



UNIVERSIDAD FASTA

Ingeniería en Informática – FIM 42 – Proyecto Final

Sprint 8

CommonJobs

(Sistema de Recursos Humanos CommonSense)

Alumnos:

- * Andrés Moschini.
- * Matías José.
- * Juan Diego Raimondi.

Director Funcional: Gabriel Buyatti.

Director Técnico: Ing. Alejandro Fantini.

Auditora: Ing. Ana Haydee Di Iorio.

Cátedra:

- * Profesor titular: AS. Hilario Fernando Schechtel
- * Profesor asociado: Ing. Roberto Giordano Lerena
- * Profesor asociado: Lic. Alejandro Nikolic

Fecha de presentación: 05/11/2012

Tabla de Contenidos

Tabla de Contenidos	1
Planificación de Sprint 8	2
Fecha inicio:	2
Fecha fin:	2
Tareas	2
Product Backlog CommonJobs (Trello)	3
Grupo 1	3
Grupo 2	3
Grupo 3	3
Grupo 4	3
Grupo 5	3
Línea de fin de proyecto	4
Resumen de Horas	6
Horas Comprometidas	6
Horas realmente trabajadas y en qué	6
Retrospectiva	7
Andrés:	7
Matías:	7
JD:	7
Revisión	8
Trabajo completado	8
Pendiente	8
Salida Sprint	8

Planificación de Sprint 8

Fecha inicio:

21 de junio.

Fecha fin:

11 de julio.

Tareas

Los números son los Story Points asignados a cada Tarea/User Story.

- (2) Resaltar términos de búsqueda en la búsqueda de archivos.
- (5) Al buscar archivos, me gustaría poder buscar por patrones del nombre de archivo.
- (5) Mejoras al menú.
- (2) Permitir seleccionar texto en UI de búsqueda de archivos sin lanzar la descarga del archivo.
- (2) Corregir documentación de Implantación referida a grupos.
- (2) Revisar la ocurrencia de archivos sin nombre en la búsqueda.
- (1) Al buscar archivos, me gustaría poder ver solo archivos con contenido de texto (por ejemplo no ignorar las fotos de los empleados).
- (7) Al buscar archivos, me gustaría que el pequeño extracto de texto tenga un formato consistente.
- (20) Empleados: Definir links para vistas (quizá temporales) para poder compartir cierta información.
- (10) Investigar sobre integración con LinkedIn.
- (5) Modificar la convención utilizada en la generación de IDs de los attachments para no colisionar con la convención utilizada por Raven Studio.
- (3) Documentación Fin Sprint 7.
- (7) Agrupar tareas / user stories del product backlog en grupos de 50 story points ordenados por prioridad para tener una mejor idea de que haremos y cuando.
- (1) Pedir que saquen el requerimiento de AD en <https://commonjobs.makingsense.com:8888/documentation>.

Product Backlog CommonJobs (Trello)

Los números son los Story Points asignados a cada Tarea/User Story.

Las tareas / user stories están agrupadas en grupos de 100 story points (aprox.) y ordenados por prioridad.

Grupo 1

- (40) Como usuario del sistema, me gustaría poder leer información de los perfiles de mis contactos en LinkedIn para poder manejar mi base de contactos como postulantes de forma eficiente.

Grupo 2

- (5) Aceptar barra y guión en los campos de fecha.
- (2) Recuperar "UploadAvatar.png".
- (5) Al parecer el reset de CSS no está funcionando correctamente en las páginas de detalles.
- (13) Mantener el historial de cambios de las entidades.
- (15) Applicants postulando online.
- (5) Preparar un entorno para test de integración (utilizarlo en los test unitarios de Visual Studio).
- (13) Preparar un entorno para test de integración (javascript).
- (13) Preparar un entorno para test de integración (base de datos).
- (3) Integrar algún sistema de Log.
- (10) Ordenamiento de applicants.
- (7) Como usuario del sistema, me gustaría poder ver el listado de postulantes a una oferta en particular, de forma de saber cuáles son aquellos que respondieron a ella.

Grupo 3

- (3) Agregar un estado "Dado de baja" para los empleados y por defecto no mostrarlos en la lista de empleados.
- (5) Como usuario del sistema, me gustaría poder compartir resultados del sistema (búsquedas, información de postulantes, etc.).
- (10) Como usuario del sistema, desearía que las ofertas de trabajo puedan generarse en base a pedidos realizados en el proyecto de Jira "Sales Pipeline", para así trabajar más eficientemente (Investigación y diseño).
- (10) Como usuario del sistema, desearía que una oferta pueda publicarse en LinkedIn, para obtener mayor alcance en su distribución (Investigación y diseño).
- (2) Al hacer clic en "Agregar nuevo" (empleado o postulante) y se cancelan los cambios, aparece un empleado con datos en blanco.
- (9) Como usuario del sistema, me gustaría poder definir carpetas en las que se sitúa cada postulante, de forma de poder definir un flujo de trabajo personalizado y alterable en el futuro. (Ejemplo: "por entrevistar", "esperando datos", etc.).
- (7) Como usuario del sistema, me gustaría poder clasificar los postulantes mediante tags arbitrarios.
- (2) Indicar claramente cuando una sección está colapsada o expandida.
- (5) Previsualización de formato en notas.
- (2) Como usuario del sistema, me gustaría que el sistema me permita acceder a los perfiles en distintos sitios del postulante, como para poder analizar yo mismo la información allí disponible.
- (9) Como manager de RRHH, me gustaría poder ver estadísticas sobre el banco de postulantes, para así poder tomar decisiones y reportar a niveles superiores según nuestro banco de postulantes.
- (10) Como usuario del sistema, desearía que una oferta pueda publicarse en Facebook, para obtener mayor alcance en su distribución (Investigación y diseño).
- (10) Como usuario del sistema, me gustaría poder obtener información de postulantes desde emails, para poder facilitar el proceso de importación de datos (Análisis y Diseño).

Grupo 4

- (40) Como usuario del sistema, desearía que las ofertas de trabajo puedan generarse en base a pedidos realizados en el proyecto de Jira "Sales Pipeline", para así trabajar más eficientemente (Implementación).
- (40) Como usuario del sistema, desearía que una oferta pueda publicarse en LinkedIn, para obtener mayor alcance en su distribución (Implementación).
- (7) Hacer que los campos "acepten" el cambio una vez que se hace clic fuera de ellos.
- (10) Crear estilos / presentación para la pantalla posterior al login.

Grupo 5

- (50) Como usuario del sistema, me gustaría poder obtener información de postulantes desde emails, para poder facilitar el proceso de importación de datos (Implementación).
- (40) Como usuario del sistema, desearía que una oferta pueda publicarse en Facebook, para obtener mayor alcance en su distribución (Implementación).
- (7) Como usuario del sistema, poder clasificar postulantes por sus perfiles.

Línea de fin de proyecto

Consideramos que las tareas que se indican a continuación no forman parte del criterio de fin de proyecto

- (4) Me gustaría ver y poder eliminar definitivamente los archivos huérfanos.
- (3) Prevenir inyección de HTML y de JavaScript en el listado de adjuntos.
- (20) Como usuario del sistema, me gustaría que el sistema automáticamente determinara según las calificaciones de un postulante, qué requerimientos cumple y qué nivel de seniority posee él, para poder tomar decisiones eficientemente.
- (50) Como administrador del sistema, me gustaría poder alterar los criterios para definir el seniority de un postulante, creando y quitando niveles, para poder ajustarlo a los criterios de la empresa.
- (20) Como usuario del sistema, me gustaría que el sistema me permita ver cuál es el postulante más apropiado según las calificaciones cumplidas para un puesto en particular, para poder tomar decisiones acertadas.
- (20) Mostrar preview relevante a la búsqueda en la búsqueda de archivos.
- (5) Integración postulantes-Doppler.
- (15) Búsqueda avanzada con comando.
- (3) Arreglar overlay de imagen de avatar.
- (1) Como manager de RRHH me gustaría poder asignar un puntaje (rating) a cada postulante, de forma de poder ver de forma rápida cuál es el más apropiado para contratar.
- (50) Como entrevistador, me gustaría poder asignar una o más fechas de entrevista a cada postulante, también pudiendo invitar a empleados de la empresa, de forma de poder mantener un registro de las entrevistas que cada postulante tiene.
- (50) Como usuario del sistema, me gustaría poder obtener información de postulantes desde su CV, para poder facilitar el proceso de importación de datos.
- (20) Como usuario, me gustaría poder ver filtros sugeridos por el sistema según los datos disponibles en el actual banco de postulantes, para así poder navegar por la información disponible de una forma más eficiente.
- (20) Configurar octopusdeploy para los deploys.
- (2) Quitar dummy text de la documentación.
- (8) Modificar el image cropping para que solo recorte el thumbnail.
- (10) Permitir resaltados por cada usuario.
- (5) Error cropping fotos con IE (bug).
- (3) ¿Quién refirió a un postulante?
- (5) Como usuario del sistema, me gustaría poder alterar qué información veré de los postulantes al momento de efectuar una búsqueda, de forma de poder ver rápidamente la información más relevante.
- (9) Como usuario, me gustaría contar con la información cargada hasta el momento sobre los postulantes (menos de 300) en el proyecto actual Jira Recruit para no perder el trabajo realizado hasta el momento en las entrevistas y acceder a ella de forma sencilla.
- (50) Como usuario, me gustaría poder definir filtros personalizados de postulantes, para así poder realizar mis búsquedas en el banco de postulantes de forma más eficiente.
- (9) Como entrevistador, me gustaría poder ver un listado de entrevistas a futuro, de forma de mantener organizados los tiempos de entrevistas a futuro y hacer buen uso de mi tiempo laboral.
- (9) Como soporte a futuro del sistema me gustaría poder exportar e importar información del sistema poder utilizarla en otros sistemas.
- (20) Como usuario del sistema, me gustaría que el sistema muestre sugerencias de postulantes apropiados basados en el rating que he asignado a postulantes similares en el pasado, para así poder realizar mis decisiones de forma más eficiente.
- (1) Como usuario del sistema, me gustaría poder quitar la información de un postulante, para poder reducir la cantidad de datos con la que trabajo.
- (1) Como manager de RRHH, me gustaría poder convertir un postulante en un empleado, para diferenciarlo de aquellos que habiéndose postulado a una oferta no consiguieron el puesto.
- (1) Como manager de RRHH, me gustaría poder ver un listado de postulantes que luego se convirtieron en empleados de la empresa, para así poder evaluar las eficiencias en nuestras búsquedas.
- (50) Como usuario del sistema, me gustaría poder obtener información de postulantes desde ZonaJobs, para poder facilitar el proceso de importación de datos.
- (50) Como usuario del sistema, desearía que una oferta pueda publicarse en Boomerang, para obtener mayor alcance en su distribución.
- (3) Indexar la metadata de los attachments.
- (50) Como usuario del sistema, desearía que una oferta pueda publicarse en BaseJobs, para obtener mayor alcance en su distribución.
- (50) Como usuario del sistema, desearía que una oferta pueda publicarse en BuscoJobs, para obtener mayor alcance en su

distribución.

- (9) Como soporte a futuro del sistema, desearía que expusiera una API o un sistema de plugins de publicación de ofertas para poder ampliar la cantidad de sitios o formatos soportados.
- (50) Como usuario del sistema, desearía que una oferta pueda publicarse en ZonaJobs, para obtener mayor alcance en su distribución.
- (9) Como usuario del sistema, desearía que una oferta pueda publicarse en Twitter, para obtener mayor alcance en su distribución.
- (50) Como usuario del sistema, me gustaría poder obtener información de postulantes desde Boomerang, para poder facilitar el proceso de importación de datos.
- (20) Como usuario del sistema, me gustaría poder guardar definiciones de vistas de postulantes (información visible rápidamente para un postulante en los resultados de una búsqueda) para así poder reutilizarlas de forma más eficiente.
- (50) Como usuario del sistema, me gustaría que la información de un postulante se pueda actualizar desde ZonaJobs, para poder facilitar el proceso de manutención de datos.
- (9) Como auditor del sistema, me gustaría poder ver los cambios que se realizaron a un postulante, para así observar el progreso de un postulante.
- (50) Como usuario del sistema, desearía que el sistema me notifique de actualizaciones realizadas al proyecto de Jira de "Sales Pipeline", para así trabajar con la información más actual.
- (50) Como usuario del sistema, desearía que los cambios efectuados en una oferta de trabajo se actualicen en el proyecto de Jira "Sales Pipeline", para que el resto de la empresa pueda estar al tanto a través de dicho proyecto sobre el avance de esa búsqueda laboral.
- (50) Como usuario del sistema, me gustaría que la información de un postulante se mantenga actualizada según su información obtenida desde un email, para poder facilitar el proceso de manutención de datos.
- (50) Como usuario del sistema, me gustaría que la información de un postulante se pueda actualizar desde Boomerang, para poder facilitar el proceso de manutención de datos.
- (50) Como soporte a futuro del sistema me gustaría que el sistema me de la opción de migrar la nueva información de los postulantes cargados en el sistema actual al nuevo de forma sencilla para no tener que cargar la información dos veces en las primeras etapas de la migración del sistema.
- (9) Como soporte a futuro del sistema, me gustaría que el sistema dispusiera de una API o sistema de plugins de actualización de datos para que pueda agregar soporte en otros sitios de empleos.
- (50) Como usuario del sistema, me gustaría poder obtener información de postulantes desde BuscoJobs, para poder facilitar el proceso de importación de datos.
- (50) Como usuario del sistema, me gustaría poder obtener información de postulantes desde BaseJobs, para poder facilitar el proceso de importación de datos.
- (50) Como usuario del sistema, me gustaría que la información de un postulante se pueda actualizar desde Basejobs, para poder facilitar el proceso de manutención de datos.
- (50) Como usuario del sistema, me gustaría que la información de un postulante se pueda actualizar desde BuscoJobs, para poder facilitar el proceso de manutención de datos.
- (1) Como usuario, me gustaría poder quitar los datos extraídos en las entrevistas a los postulantes en caso que sea necesario para facilitar la visualización de los datos relevantes.

Resumen de Horas

Horas Comprometidas

Se incluyen las reuniones de planificación, revisión y retrospectiva

- JD 30:00 horas
- Matías 25:00 horas
- Andrés 30:00 horas

Horas realmente trabajadas y en qué

JD:

- Aprendizaje / Investigación: 10.5 horas
- Desarrollo: 15.5 horas
- Planificación: 2 horas

Total: 28 horas

Matías:

- Documentación: 4 horas
- Aprendizaje / Investigación: 10 horas
- Planificación: 2 horas
- Desarrollo: 3 horas

Total: 19 horas

Andrés:

- Planificación: 4:45 horas
- Análisis: 2:15 horas
- Formalidades: 0:30 horas
- Aprendizaje / investigación: 4:30 horas
- Diseño: 2:30 horas
- Desarrollo: 12:30 horas

Total: 27 horas

Retrospectiva

Andrés:

- Empezamos antes tareas más complejas, pudimos corregir cosas al final.

Matías:

- Complicaciones de tiempo en este sprint, vamos a coordinar esta situaciones para que los procesos se bloqueen con él.

JD:

- Empezamos con las tareas más difíciles, nos permitió organizar mejor el tiempo
- Un poco más de motivación, algo bueno, la empresa está más involucrada.
- Riesgo de bloquearse por desconocimiento en RavenDB, comenzará a meterse del lado del backend con tareas más simples que involucren a esta tecnología.

Revisión

Trabajo completado

Las siguientes tareas surgieron o se desprendieron de otra tarea de lo planificado para este sprint.

- Corregir documentación de Implantación referida a grupos.
- Modificar la convención utilizada en la generación de IDs de los attachments para no colisionar con la convención utilizada por Raven Studio.
- Agrupar tareas / user stories del product backlog en grupos de 100 story points ordenados por prioridad para tener una mejor idea de que haremos y cuando.
- Investigar sobre integración con LinkedIn.
- Documentación Fin Sprint 7.
- Crear backend para compartir vistas de Applicants.
- Actualizar RavenDb server al build 960.
- Revisar la ocurrencia de archivos sin nombre en la búsqueda.
- Applicants: Definir links para vistas (quizá temporales) para poder compartir cierta información.
- Al buscar archivos, me gustaría poder ver solo archivos con contenido de text (por ejemplo no ignorar las fotos de los empleados).
- Permitir seleccionar texto en UI de búsqueda de archivos sin lanzar la descarga del archivo (bug).
- Mejoras al menú.
- Bug con las fechas en los links compartidos (bug).

Pendiente

Las tareas pendientes surgieron o se desprendieron de otra tarea de este Sprint. Estas tareas se agregan al Product Backlog o a la planificación del Sprint siguiente.

- Mejorar el pequeño extracto de texto al buscar archivos en cuanto a consistencia.
- Reemplazar el uso de StartsWith por Search con AllowWildcards en la búsqueda por nombre de archivo. Para evitar problemas en futuros updates de RavenDB ya que ahora estamos aprovechando un bug.
- Resaltar términos de búsqueda en la búsqueda de archivos.
- Revisar Product Backlog con Gabriel.

Salida Sprint

Como salida del Sprint 8 mostramos el estado del proyecto al día de la fecha en nuestro [Repositorio de Git Hub](#).