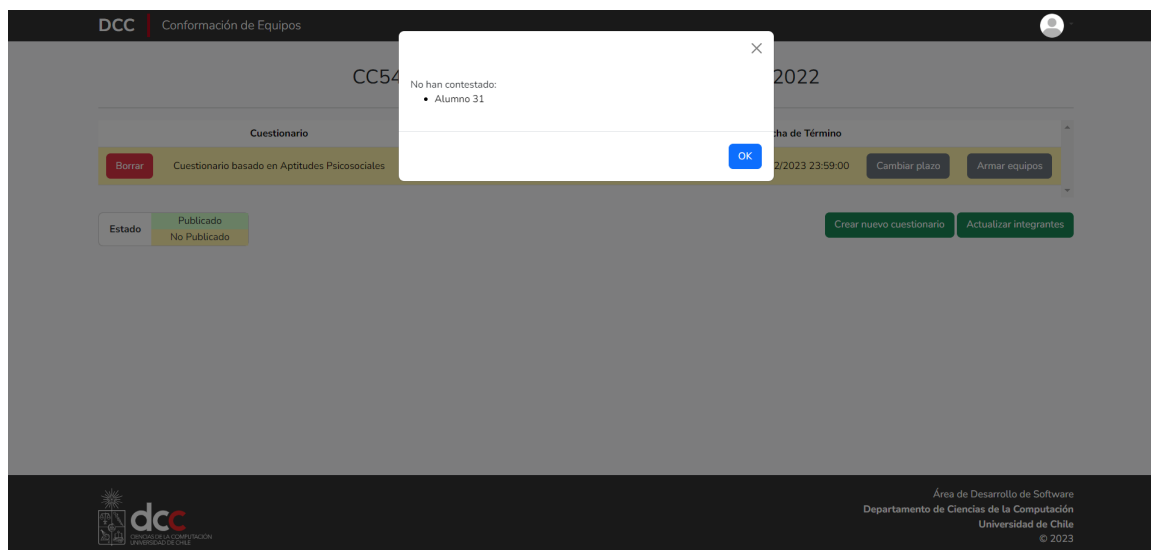


Figura 5.6: Número de estudiantes que aún no han contestado el cuestionario

Luego se muestra una celda que indica uno de tres estados del cuestionario: estudiantes respondiendo, plazo terminado y equipos publicados. Además, se muestran las fechas de inicio y término, permitiendo modificar la fecha de término, es decir, extender el plazo para recibir respuestas al cuestionario (Figura 5.7). Finalmente, se dispone un botón para armar equipos que redirige a otra vista que se describe más adelante en este documento.

Notar que hay un botón llamado “Actualizar integrantes”. Este botón, como indica su nombre, actualiza los integrantes de un curso. Este botón debe ser presionado por un integrante del equipo docente para ingresar al sistema todos los estudiantes de un curso. Esta actualización puede realizarse en cualquier momento, pero idealmente debe llevarse a cabo al comienzo del semestre, luego de finalizar los procesos de modificación de la inscripción académica. Se recomienda esto, ya que un estudiante que no pudo inscribir el curso en una

primera instancia, podría quedar fuera del proceso de conformación de equipos debido a su incorporación tardía al curso.

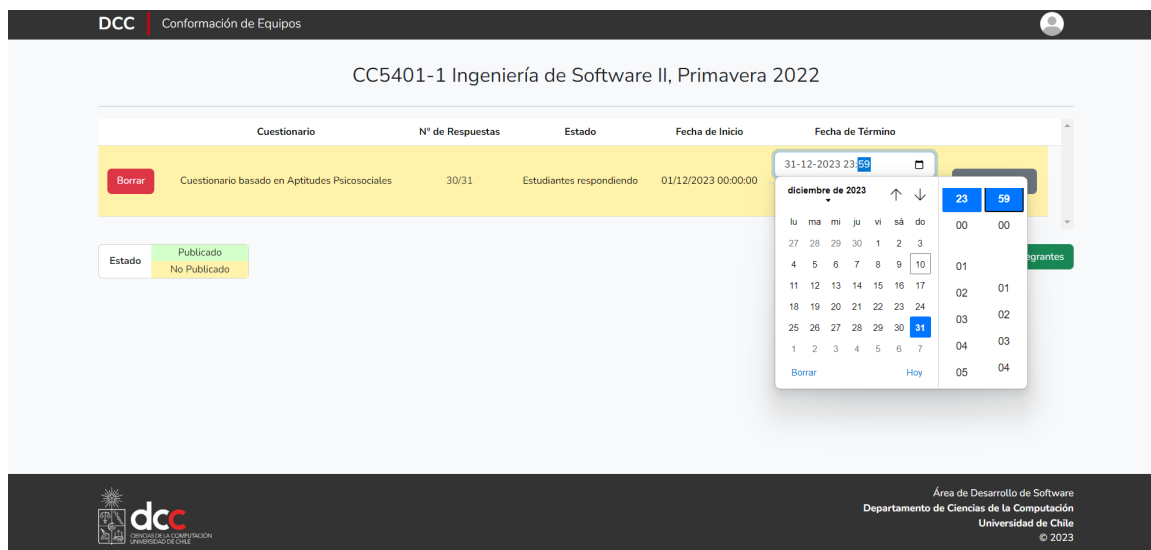


Figura 5.7: Extender plazo del período de respuestas de un cuestionario

Vista para la conformación de equipos

Para la conformación de equipos se mantuvo la interfaz del sistema legado con pequeñas modificaciones estéticas.

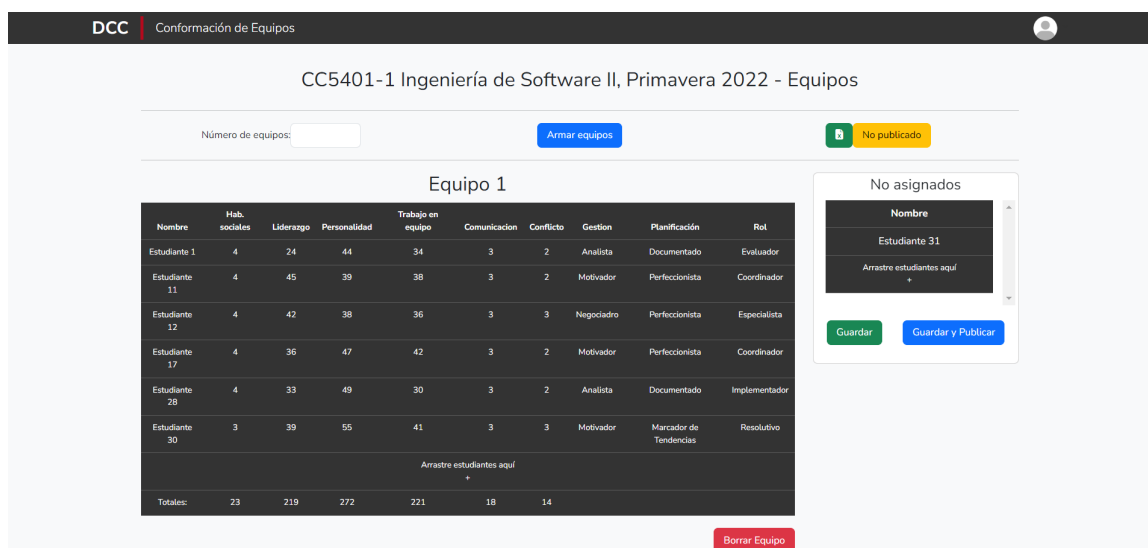


Figura 5.8: Vista para la conformación de equipos

El flujo de la vista comienza cuando el usuario ingresa el número de equipos que desea conformar y presiona el botón “Armar equipos”. Presionar este botón gatilla la conformación de equipos por parte del sistema, es decir, se ejecuta el algoritmo mencionado en la sección 4.7.

Cuando el sistema termina de armar los equipos, los resultados se muestran en forma de tabla; una por equipo. Cada tabla tiene a los integrantes del equipo y un desglose de los puntajes en cada una de las áreas consideradas en el cuestionario basado en aptitudes psicosociales. Al costado derecho se muestran los estudiantes que no fueron asignados por el algoritmo debido a que no respondieron el cuestionario.

El usuario docente puede modificar a su criterio los equipos, incluso puede asignar a un equipo los estudiantes que no hayan respondido el cuestionario. Esto se logra arrastrando y soltando (*drag and drop*) la fila correspondiente a un estudiante. El sistema *drag and drop* es una forma intuitiva y fácil de realizar cambios en la posición de elementos en una interfaz gráfica.

En la parte superior de la interfaz hay un botón verde, el cual permite descargar un archivo con los equipos en formato CSV. En la Figura 5.9 se muestra un ejemplo del archivo descargado.

Equipo 1									
	Habilidades Sociales	Liderazgo	Personalidad	Aptitud Para Trabajo en Equipo	Estilo de Comunicación	Conflicto	Capacidad de Gestión	Estilo de Planificación	Tipo de Rol
Estudiante 1	4	24	44	34	3	2	Analista	Documentado	Evaluador
Estudiante 11	4	45	39	38	3	2	Motivador	Perfeccionista	Coordinador
Estudiante 12	4	42	38	36	3	3	Negociador	Perfeccionista	Especialista
Estudiante 17	4	36	47	42	3	2	Motivador	Perfeccionista	Coordinador
Estudiante 28	4	33	49	30	3	2	Analista	Documentado	Implementador
Estudiante 30	3	39	55	41	3	3	Motivador	Marcador de Tendencias	Resolutivo
Equipo 2									
	Habilidades Sociales	Liderazgo	Personalidad	Aptitud Para Trabajo en Equipo	Estilo de Comunicación	Conflicto	Capacidad de Gestión	Estilo de Planificación	Tipo de Rol
Estudiante 8	4	30	45	31	3	2	Analista	Perfeccionista	Especialista

Figura 5.9: Ejemplo de archivo CSV con los equipos conformados

5.3. Interfaces de usuario estudiante

Un usuario de tipo estudiante tiene acceso al sistema, principalmente, para responder los cuestionarios activos de sus cursos. A continuación se describen las vistas a las que tiene acceso.

Cuestionarios de un curso

Al igual que un usuario docente, el usuario estudiante tiene una vista para ver los cuestionarios activos de sus cursos. Sin embargo, estas vistas son radicalmente distintas según el tipo de usuario.

La vista para un usuario estudiante muestra los cuestionarios en una tabla, donde cada fila corresponde a un cuestionario. Por cada cuestionario se muestra la fecha límite para responder, y el estado en el que se encuentra (en plazo, plazo terminado o equipos publicados). Finalmente, se tiene un botón que permite al usuario responder el cuestionario si está en el plazo. El texto del botón tiene varias alternativas, las cuales se muestran a continuación:

1. “Responder”: Cuando el estudiante no ha respondido ninguna pregunta del cuestionario y se encuentra dentro del plazo.

2. “Continuar respondiendo”: Cuando el estudiante respondió algunas preguntas, pero no completó el cuestionario y aún está dentro del plazo.
3. “Editar respuestas”: Cuando el estudiante completó el cuestionario y aún queda tiempo para el final del plazo.
4. “Ver respuestas”: Cuando el estudiante completó el cuestionario y terminó el plazo.

En caso de que pase la fecha de término y el estudiante no completó el cuestionario, se le informa con un texto que dice “Plazo terminado” o “Equipos publicados”, de acuerdo al estado del cuestionario.

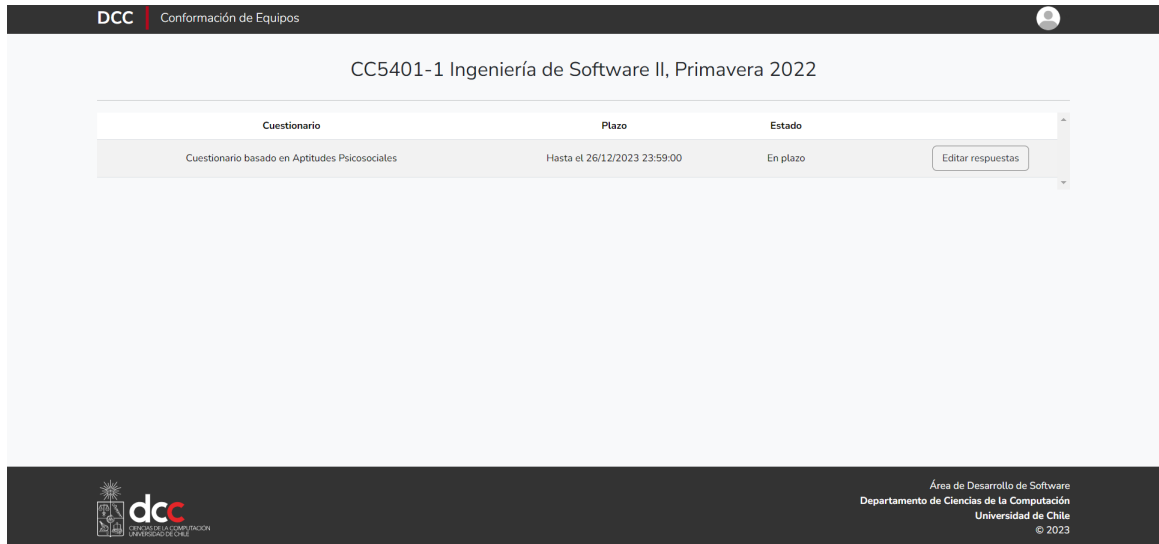


Figura 5.10: Vista de los cuestionarios de un curso para un usuario estudiante

Formulario de información personal

Cuando el estudiante presiona el botón para responder el cuestionario, se le redirige al formulario de información personal. Este formulario consta de tres secciones:

1. Correo electrónico: En esta sección se solicita al estudiante ingresar un correo electrónico donde desee recibir los resultados de la conformación de equipos. El sistema accede con su rut a la API del Importador de Data de Docencia, e intenta extraer el correo electrónico de la persona. En caso de que se obtenga dicho correo, se le indica al usuario que hay un correo registrado y se le solicita cambiarlo si así desea.
2. Identidad de género: Se le solicita al estudiante indicar con qué género se identifica, esto para integrar la perspectiva de género correctamente, ya que desde la API del Importador de Data de Docencia se pueden extraer géneros erróneos. En el selector hay cuatro opciones: “masculino”, “femenino”, “ninguno de los anteriores me identifica” y “prefiero no decirlo”.
3. Registro de conflictos: En esta sección opcional, el usuario estudiante puede indicar un máximo de dos personas con las que no desee trabajar, debido a algún conflicto previo. El registro se realiza seleccionando los nombres de los integrantes en cuestión.

DCC | Conformación de Equipos

Información personal

Responda las siguientes preguntas para una mejor conformación de equipos

1. Correo electrónico
 Su correo electrónico actual es: estudiante1@ug.uchile.cl
 Si desea que los resultados de los equipos sean enviados a otro correo, ingréselo a continuación.

2. Identidad de género
 Indique con qué género se identifica.

3. Registro de conflictos [Opcional]
 Si usted tiene un conflicto con un integrante del curso indíquelo marcando el nombre de la persona.
 Puede seleccionar un máximo de 2 personas.

- Estudiante 2
- Estudiante 3
- Estudiante 4
- Estudiante 5
- Estudiante 6
- Estudiante 7
- Estudiante 8
- Estudiante 9

Figura 5.11: Formulario de información personal

Responder un cuestionario

Luego de llenar y enviar el formulario de información personal, la plataforma redirige al usuario a responder el cuestionario.

DCC | Conformación de Equipos

Cuestionario basado en Aptitudes Psicosociales

0%

[Siguinte](#)

Módulo 1: Habilidades Sociales

En las siguientes preguntas indica lo que tú harías en cada una de estas situaciones marcando la casilla del inicio de la oración correspondiente, la respuesta es única. Por favor lee y responde a la respuesta que más acomode a tu forma de pensar y de actuar:

1.- Estés en un acto social donde no conoces a nadie excepto a los anfitriones. Estos están ocupados recibiendo a los invitados. ¿Qué haces?

- Eliges a un invitado que te parezca amable y simpático, te presentas y te quedas con él toda la velada.
- Al haber llegado primero y ser amigo de los anfitriones, te encargas de ir acomodando a los invitados al tiempo que te vas presentando.
- Te presentas a varias personas y conversas un rato con cada una de ellas. Luego es a ti a quien presentan a las demás personas.
- Te sientas en un sillón y no haces nada hasta que llegue un anfitrión y te presente a alguien o se encargue de ti.

2.- Una persona cercana a ti te ha hecho un favor muy importante y estás sorprendido por los buenos resultados. Ahora es el momento de agradecerse...

- Se lo agradece sinceramente y le haces un regalo a cambio del favor que te ha hecho.
- Se lo agradece con cumplidos y te dices que vas a hacer algo grande por esta persona cuando tengas una oportunidad.

Figura 5.12: Vista inicial del cuestionario basado en aptitudes psicosociales

El cuestionario consta de varios módulos, y la navegación entre ellos se realiza a través de botones. Existen botones para ir al módulo anterior y al siguiente, un botón para ir al inicio del cuestionario (es decir, a la posición de la Figura 5.12), y un botón de guardado parcial, que permite al estudiante responder una parte del cuestionario y seguir respondiendo el resto en otro momento. Cuando el estudiante llega al final del cuestionario, debe presionar el botón “Enviar” y se considerarán enviadas sus respuestas.

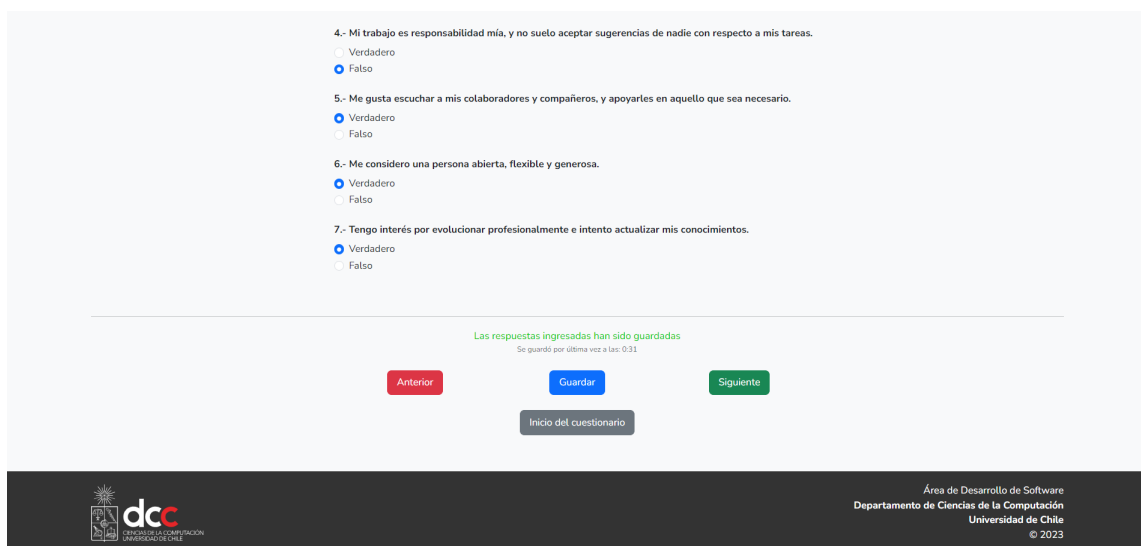


Figura 5.13: Botones de navegación dentro del cuestionario

5.4. Resumen de la Solución

A modo de resumen, se presenta una tabla comparativa que destaca los principales ítem modificados en el sistema legado, la justificación del cambio y la nueva solución implementada. Además, se mencionan las nuevas funcionalidades incorporadas al sistema.

Ítem	Solución anterior	Razón del cambio	Nueva solución
Versión de Django	Versión de Django deprecada (1.11.29)	Código deprecado y desactualizado	Versión de Django con soporte (>4.2)
Gobernabilidad del sistema	Sistema desarrollado por estudiantes, en memorias y cursos	El sistema debe ser gobernable por el DCC	Basar el repositorio en el “django_utils”, desarrollo del DCC
Orden del repositorio	Código dispuesto en un solo directorio	El código no es mantenible	Crear aplicaciones de Django
Carga de datos	Carga manual de los datos de un curso	Facilitar el trabajo del equipo docente y evitar error humano	Integración del Importador de Data de Docencia
Perspectiva de género	No existe	Mayor diversidad, equidad y comodidad dentro de los equipos	Se incluye variable de género en el algoritmo de conformación
Registro de conflictos	No existe	Evitar conflictos internos en los equipos semestrales	Se incluye variable de conflictos en el algoritmo de conformación

Tabla 5.1: Resumen de la solución

Capítulo 6

Evaluación de la Solución

Evaluar la solución en un proyecto de desarrollo de software es fundamental para garantizar el éxito de lo construido, ya que proporciona una visión crítica y objetiva del trabajo desarrollado, permitiendo identificar deficiencias y oportunidades de mejora.

La evaluación de la solución busca validar si se cumplen los objetivos del proyecto desde la propia visión de los usuarios finales. Es por esto que se evaluarán dos aspectos muy importantes: usabilidad y utilidad percibida.

6.1. Evaluación de la usabilidad percibida

Según Andreas Holzinger [18], la evaluación final se debe llevar a cabo, idealmente, con 20 o más usuarios en caso de aplicaciones masivas o parcialmente masivas. Por esta razón, la evaluación de usabilidad para el perfil de estudiantes se realizó con un grupo de 20 personas, las cuales debieron ingresar a la plataforma como usuario de tipo estudiante y responder un cuestionario, tal como si estuvieran a inicio de semestre en un curso que utiliza esta herramienta como apoyo. A cada persona que participó de la evaluación se le pidió realizar los siguientes pasos:

- Paso 1: Iniciar sesión en la plataforma.
- Paso 2: Ingresar al curso “CC5401-50 Ingeniería de Software II, Primavera 2022” (Curso ficticio creado para efectos de la evaluación).
- Paso 3: Responder el cuestionario disponible rellenando tanto su información personal como el cuestionario basado en aptitudes psicosociales.
- Paso 4: Responder un formulario con la encuesta SUS.

Todas las personas que participaron en la evaluación siguieron los pasos sin equivocaciones. A continuación, se detallan los resultados de la encuesta SUS realizada a este grupo.

La encuesta SUS consta de 10 enunciados (ver detalle en el Anexo C), de los cuales cinco son positivos y cinco son negativos; específicamente los enunciados impares son los positivos. Cada enunciado tiene cinco respuestas posibles, las cuales corresponden a la escala de Likert [19] y pueden transformarse a un valor numérico: “Totalmente en desacuerdo” (1), “En desacuerdo” (2), “Neutro” (3), “De acuerdo” (4), “Totalmente de acuerdo” (5).

El resultado de la encuesta SUS se obtiene realizando los siguientes cálculos:

1. Suma las respuestas de los enunciados impares y después resta 5.
2. Suma las respuestas de los enunciados pares y resta ese total a 25.

Con ambos valores calculados se realiza la suma de ambos, y el resultado se multiplica por 2.5. Estos pasos se deben aplicar sobre cada una de las encuestas respondidas, para finalmente calcular el promedio de todas las encuestas, llegando así al resultado final.

Aplicando estos pasos se llegó a un puntaje de 86.6 puntos. Como referencia, Jeff Sauro [20] establece, en un estudio, que el puntaje promedio de la encuesta SUS corresponde a 68 puntos, y menciona que cualquier puntaje por encima de ese valor se considera aceptable.

	Enun. 1	Enun. 2	Enun. 3	Enun. 4	Enun. 5	Enun. 6	Enun. 7	Enun. 8	Enun. 9	Enun. 10	Impares	Pares	Total
Usuario 1	5	1	5	1	5	2	5	1	5	1	20	19	97,5
Usuario 2	3	3	4	4	4	4	4	2	4	1	14	11	62,5
Usuario 3	4	2	4	2	4	2	3	2	4	2	14	15	72,5
Usuario 4	4	1	4	2	4	3	5	2	4	3	16	14	75
Usuario 5	5	1	5	1	5	1	5	1	5	1	20	20	100
Usuario 6	1	3	4	2	3	1	4	2	3	1	10	16	65
Usuario 7	4	1	5	1	5	1	4	1	5	5	18	16	85
Usuario 8	5	1	5	2	4	1	5	2	5	2	19	17	90
Usuario 9	5	1	5	1	5	1	5	1	5	1	20	20	100
Usuario 10	4	4	2	3	4	2	3	2	4	2	12	12	60
Usuario 11	5	1	5	1	5	1	5	1	5	1	20	20	100
Usuario 12	5	1	5	1	5	1	5	1	5	1	20	20	100
Usuario 13	5	1	5	1	5	1	5	1	5	1	20	20	100
Usuario 14	5	2	4	2	5	1	5	1	5	1	19	18	92,5
Usuario 15	5	1	5	2	5	2	5	1	5	5	20	14	85
Usuario 16	4	2	5	1	4	2	5	3	5	1	18	16	85
Usuario 17	5	1	5	1	5	1	5	1	5	1	20	20	100
Usuario 18	5	1	5	1	5	1	5	1	5	1	20	20	100
Usuario 19	5	1	5	1	5	1	5	1	5	1	20	20	100
Usuario 20	3	3	4	3	5	3	3	3	4	2	14	11	62,5
													86,6

Figura 6.1: Resultados de la evaluación de usabilidad percibida

Los resultados de la encuesta SUS realizada son claros: El sistema es usable.

Comparando el puntaje de cada encuesta respondida, se puede identificar algunos que son menores o cercanos al promedio. Los comentarios recibidos por los encuestados que dieron estos bajos puntajes se basan principalmente en la extensión del cuestionario; que era bastante largo y tedioso de responder. Es por esto que el enunciado “Pienso que me gustaría usar frecuentemente este sistema” fue el peor evaluado. Por otra parte, “Las personas aprenderían

rápidamente cómo utilizar el sistema” y “Opino que el sistema fue fácil de usar” fueron los mejor evaluados, indicando una buena construcción y mejora del software.

6.2. Evaluación de la utilidad percibida

Para evaluar la utilidad percibida, se citó a una reunión virtual a los equipos docentes de los cursos Ingeniería de Software II y Formulación, Evaluación y Gestión de Proyectos, quienes utilizan todos los semestres el sistema de conformación de equipos. A esta reunión asistieron cuatro personas (dos profesores y dos auxiliares), quienes tienen el perfil de administrador (o académico/auxiliar de un curso), y evaluaron la funcionalidad del sistema para ese perfil de usuario.

A estas personas se les brindó una explicación de la plataforma y de sus nuevas funcionalidades. Para ello se realizó el flujo completo que llevaría a cabo un usuario docente, esta vez sobre un curso ficticio creado para la ocasión.

Después de la reunión se les pidió completar una versión reducida del cuestionario de utilidad percibida propuesto por TAM (*Technology Acceptance Model*) [6]. Este es un modelo que establece el grado de aceptación de una nueva tecnología o aplicación, en función de la utilidad percibida por los potenciales usuarios de la misma. Este modelo teoriza que se puede inferir si cierta población, en este caso, tipo de usuario, está predispuesto a incorporar cosas nuevas o no.

En el contexto de la evaluación de la solución, se realizó una encuesta de tres enunciados, los cuales se encuentran en el Anexo D), y buscan determinar la utilidad percibida por los usuarios reales, respecto a la funcionalidad entregada por el sistema intervenido. De los enunciados, dos son positivos y uno es negativo; los enunciados impares son los positivos.

La escala utilizada para la evaluación fue una variación de la escala de Likert [19], transformando cada respuesta a un valor numérico. Las posibles respuestas son: “Muy en desacuerdo” (1), “En desacuerdo” (2), “Ligeramente en desacuerdo” (3), “Neutral” (4), “Ligeramente de acuerdo” (5), “De acuerdo” (6), “Muy de acuerdo” (7), siendo 21 el puntaje máximo. Los resultados de la encuesta TAM se muestran a continuación.

	Enunciado 1	Enunciado 2	Enunciado 3	Total
Usuario 1	Muy de acuerdo	Muy en desacuerdo	Muy de acuerdo	21
Usuario 2	Muy de acuerdo	En desacuerdo	Muy de acuerdo	20
Usuario 3	De acuerdo	En desacuerdo	De acuerdo	18
Usuario 4	Muy de acuerdo	Muy en desacuerdo	Muy de acuerdo	21

Figura 6.2: Resultados de la evaluación de utilidad percibida

Para calcular el puntaje final de cada encuesta respondida, se utilizó la siguiente fórmula:

$$Total = Puntaje\ Enunciado\ 1 + (8 - Puntaje\ Enunciado\ 2) + Puntaje\ Enunciado\ 3 \quad (6.1)$$

A partir de los resultados de la encuesta y los comentarios de los usuarios, se concluye que el sistema es útil.

Los comentarios destacaron la solución de uno de los problemas más significativos que había en el sistema legado: la creación de cursos y la carga de sus miembros. Asimismo, se elogió la actualización de la herramienta y su capacidad para continuar evolucionando. Además, se recibieron sugerencias que facilitaron la implementación de pequeños ajustes finales en el sistema.

6.3. Evaluación de la consistencia en la conformación de los equipos

Aparte de evaluar la usabilidad y utilidad percibida, se evaluó la consistencia del nuevo algoritmo de conformación de equipos con respecto al del sistema legado, debido a que se desea mantener la diversidad psicosocial que éste producía en los equipos. Si bien, el nuevo algoritmo sigue una heurística similar al anterior, se debe evaluar que la inclusión del género y los conflictos no afecten la diversidad psicosocial de los equipos conformados, ya que al agregar estos nuevos parámetros se generan más restricciones en la conformación.

Para llevar a cabo esta evaluación se compararon dos procesos de conformación de equipos usando los datos del mismo curso, es decir, se tomaron las respuestas de los estudiantes en el proceso de conformación de equipos realizado en el curso Ingeniería de Software II durante el semestre de primavera el año 2022, y se copiaron en la base de datos de desarrollo para utilizarlas en una nueva instancia de conformación de equipos; pero esta vez, utilizando el nuevo algoritmo. En la evaluación solo se consideró a los 29 estudiantes que respondieron el cuestionario.

El resultado esperable es que la conformación de equipos en ambas instancias sean distintas, pero equivalentes, ya que el nuevo algoritmo comienza a armar los equipos de una forma diferente al legado. Sin embargo, el nuevo algoritmo mantiene la estrategia de balanceo de capacidades y calidad entre equipos.

Para comparar ambos algoritmos, se revisó cuántos estudiantes de cada macrogrupo hay en los primeros tres equipos conformados, puesto que ese es un buen indicador de diversidad psicosocial.

Debido a que en ambos algoritmos se mantuvo la forma de determinar el macrogrupo de un estudiante, ambas conformaciones se realizaron con los mismos macrogrupos. La cantidad de estudiantes en cada macrogrupo se muestra a continuación.

Macrogrupo 1	Macrogrupo 2	Macrogrupo 3	Macrogrupo 4
19	8	2	0

Tabla 6.1: Resultados de asignación de macrogrupos (evaluación de consistencia)

Sistema legado

Realizada la conformación de equipos en el sistema legado se revisaron los primeros tres equipos creados y se identificó la cantidad de estudiantes que pertenecían a cada macrogrupo. Los resultados se muestran en la siguiente tabla.

	Macrogrupo 1	Macrogrupo 2	Macrogrupo 3	Macrogrupo 4
Equipo 1	4	1	1	0
Equipo 2	4	2	0	0
Equipo 3	3	2	1	0

Tabla 6.2: Cantidad de estudiantes por macrogrupo en los equipos conformados por el sistema legado (evaluación de consistencia)

Nuevo sistema

Por otra parte, se realizó la misma conformación de equipos en el nuevo sistema. Nuevamente se revisó la composición de los primeros tres equipos. Los resultados se muestran a continuación.

	Macrogrupo 1	Macrogrupo 2	Macrogrupo 3	Macrogrupo 4
Equipo 1	3	2	1	0
Equipo 2	4	2	0	0
Equipo 3	4	1	1	0

Tabla 6.3: Cantidad de estudiantes por macrogrupo en los equipos conformados por el nuevo sistema (evaluación de consistencia)

A partir del análisis de ambas tablas, se puede evidenciar que en ambos casos hay una diversidad psicosocial similar. Esto se refleja bajo el indicador de que ambos algoritmos son consistentes si los equipos tienen una variedad en la cantidad de estudiantes pertenecientes a cada macrogrupo.

6.4. Resumen del proceso de evaluación

La evaluación de la solución desarrollada para el Sistema de Conformación de Equipos se dividió en tres componentes clave: la evaluación de la usabilidad percibida, la evaluación de

la utilidad percibida y la evaluación de la consistencia en la conformación de equipos.

Evaluación de la usabilidad percibida

La usabilidad percibida se evaluó mediante la aplicación de la encuesta SUS a un grupo de 20 usuarios representativos del perfil de estudiantes. El puntaje promedio obtenido fue de 86.6 puntos, superando significativamente el puntaje de referencia de 68 puntos establecido por Jeff Sauro [20]. Los resultados indican que el sistema es altamente usable, aunque se identificaron algunas críticas relacionadas con la longitud del cuestionario.

Evaluación de la utilidad percibida

La utilidad percibida se evaluó a través de una encuesta basada en el modelo TAM, con la participación de equipos docentes de los cursos donde se utiliza este sistema. Los resultados indicaron una percepción positiva de la utilidad del sistema. Los comentarios resaltaron la solución de problemas significativos y la capacidad del sistema para evolucionar.

Evaluación de la consistencia en la conformación de equipos

Se evaluó la consistencia del nuevo algoritmo de conformación de equipos comparándolo con el sistema legado. La comparación se centró en la distribución de estudiantes de cada macrogrupo en los primeros tres equipos conformados. Los resultados demostraron que, aunque los equipos eran distintos, mantenían una consistencia equivalente en términos de diversidad psicosocial.

Capítulo 7

Conclusiones y Trabajo a Futuro

El desarrollo de este trabajo de título ha permitido abordar de forma integral problemas profundos en las arquitecturas física y lógica del sistema legado, e integrar aspectos relevantes para la conformación de equipos.

La actualización del framework sobre el cual está construido el sistema, ha sido el principal objetivo de esta memoria. Esto requería actualizar varias dependencias que también presentaban versiones desactualizadas. El logro de este objetivo utilizando una herramienta que está en constante actualización, como lo es el repositorio “django_utils”, no solo contribuye al aprovechamiento de nuevas características, mejoras de seguridad y correcciones de *bugs*, sino que también hace al sistema gobernable y mantenible a través del tiempo. Esto lo prepara para futuras mejoras y garantiza la vigencia de la herramienta para la conformación de equipos de los cursos Ingeniería de Software II y Formulación, Evaluación y Gestión de Proyectos.

Además de actualizar las dependencias utilizadas en el sistema, se realizaron diversas mejoras al código. Se utilizaron herramientas provistas por la nueva versión de Django para mantener buenas prácticas, se optimizaron algoritmos y consultas a la base de datos, se crearon y actualizaron templates para incluir nuevas funcionalidades, como el formulario de información personal y la navegación por el cuestionario que responden los estudiantes, y se mejoró la experiencia del usuario en la plataforma al hacer visible el estado del sistema en casos necesarios e incluir confirmaciones para algunas acciones críticas.

En la actualidad, el trabajo en equipo es fundamental en el desarrollo de software, donde la comunicación, el respeto mutuo y el complemento de habilidades es clave para el éxito de un proyecto. La conformación de equipos cohesivos no solo se basa en habilidades técnicas, también se deben considerar factores psicosociales, aspecto que trataba bastante bien el sistema legado, pero faltaban dos factores muy importantes: adoptar una perspectiva de género y la gestión de conflictos previos entre pares.

La incorporación de una perspectiva de género en el sistema es un avance significativo hacia un ambiente más inclusivo. Esta consideración para la conformación de equipos no solo promueve un entorno de trabajo más cómodo y seguro para las personas involucradas, sino que también potencia la comunicación, creatividad y resolución de problemas en los equipos.

Incluir el mecanismo de registro de conflictos previos, y posterior integración en el algoritmo de conformación de equipos, permite evitar posibles tensiones dentro de los mismos, aportando a un ambiente más favorable para el éxito del proyecto.

La automatización de los procesos de actualización de cursos y sus integrantes accediendo a una API del DCC, añade un gran valor en cuanto a la eficiencia del proceso de conformación de equipos. Eliminar la necesidad de realizar una carga manual de datos, aparte de reducir la probabilidad de cometer errores, libera mucho tiempo a los docentes.

En conclusión, la reingeniería del Sistema de Conformación de Equipos ha permitido superar diversas limitaciones y mejorar significativamente su experiencia de usuario y funcionalidad, adaptándola a demandas actuales. Este trabajo de título contribuye no solo en el proceso de conformación de equipos, sino que también en la experiencia de aprendizaje de los estudiantes, preparándolos de manera más efectiva para los desafíos del trabajo en equipo.

Gracias a la reingeniería realizada al Sistema de Conformación de Equipos es que es posible extender sus funcionalidades a futuro. Algunas de estas extensiones son las siguientes:

- **Integración con el Sistema de Coevaluaciones:** Como se mencionó al comienzo del documento, un servicio que integre ambos sistemas sería muy beneficioso para obtener datos valiosos que permitan seguir mejorando estas herramientas.
- **Optimización de la conformación de equipos:** Analizar si existen variables que permitan evaluar la conformación de equipos. En caso de encontrar alguna variable, definir una medida cuantitativa de la calidad de los equipos y reformar el algoritmo para obtener equipos con un valor similar.
- **Análisis de datos y estadísticas:** Implementar un módulo de análisis de datos que permita a los docentes revisar estadísticas sobre la eficacia de la conformación de equipos, el rendimiento de los estudiantes y otros indicadores relevantes.

Bibliografía

- [1] Y. Lindsjörn, D. Sjøberg, T. Dingsøy, G. Bergersen, and T. Dybå. *Teamwork quality and project success in software development: a survey of agile development teams*. Journal on Systems and Software 122: 274-286, 2016.
- [2] J. S. Karn, S. Syed-Abdullah, A. J. Cowling, and M. Holcombe. *A study into the effects of personality type and methodology on cohesion in software engineering teams*. Behavior Information Technology, volume 26, issue 2, Taylor Francis, pages: 99 – 111, 2007.
- [3] L. Silvestre. *Diseño de equipos de desarrollo de software en escenarios universitarios*. Tesis de Magíster en Ciencias Mención Computación. DCC, FCFM, Universidad de Chile, 2012.
- [4] *Herramienta de conformación de equipos*. Disponible en: <https://equipos.dcc.uchile.cl/>, 2012.
- [5] J. Brooke. *SUS: A quick and dirty usability scale*. Usability Eval. Ind. 189, 1995.
- [6] F. D. Davis. *Perceived usefulness, perceived ease of use, and user acceptance of information technology*. MIS Quarterly 13 (3): 319-340, doi:10.2307/249008, 1989.
- [7] C. A. Wellington, T. Briggs, and C. D. Girard. *Examining team cohesion as an effect of software engineering methodology*. Proceedings of the 2005 Workshop on Human and Social Factors of Software Engineering (HSSE'05), 2005.
- [8] A. Law and R. Charron. *Effects of agile practices on social factors*. Proceedings of the 2005 workshop on Human and social factors of software engineering (HSSE'05), 2005.
- [9] J. Lavados. *Sistema de conformación de equipos para proyecto de software*. Memoria de Ingeniería Civil en Computación. DCC, FCFM. Universidad de Chile, 2023.
- [10] D. Díaz. *Optimización de la conformación de equipos de proyecto de software*. Memoria de Ingeniería Civil en Computación. DCC, FCFM. Universidad de Chile, 2020.
- [11] Curso CC5401 Ingeniería de Software II. *Documento Histórico del Proyecto Conformación de Equipos de Desarrollo en Escenarios Académicos*. Reporte Técnico. DCC, Universidad de Chile, 2018.
- [12] M. Belbin. *Management Teams, Why They Succeed or Fail*. London: Heinemann, 1981.

- [13] Área de Desarrollo del Departamento de Ciencias de la Computación. FCFM. Universidad de Chile. Disponible en: <https://github.com/DCC-FCFM-UCHILE/django-boilerplate>, 2022.
- [14] Área de Desarrollo del Departamento de Ciencias de la Computación. FCFM. Universidad de Chile. *Importador de Data de Docencia*. Disponible en: <https://pad.test.dcc.uchile.cl/importer/mufasa>, 2023.
- [15] M. C. Bastarrica and J. Simmonds. *The effects of team gender composition in capstone software development projects*. Proc. of the 41st International Conference of the Chilean Computer Science Society (SCCC), pp. 1-6, doi: 10.1109/SCCC57464.2022.10000331, 2022.
- [16] F.R. De Wit, K.A. Jehn, and D. Scheepers. *Task conflict, information processing, and decision-making: the damaging effect of relationship conflict*. Organ. Behav. Hum. Decis. Process. 122 (2) 177–189, 2013.
- [17] Curso CC5401 Ingeniería de Software II. *Documento Histórico del Proyecto Co-Evaluaciones de Equipos de Desarrollo de Software*. Reporte Técnico. DCC, Universidad de Chile, 2017.
- [18] A. Holzinger. *Usability engineering methods for software developers*. Commun. ACM 48, 1, 71–74, 2005.
- [19] R. A. Likert. *Technique for the measurement of attitudes*. Archives of Psychology, 22 140, 55, 1932.
- [20] J. Sauro. *Measuring usability with the System Usability Scale (SUS)*. Disponible en: <https://measuringu.com/sus/> Acceso en: 11 dic. 2023, 2009.

Anexo A: Módulos del cuestionario basado en aptitudes psicosociales

En este anexo se muestran los distintos módulos del cuestionario basado en aptitudes psicosociales.

DCC | Conformación de Equipos

Cuestionario basado en Aptitudes Psicosociales

0%

Siguiente

Módulo 1: Habilidades Sociales

En las siguientes preguntas indica lo que tú harías en cada una de estas situaciones marcando la casilla del inciso de la oración correspondiente, la respuesta es única. Por favor lee y responde a la respuesta que más acomode a tu forma de pensar y de actuar:

1.- Estás en un acto social donde no conoces a nadie excepto a los anfitriones. Estos están ocupados recibiendo a los invitados. ¿Qué haces?

- Eliges a un invitado que te parezca amable y simpático, te presentas y te quedas con él toda la velada.
- Al haber llegado primero y ser amigo de los anfitriones, te encargas de ir acomodando a los invitados al tiempo que te vas presentando.
- Te presentas a varias personas y conversas un rato con cada una de ellas. Luego es a ti a quien presentan a las demás personas.
- Te sientas en un sillón y no haces nada hasta que llegue un anfitrión y te presente a alguien o se encargue de ti.

2.- Una persona cercana a ti te ha hecho un favor muy importante y estás sorprendido por los buenos resultados. Ahora es el momento de agradecersele...

- Se lo agradeces sinceramente y le haces un regalo a cambio del favor que te ha hecho.
- Se lo agradeces con cumplidos y le dices que vas a hacer algo grande por esta persona cuando tengas una oportunidad.

Figura 7.1: Módulo 1 del cuestionario de aptitudes psicosociales



Figura 7.2: Módulo 2 del cuestionario de aptitudes psicosociales



Figura 7.3: Módulo 3 del cuestionario de aptitudes psicosociales

DCC | Conformación de Equipos

Cuestionario basado en Aptitudes Psicosociales

16%

Anterior Inicio del cuestionario Siguiente

Módulo 4: Trabajo en Equipo

En las siguientes preguntas indica lo que tú harías en cada una de estas situaciones marcando la casilla del inciso de la oración correspondiente, la respuesta es única. Por favor lee y responde a la respuesta que más acomode a tu forma de pensar y de actuar:

1.- Trabajar con personas que tienen puntos de vista distintos al tuyo...

- No te resulta incómodo, aunque preferirías no trabajar con ellas.
- Oír otros puntos de vista enriquece, además de ser un buen complemento.
- Es muy difícil para ti trabajar con ellas.

2.- En el trabajo, cuando no estás de acuerdo con algo...

- Lo dices abiertamente y expones los motivos que te llevan a ello.
- Tienes miedo de que tu opinión resulte ridícula y no sea tenida en cuenta.
- Si no hay más remedio, lo dices pero evitas entrar en discusiones.

3.- ¿Sueles tener en cuenta opiniones y puntos de vista de los demás?

- Te molestan las sugerencias de los demás, pero sueles atenderlas abiertamente.
- Siempre es enriquecedor escuchar otras opiniones.
- Aceptas abiertamente otros puntos de vista, aunque al final decides tú.

Figura 7.4: Módulo 4 del cuestionario de aptitudes psicosociales

DCC | Conformación de Equipos

Cuestionario basado en Aptitudes Psicosociales

24%

Anterior Inicio del cuestionario Siguiente

Módulo 5: Comunicación

En las siguientes preguntas indica lo que tú harías en cada una de estas situaciones marcando con una "X" la casilla del inciso de la oración correspondiente, la respuesta es única. Por favor lee y responde a la respuesta que más acomode a tu forma de pensar y de actuar:

1.- Usted piensa que la gente

- Debe ser como usted.
- Debe buscar consenso.
- Tiene iguales derechos que usted.
- No debe expresar desacuerdos.

2.- Cuando establece una comunicación

- Considera los sentimientos de los demás.
- Nunca contradice a su interlocutor.
- Tiene en mente la negociación.
- Trata de monopolizarla.

3.- Cuando usted se está comunicando

Figura 7.5: Módulo 5 del cuestionario de aptitudes psicosociales

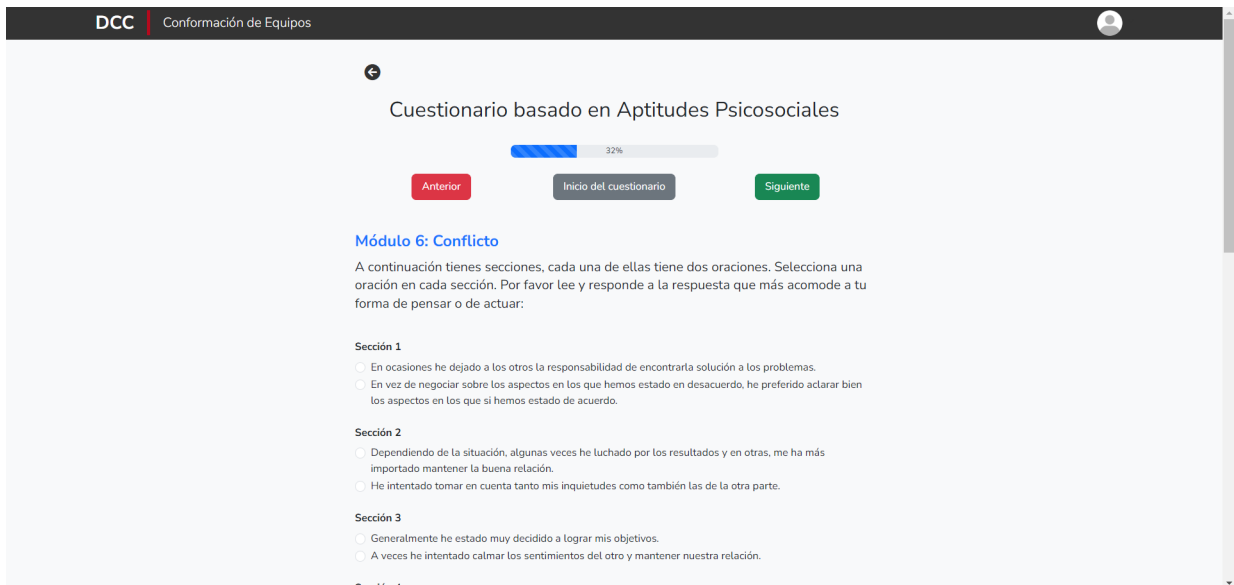


Figura 7.6: Módulo 6 del cuestionario de aptitudes psicosociales



Figura 7.7: Módulo 7 del cuestionario de aptitudes psicosociales



Figura 7.8: Módulo 8 del cuestionario de aptitudes psicosociales



Figura 7.9: Módulo 9 del cuestionario de aptitudes psicosociales

Anexo B: Ejemplo de uso de *models.TextChoices*

En este anexo se muestra un ejemplo de uso de *models.TextChoices* de Django.

```
class Pregunta(models.Model):
    """
    Corresponde a una pregunta de un cuestionario
    """

    class Tipo(models.TextChoices):
        RESPUESTA_UNICA = "U", "Respuesta única"
        RESPUESTA_MULTIPLE = "M", "Respuesta múltiple"
        ASIGNAR_PUNTAJE = "R", "Asignar puntaje"

    class Area(models.TextChoices):
        TECNICA = "T", "Técnica"
        SOCIAL = "S", "Social"
        HABILIDADES_SOCIALES = "HS", "Habilidades Sociales"
        LIDERAZGO = "L", "Liderazgo"
        PERSONALIDAD = "P", "Técnica"
        TRABAJO_EN_EQUIPO = "TE", "Trabajo en Equipo"
        COMUNICACION = "C", "Comunicación"
        CONFLICTO = "CF", "Conflicto"
        CAPACIDAD_DE_GESTION = "CG", "Capacidad de Gestión"
        PLANIFICACION = "PL", "Planificación"
        BELBIN = "B", "Test de Belbin"

    tipo = models.CharField(max_length=1, choices=Tipo.choices, default=Tipo.RESPUESTA_UNICA)
    area = models.CharField(max_length=2, choices=Area.choices, default=Area.TECNICA)
    numero_pregunta = models.PositiveIntegerField(null=True)
    texto = models.CharField(max_length=500)
    respuestas = models.ManyToManyField(PuntajeRespuesta)

    def __str__(self):
        return f"{self.texto} (Área: {self.get_area_display()}, Tipo: {self.get_tipo_display()})"
```

Figura 7.10: Clase “Pregunta” del modelo de datos

Anexo C: Enunciados de la encuesta SUS

En este anexo se listan los enunciados de la encuesta SUS.

1. Pienso que me gustaría usar frecuentemente este sistema.
2. Encontré el sistema innecesariamente complejo. Opino que el sistema fue fácil de usar.
3. Creo que necesitaría ayuda para poder usar este sistema.
4. Considero que las distintas funcionalidades estaban bien integradas.
5. Pienso que había mucha inconsistencia en este sistema.
6. Las personas aprenderían rápidamente cómo utilizar el sistema.
7. Creo que el sistema era muy complicado de usar.
8. Me sentí muy seguro al usar el sistema.
9. Necesité aprender cosas antes de usar cómodamente el sistema.

Anexo D: Enunciados de la encuesta TAM

En este anexo se listan los enunciados de la encuesta TAM.

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