

GESTIÓN DE TICKETS CON JIRA

Eric Quintero Moreno

2º DAW | 21/04/2023



ÍNDICE

Contenido

1. ¿QUÉ ES JIRA Y PARA QUE SIRVE?	3
2. BREVE INTRODUCCIÓN A JIRA	4
2.1. ¿Qué es una instancia?	4
2.2. ¿Qué es un proyecto?	5
2.3. Todo lo relacionado dentro del proyecto	6
3. COMO USAR JIRA	8
3.1. Acceder a Jira	8
3.2. Diferencias entre proyectos	9
3.3. Como crear incidencias	10
4. JUSTIFICACIÓN Y OBJETIVOS DEL PROYECTO	11
5. METODOLOGÍAS USADAS	12
6. COMPONENTES	13
7. PROBLEMAS/DIFICULTADES ENCONTRADAS	15
8. ESTRUCTURA DEL PROYECTO	16
9. LÍNEAS FUTURAS	17
10. CONCLUSIÓN	18
11. ANEXOS	19
11.1. ANEXO 1: PROYECTO JIRA SOFTWARE: RRHH	19
11.2. ANEXO 2: PROYECTO JIRA SERVICE MANAGMENT	39
11.3. ANEXO 3: PROYECTO JIRA SOFTWARE: DESARROLLO	57
12. BIBLIOGRAFÍA	66

1. ¿QUÉ ES JIRA Y PARA QUE SIRVE?

Jira es una herramienta de gestión de proyectos desarrollada por Atlassian. Se utiliza en el desarrollo empresarial y de software para la planificación, el seguimiento y la gestión de proyectos.

Jira permite crear proyectos y asignar tareas a los miembros del equipo, así como hacer un seguimiento del progreso de cada tarea. También proporciona una amplia gama de herramientas de seguimiento, informes y análisis para ayudar a los equipos a mantenerse al día y cumplir con los plazos.

Jira se usa comúnmente para la gestión de problemas y errores, la planificación de versiones y la gestión de tareas de desarrollo en el desarrollo de software. Sin embargo, su versatilidad permite su uso en otros contextos, como la gestión de proyectos de marketing o la gestión de tareas de recursos humanos.



Figura 1: Logo de Jira

2. BREVE INTRODUCCIÓN A JIRA

2.1. ¿Qué es una instancia?

En Jira, una instancia es una instalación de la herramienta en un servidor o en la nube. Cada instancia de Jira tiene su propia base de datos y configuración única que una empresa puede usar para administrar sus proyectos y tareas.

Cada instancia de Jira tiene sus propios usuarios y proyectos. Los usuarios pueden asignarse a proyectos específicos y recibir diferentes niveles de permisos para ver, editar y cambiar proyectos.

Las instancias de Jira también se pueden personalizar para satisfacer las necesidades específicas de una organización, incluida la creación de campos, flujos de trabajo y dashboards personalizados.

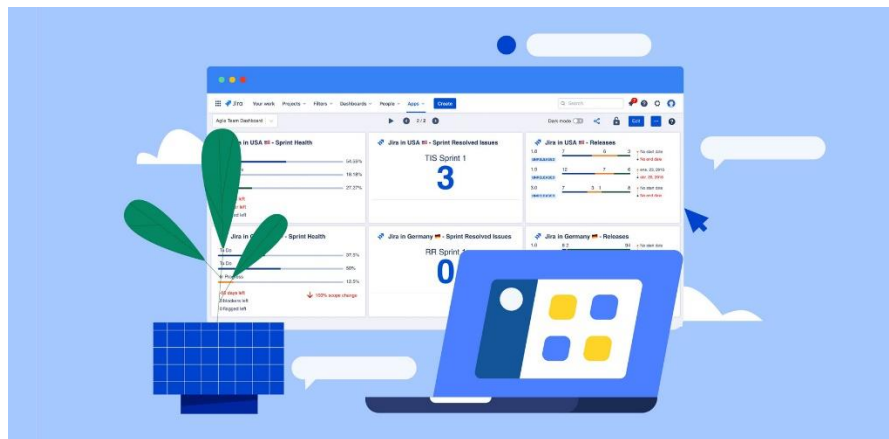


Figura 2: Ejemplo de instancia

2.2. ¿Qué es un proyecto?

Un proyecto es un grupo de tareas relacionadas realizadas para lograr un objetivo común. Los proyectos en Jira se pueden usar para gestionar proyectos de software, proyectos de marketing, proyectos de recursos humanos y cualquier otra tarea o proyecto que requiera un seguimiento y una gestión detallados.

Cada proyecto tiene su propio espacio de trabajo dentro de la instancia de Jira, que se puede personalizar para satisfacer las necesidades específicas del proyecto. Los proyectos en Jira pueden contener varias tareas y problemas, y se pueden configurar flujos de trabajo específicos para realizar un seguimiento de cada tarea.

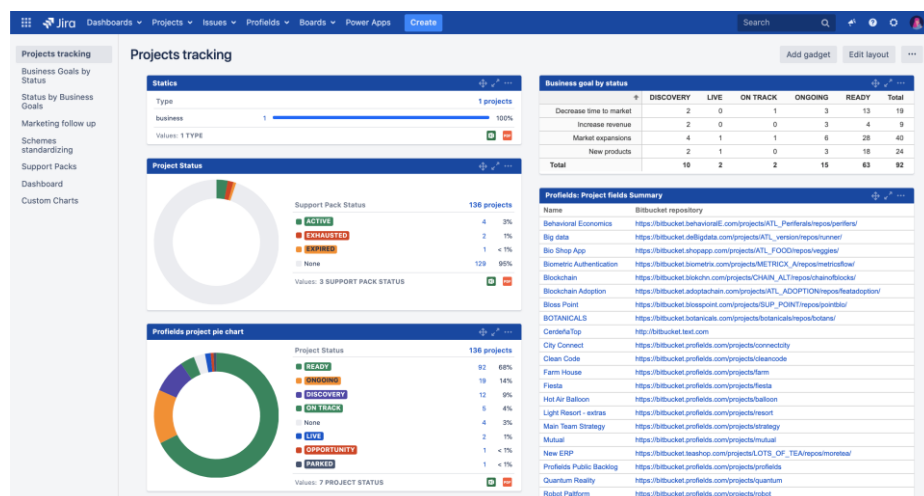


Figura 3: Ejemplo de proyecto

2.3. *Todo lo relacionado dentro del proyecto*

En un proyecto de Jira hay varios tipos de cosas para tener en cuenta. Aquí mencionaré algunos de ellos que voy a mencionar posteriormente:

- **Incidencias:** Una incidencia se refiere a cualquier problema, error o fallo que se encuentra durante el desarrollo de un proyecto. Las incidencias se registran en Jira para que los miembros del equipo puedan hacer un seguimiento de ellas y trabajar en su resolución.

Cada incidencia en Jira se registra como un ticket individual, con su propio número de identificación y detalles específicos sobre el problema.

- **Solicitudes:** Las solicitudes están dentro de las incidencias, dentro de ese tipo de incidencia, hay un tipo de solicitud. Como, por ejemplo, si el usuario quiere hacer una incidencia de tipo usuario, los tipos de solicitudes podrán ser dar de alta al usuario o darlo de baja.

- **Roles y grupos:** Los roles son funciones predefinidas que determinan las responsabilidades de los usuarios en un proyecto.

Los grupos son conjuntos de usuarios que se organizan juntos por su rol o función en un proyecto.

- **Workflows:** Los workflows son un conjunto de pasos predefinidos que se utilizan para gestionar y hacer un seguimiento de las incidencias o en un proyecto. El flujo de trabajo se utiliza para definir los estados por los que pasan las incidencias a medida que se completan.

- **Dashboards:** Los dashboards son una página personalizable que muestra información importante sobre el estado y el progreso de un proyecto o instancia de Jira. Los dashboards se pueden utilizar para visualizar estadísticas y gráficos en tiempo real.
- **Automation:** Es una herramienta que permite automatizar tareas y flujos de trabajo en un proyecto. Se utiliza para simplificar procesos y reducir la cantidad de trabajo manual que los miembros del equipo deben realizar.
- **SLAs:** Los SLAs se utilizan para definir y medir los niveles de servicio que se deben cumplir para satisfacer las expectativas de los clientes en términos de calidad y rapidez de la entrega de un servicio.
- **Assets:** Estas herramientas permiten a los equipos mantener registros completos y actualizados de todos los activos que forman parte de un proyecto o sistema. Los assets vendría a ser la propia base de datos que utiliza Jira.
- **Escalados:** un escalado básicamente es la facilidad para que, al crear una incidencia en un proyecto, se pueda clonar en otro proyecto para que ellos trabajen en esa incidencia.

3. **COMO USAR JIRA**

3.1. **Acceder a Jira**

Para acceder a Jira simplemente hay que hacer clic en los enlaces que se proporcionarán más adelante y registrarse. Una vez registrados, podréis crear incidencias desde el portal o desde la propia aplicación.

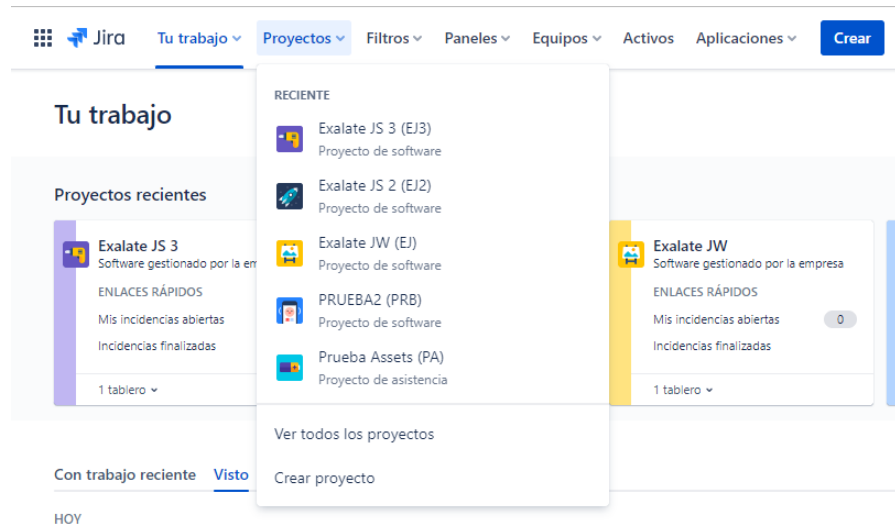


Figura 4: Acceder a los proyectos

Una vez dentro desde Proyectos se podrá acceder al proyecto deseado, en este caso, solo habrá un proyecto por instancia.

Para este proyecto se facilitan dos cuentas:

Directivo y Senior:

cuentadeejemplojira1@gmail.com

Hola-Adios1234

Técnico y Junior:

cuentadeejemplojira2@gmail.com

Hola-Adios1234

3.2. *Diferencias entre proyectos*

En las instancias existen tres tipos de proyectos, cada proyecto está en su instancia. Un proyecto es de Jira Service Management y los otros dos de Jira Software.

El proyecto de Jira Service Management es una herramienta para administrar y resolver problemas de soporte y servicio al cliente de manera eficiente y efectiva.

Por otra parte, los proyectos de Jira Software sirven para administrar equipos, permitiéndolos trabajar de manera colaborativa, planificar y hacer seguimiento de sus tareas y proyectos de manera eficiente y efectiva.

Dentro de los proyectos de Jira Software existen dos tipos de tableros. El tablero Kanban, es como un Trello, simplemente sirve para ordenar las tareas y saber si están hechas, en progreso o acabadas. El otro tipo de tablero es el Scrum, que sirve para los equipos de desarrollo, ya que las tareas se organizan por sprints de x semanas.



Figura 5: Logo JSM



Figura 6: Logo JS

3.3. Como crear incidencias

Existen dos maneras de crear incidencias, desde dentro del tablero o desde el portal.

Para crearlas desde el tablero, hay que entrar en el proyecto y darle al botón de crear, tal y como se muestra en la imagen. Rellenas los campos y la creas.

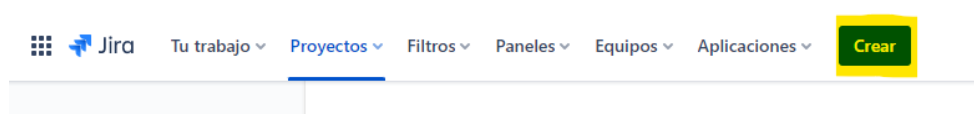


Figura 7: Crear incidencia desde proyecto

Para crear incidencias desde el portal, hace falta la URL, una vez se te tenga la URL, los pasos son muy sencillos.

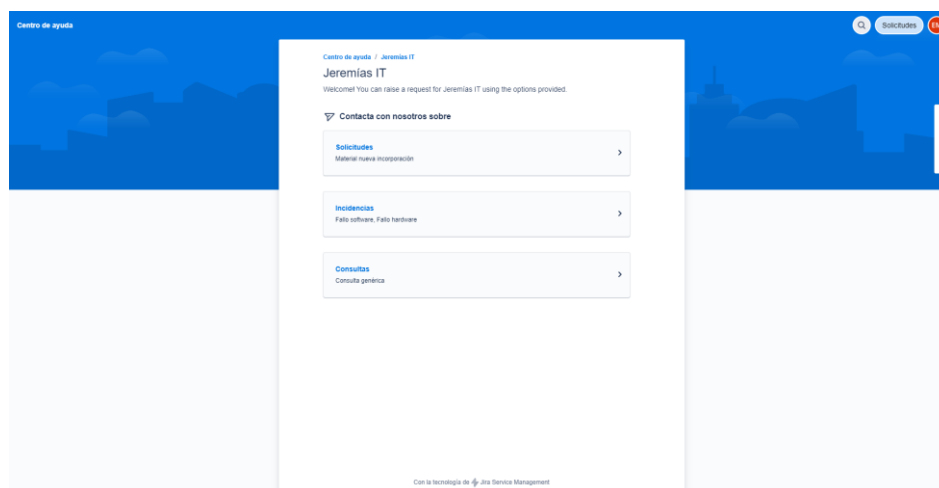


Figura 8: Crear incidencia desde el portal

4. JUSTIFICACIÓN Y OBJETIVOS DEL PROYECTO

Este proyecto ha surgido con la idea de que en la empresa donde estoy trabajando, utilizaban esta herramienta y me dijeron que podía usarla para hacer otros proyectos personales. De esta forma, pensé que era una buena idea hacer mi proyecto en base a esta herramienta y potenciar mis conocimientos en ella para ser más productivo en la empresa.

El proyecto va dirigido a una empresa ficticia llamada Jeremías, donde hay 3 departamentos, el primero sería Recursos Humanos, el otro IT y el último el de Desarrollo.

La finalidad es para que esta empresa pueda organizarse de una manera sencilla y eficaz teniendo un control constante de las tareas que tienen por hacer.

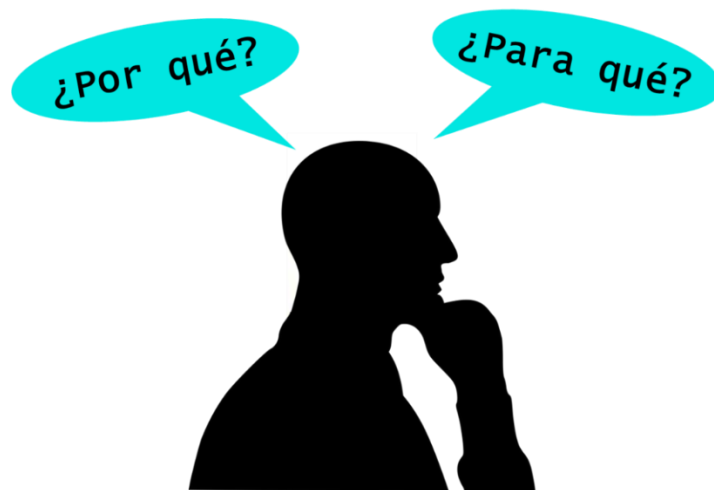


Figura 9: Justificación y Objetivos

5. METODOLOGÍAS USADAS

Se han usado distintas metodologías para este proyecto. Sin ellas hubiera sido más difícil el trabajo ya que no habría un orden y un control tan claro a la hora de ir desarrollando el proyecto. A continuación, detallo algunas de ellas:

- **Scrum:** esta metodología a mi parecer ha sido la más útil ya que teníamos sprints de 2 o 3 semanas donde teníamos reunión con la profesora para ir resolviendo dudas e ir enseñando nuestros proyectos. También hicimos una entrega intermedia donde enviamos lo que teníamos para que no nos quedásemos dormidos.
- **Kanban:** la metodología Kanban también se ha usado con la aplicación Trello. Básicamente se ha ido haciendo un registro de trabajo donde en un tablero iba poniendo en que estado estaban las tareas.

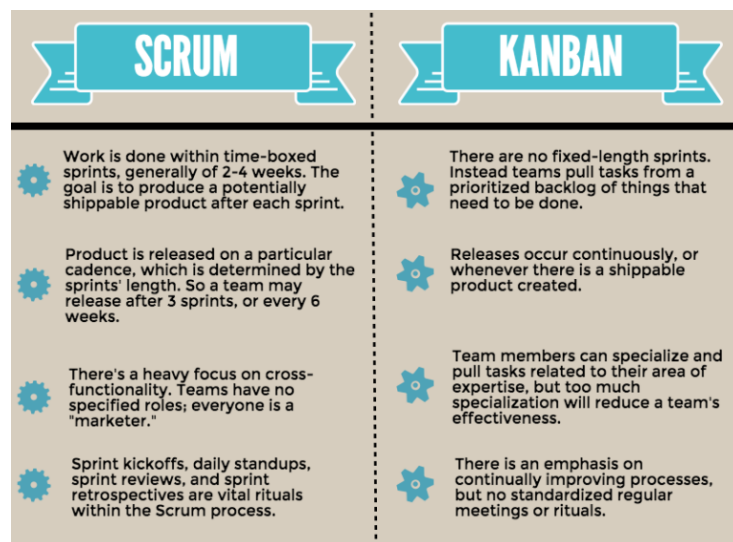


Figura 10: Scrum y Kanban

6. COMPONENTES

Se han usado varias herramientas para el desarrollo de este proyecto. La primera y más importante ha sido Jira, ya que sin ella no podría haber desarrollado este proyecto.

Todo el proyecto se ha desarrollado con Jira y dentro de Jira se han utilizado las herramientas Jira Software y Jira Service Managment. Dentro de Jira se han utilizado otras aplicaciones que voy a mencionar a continuación:

- **Exalate:** es una aplicación que permite sincronizar información entre diferentes instancias de Jira Cloud, lo que significa que puedes compartir información de tareas. Esto puede ayudar a mejorar la colaboración, la eficiencia y la visibilidad en los procesos de trabajo, y asegurar que todos los miembros del equipo estén en la misma página.
- **ScriptRunner:** permite automatizar tareas y personalizar tu flujo de trabajo en Jira. Se pueden crear scripts para automatizar tareas repetitivas, como actualizar campos de tareas, enviar notificaciones o incluso integrar Jira con otras herramientas de software.
- **Microsoft365 for Jira:** sirve para poder vincular tus cuentas de Jira y Microsoft para trabajar con las herramientas de Microsoft. Desde dicha aplicación puedes, por ejemplo, crear incidencias desde Outlook, crear conversaciones de Teams...
- **GitHub:** es como una nube para guardar todos tus proyectos de desarrollo.
- **Trello:** sirve para poder organizar tus tareas de forma sencilla y práctica.

Durante todo el desarrollo de la aplicación se ha utilizado el lenguaje de programación Groovy, que es el lenguaje de ScriptRunner.

El proyecto no tiene Mockup ni guía de estilos, ya que no se puede cambiar su apariencia por lo tanto no iban a ver cambios en sus estilos.



Figura 11: Logo de Jira



Figura 12: Logo de Exalate



Figura 13: Logo de GitHub



Figura 14: Logo de ScriptRunner



Figura 15: Logo de Microsoft365 for Jira



Figura 16: Logo de Trello

7. PROBLEMAS/DIFICULTADES ENCONTRADOS

Ha habido bastantes problemas y dificultades a lo largo de todo el proyecto debido a que ha sido algo nuevo para mí. Iba aprendiendo a medida que iba haciendo el proyecto y lo que en un principio parecía sencillo, resultó siendo en algunos casos un quebradero de cabeza ya que tiene muchas cosas que configurar.

Otro de los problemas más comunes que he tenido ha sido con el scripting, ya que para todo se accede con API y he tenido que ir documentándome sobre como funciona la API. A decir verdad, creo que es la API más difícil que he encontrado hasta la fecha ya que tiene cientos de maneras de acceder y hacer una cosa.

Esos son los dos problemas más grandes que he tenido y que vale la pena remarcar.



Figura 17: Problemas y dificultades

8. ESTRUCTURA DEL PROYECTO

La estructura del proyecto consta de tres instancias, cada instancia tiene 1 proyecto asignado a dicha instancia.

El primer proyecto es un Jira Software para el departamento de recursos humanos, donde utilizan un tablero Kanban para organizarse el trabajo y crear incidencias. Dentro del proyecto existen varios tipos de incidencias que ya mencionaremos posteriormente.

Por otro lado, el otro proyecto de la otra instancia es un Jira Service Managment que está ligado al departamento de IT. Ellos se encargarán de resolver todas las incidencias que tenga recursos humanos y el resto de la empresa. Para ello, utilizarán el portal ya mencionado anteriormente.

Por último, el otro proyecto es un Jira Software para el departamento de desarrollo, donde utilizan un tablero Scrum poder organizarse los sprints y tener un control de las tareas que tienen.



Figura 18: Imagen de estructura

9. LÍNEAS FUTURAS

Como líneas futuras de trabajo estaría bien implementar todos los demás departamentos que pueda tener la empresa e ir expandiendo un poco más para que no pueda faltar nada en la aplicación.

También se podría añadir todas las demandas que el cliente nos pudiera pedir más adelante y también ir ampliando toda la base de datos.

También estaría bien implementar otras aplicaciones para facilitar el trabajo de los equipos, como, por ejemplo, implementar GitHub con el equipo de desarrollo para que de esta forma no tengan que ir haciendo los cambios desde fuera de la aplicación y puedan hacerlo desde dentro.

Por último, haría falta hacer más pruebas con el uso de XRay para poder ver si hay algún que otro fallo en la aplicación, ya que el uso de XRay es bastante complejo pero funcional a la hora de hacer pruebas en la aplicación.



Figura 19: Líneas futuras de trabajo

10. CONCLUSIÓN

Ha sido un proyecto que he disfrutado mucho haciéndolo ya que me parece una herramienta muy versátil y útil. Todo ha sido un reto para mi ya que no había usado nada parecido antes. El lenguaje era diferente, la base de datos no era una base de datos sino, un esquema con objetos, y las APIs eran verdaderamente complicadas.

Pero al final, he conseguido sacar el proyecto adelante y creo que se ha quedado un buen proyecto que incluso podría servir para alguna empresa pequeña.



Figura 20: Conclusión

11. ANEXOS

11.1. ANEXO 1: PROYECTO JIRA SOFTWARE: RRHH

a. Tipos de incidencia

Tipo de incidencia	Campos
Baja Laboral	Resumen* Tipo de Incidencia* Responsable Usuario → Assets*
Obtención de Nómina	Resumen* Tipo de Incidencia* Responsable Usuario → Assets*
Alta Laboral	Resumen* Tipo de Incidencia* Responsable Nombre* Apellido 1* Apellido 2* Dirección* Departamento* Nómina* Puesto de Trabajo*
Tarea/Otros	Resumen* Tipo de Incidencia* Adjunto Descripción Informador Responsable Prioridad

	Etickettas
Añadir Ciudad	Resumen* Tipo de Incidencia* Responsable Nombre*
Añadir Departamento	Resumen* Tipo de Incidencia* Responsable Nombre*

Tipos de incidencias

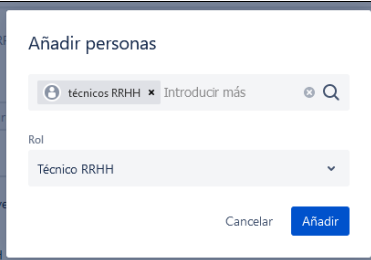
Nota: Aquí se ven los tipos de incidencias

b. Roles y grupos

El proyecto tiene 2 roles asociados y 2 grupos creados. Un grupo está asociado a los directivos y el otro a los técnicos de recursos humanos. Los usuarios estarán en un grupo ya que es más práctico a la hora de dar permisos. Los roles también son dos y se dividen de la misma manera.

Los directivos se encargarán de administrar todo el proyecto y comprobar que todo funcione correctamente, se podría decir que son los jefes de proyecto.

Mientras tanto, los técnicos se encargarán de resolver todas las incidencias.

Rol	Grupo Correspondiente	Asignación
Directivo RRHH	Directivos RRHH	 <p><i>Figura 21: Asignando el rol al grupo</i></p>
Técnico RRHH	Técnicos RRHH	 <p><i>Figura 22: Asignando el rol al grupo</i></p>

Grupos y Roles

Nota: Aquí se ven los roles y grupos que existen

c. Workflows

Los workflows se han asignado dependiendo del tipo de incidencia, en la tabla de más adelante se puede ver que workflows están asociados a que incidencia.

En este proyecto se han creado tres ya que dependiendo de la tarea sigue un flujo de trabajo u otro.

Flujo de trabajo	Tipos de incidencia	Acciones
Alta y Baja Laboral (Ver como texto / diagrama)	<div><div>Alta Laboral</div><div>Añadir Ciudad</div><div>Añadir Departamento</div><div>Baja Laboral (Asignar)</div></div>	<div><div></div><div></div></div>
Obtencion de nomina (Ver como texto / diagrama)	<div><div>Obtención de Nómina (Asignar)</div></div>	<div><div></div><div></div></div>
Tarea y Error (Ver como texto / diagrama)	<div><div>Error</div><div>Tarea (Asignar)</div></div>	<div><div></div><div></div></div>

Figura 23: Tipos de incidencias con sus workflows

El primero es el workflow asociado a las incidencias de tipo “**Alta Laboral**”, “**Baja Laboral**”, “**Añadir Ciudad**” y “**Añadir Departamento**”. Este workflow tiene tres estados, “**Esperando Validación**” ya que, al tratarse de datos de los usuarios, los directivos tendrán que validar si es así o no. Una vez validada, pasa al proceso “**Validado**”, donde se guardarán los datos en la base de datos o se quitarán de la base de datos. Una vez sea terminada o no se haya aprobado, la incidencia pasará al estado “**Cerrada**”.



Figura 24: Workflow de Alta Laboral y Baja Laboral

Siguiendo con los workflows, otro de ellos es el de “**Obtención de nómina**”. Este es sencillo y simplemente cuando se crea la incidencia se queda en “Abierto”, cuando se ponen a trabajar en ello pasa a “En Progreso” o en caso de que haya algún problema y no se pueda seguir se pondrá en “Pendiente”, una vez finalizada pasará al estado “Cerrado”.



Figura 25: Workflow de Obtención de Nómina

El último, es para el tipo de incidencia “**Tarea/Otros**”, que es un workflow muy sencillo. Cuando se cree la incidencia estará en el estado “Abierto”, una vez se pongan a trabajar en ello estará “En

progreso", cuando se termine estará en "Resuelta" y, por último, cuando se revise, entrará en el estado "Cerrada".

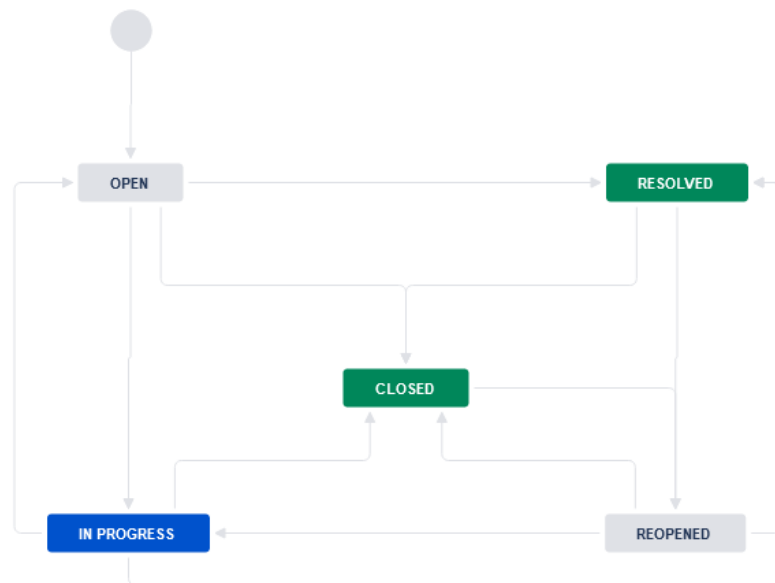


Figura 26: Workflow de Tareas/Otros

d. Dashboard

El dashboard contiene varios gráficos para que los directivos puedan ver la información de su equipo y así poder distribuir mejor todas las tareas.

El primer gráfico es el de incidencias en curso para ver cuantas incidencias hay en el estado **"En curso"** actualmente. El gráfico tiempo promedio en estado nos muestra el tiempo promedio que se tarda en cambiar el estado de las tareas.

Alguno más como el gráfico de creados vs resueltos, que sirve para ver si hay más incidencias creadas o resueltas. Por último, está el gráfico de estadísticas de incidencias, que sirve para ver en qué estado están la mayoría de las incidencias.

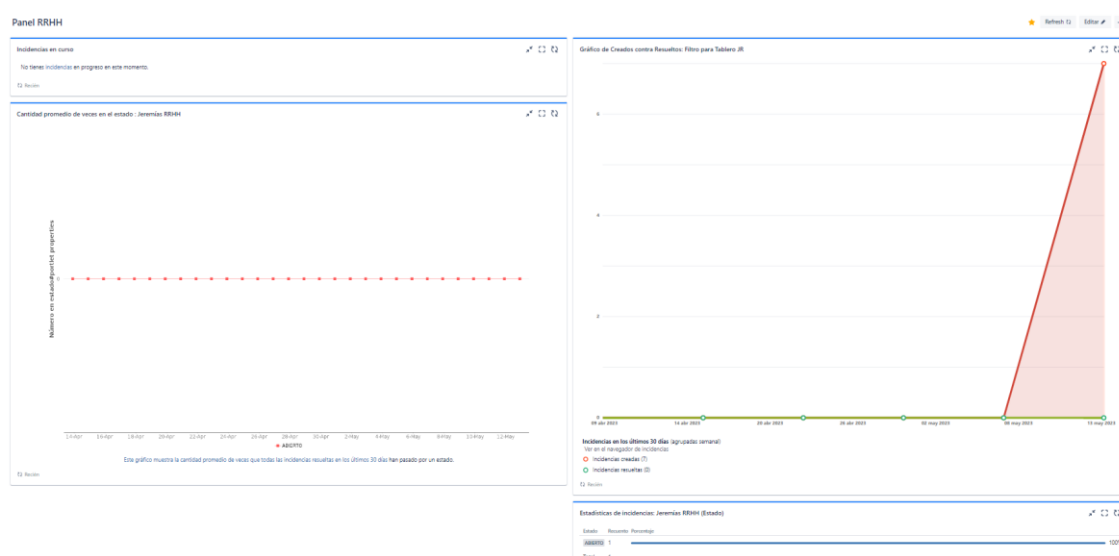


Figura 27: Dashboard con los gráficos

e. Assets

En el proyecto se utilizan los assets para dar de alta o baja a las personas de la empresa. Se utilizan tres objetos, ciudades, departamentos y empleados. Un empleado está ligado a un departamento y un departamento está ligado a una ciudad.

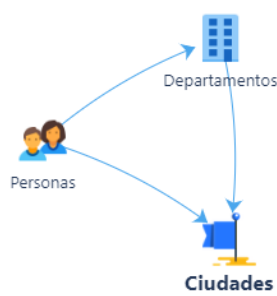


Figura 28: Relación de las entidades

Ciudades

Buscar en todos los esquemas Crear objeto ***

ID	Nombre	Descripción	Tipo	Valor de tipo	Valor adicional
20	Key		Predeterminada	Texto	
21	Name	The name of the object	Predeterminada	Texto	***
22	Created		Predeterminada	Fecha y hora	
23	Updated		Predeterminada	Fecha y hora	

Predeterminada Texto Añadir

Figura 29: Objeto Ciudades con sus campos

Departamentos

Buscar en todos los esquemas Crear objeto ***

ID	Nombre	Descripción	Tipo	Valor de tipo	Valor adicional
16	Key		Predeterminada	Texto	
17	Name	The name of the object	Predeterminada	Texto	***
18	Created		Predeterminada	Fecha y hora	
19	Updated		Predeterminada	Fecha y hora	
28	Ciudad		Objeto	Ciudades	Reference ***

Predeterminada Texto Añadir

Figura 30: Objeto Departamentos con sus campos

Personas

Crear objeto
...

Objetos
Atributos

ID	Nombre	Descripción	Tipo	Valor de tipo	Valor adicional
24	Key		Predeterminada	Texto	
25	Nombre_Usuario	The name of the object	Predeterminada	Texto	...
26	Created		Predeterminada	Fecha y hora	
27	Updated		Predeterminada	Fecha y hora	
29	Apellido_1		Predeterminada	Texto	...
30	Apellido_2		Predeterminada	Texto	...
31	Dirección		Objeto	Ciudades	Reference
32	Puesto_de_Trabajo		Predeterminada	Texto	...
33	Departamento		Objeto	Departamentos	Reference
34	Empleado	Determina si el empleado se ha ido o no	Predeterminada	Booleano	...
39	Nomina		Predeterminada	Texto	...

Predeterminada
▼
Texto
▼
Añadir

Figura 31: Objeto Empleados con sus campos

f. Permisos

Hay muchos permisos, así que solo se mencionarán algunos de ellos. Los directivos tendrán absolutamente todos los permisos, esto quiere decir que serán los administradores del proyecto y podrán configurarlo y trabajar en las incidencias.

Los técnicos estarán más limitados, ya que solo van a poder trabajar en las incidencias. Los permisos que destacar son, por ejemplo, crear, asignar, editar, borrar, resolver incidencias y algunos más.

Project Permissions	
Permission	Users / Groups / Project Roles
Administer Projects Ability to administer a project in Jira.	Project Role (atlassian-addons-project-access) Project Role (Directivo RRHH)
Browse Projects Ability to browse projects and the issues within them.	Project Role (atlassian-addons-project-access) Project Role (Técnico RRHH) Project Role (Directivo RRHH)
Manage sprints Ability to manage sprints.	Project Role (atlassian-addons-project-access)
Service Project Agent Allows users to interact with customers and access Jira Service Management features of a project.	Project Role (Directivo RRHH) Project Role (Técnico RRHH)
View aggregated data NEW Users with this permission will have access to view combined and summarised project data, regardless of their individual permissions. Learn more.	Project Role (atlassian-addons-project-access)
View Development Tools Allows users in a software project to view development-related information on the issue, such as commits, reviews and build information.	Project Role (atlassian-addons-project-access)
View Read-Only Workflow Users with this permission may view a read-only version of a workflow.	Project Role (atlassian-addons-project-access) Project Role (Directivo RRHH) Project Role (Técnico RRHH)

Figura 32: Permisos de proyecto

Issue Permissions	
Permission	Users / Groups / Project Roles
Assignable User Users with this permission may be assigned to issues.	Project Role (atlassian-addons-project-access) Project Role (Técnico RRHH)
Assign Issues Ability to assign issues to other people.	Project Role (atlassian-addons-project-access) Project Role (Directivo RRHH) Project Role (Técnico RRHH)
Close Issues Ability to close issues. Often useful where your developers resolve issues, and a QA department closes them.	Project Role (atlassian-addons-project-access) Project Role (Técnico RRHH) Project Role (Directivo RRHH)
Create Issues Ability to create issues.	Project Role (atlassian-addons-project-access) Project Role (Técnico RRHH) Project Role (Directivo RRHH) Service Project Customer - Portal Access
Delete Issues Ability to delete issues.	Project Role (atlassian-addons-project-access) Project Role (Directivo RRHH) Project Role (Técnico RRHH)
Edit Issues Ability to edit issues.	Project Role (atlassian-addons-project-access) Project Role (Directivo RRHH) Project Role (Técnico RRHH)
Link Issues Ability to link issues together and create linked issues. Only useful if issue linking is turned on.	Project Role (atlassian-addons-project-access) Project Role (Directivo RRHH) Project Role (Técnico RRHH)
Modify Reporter Ability to modify the reporter when creating or editing an issue.	Project Role (atlassian-addons-project-access) Project Role (Técnico RRHH) Project Role (Directivo RRHH)

Figura 33: Permisos de incidencias

Move Issues Ability to move issues between projects or between workflows of the same project (if applicable). Note the user can only move issues to a project they have the create permission for.	Project Role (atlassian-addons-project-access) Project Role (Técnico RRHH) Project Role (Directivo RRHH)
Resolve Issues Ability to resolve and reopen issues. This includes the ability to set a fix version.	Project Role (atlassian-addons-project-access) Project Role (Técnico RRHH) Project Role (Directivo RRHH)
Schedule Issues Ability to view or edit an issue's due date.	Project Role (atlassian-addons-project-access) Project Role (Técnico RRHH) Project Role (Directivo RRHH)
Set Issue Security Ability to set the level of security on an issue so that only people in that security level can see the issue.	Project Role (atlassian-addons-project-access) Project Role (Técnico RRHH) Project Role (Directivo RRHH)
Transition Issues Ability to transition issues.	Project Role (atlassian-addons-project-access) Project Role (Técnico RRHH) Project Role (Directivo RRHH)

Figura 34: Permisos de incidencias 2

Voters & Watchers Permissions	
Permission	Users / Groups / Project Roles
Manage Watchers Ability to manage the watchers of an issue.	Project Role (atlassian-addons-project-access) Project Role (Directivo RRHH) Project Role (Técnico RRHH)
View Voters and Watchers Ability to view the voters and watchers of an issue.	Project Role (atlassian-addons-project-access) Project Role (Directivo RRHH) Project Role (Técnico RRHH)

Figura 35: Permisos de votaciones

Comments Permissions	
Permission	Users / Groups / Project Roles
Add Comments Ability to comment on issues.	Project Role (atlassian-addons-project-access) Project Role (Directivo RRHH) Service Project Customer - Portal Access Project Role (Técnico RRHH)
Delete All Comments Ability to delete all comments made on issues.	Project Role (atlassian-addons-project-access) Project Role (Técnico RRHH) Project Role (Directivo RRHH)
Delete Own Comments Ability to delete own comments made on issues.	Project Role (atlassian-addons-project-access) Project Role (Técnico RRHH) Project Role (Directivo RRHH)
Edit All Comments Ability to edit all comments made on issues.	Project Role (atlassian-addons-project-access) Project Role (Directivo RRHH) Project Role (Técnico RRHH)
Edit Own Comments Ability to edit own comments made on issues.	Project Role (atlassian-addons-project-access) Project Role (Directivo RRHH) Project Role (Técnico RRHH) Service Project Customer - Portal Access
Attachments Permissions	
Permission	Users / Groups / Project Roles
Create Attachments Users with this permission may create attachments.	Project Role (atlassian-addons-project-access) Project Role (Directivo RRHH) Service Project Customer - Portal Access Project Role (Técnico RRHH)
Delete All Attachments Users with this permission may delete all attachments.	Project Role (atlassian-addons-project-access) Project Role (Técnico RRHH) Project Role (Directivo RRHH)
Delete Own Attachments Users with this permission may delete own attachments.	Project Role (atlassian-addons-project-access) Project Role (Técnico RRHH) Project Role (Directivo RRHH) Service Project Customer - Portal Access

Figura 36: Permisos de comentarios y adjuntos

Time Tracking Permissions	
Permission	Users / Groups / Project Roles
Delete All Worklogs Ability to delete all worklogs made on issues.	Project Role (atlassian-addons-project-access) Project Role (Directivo RRHH)
Delete Own Worklogs Ability to delete own worklogs made on issues.	Project Role (atlassian-addons-project-access) Project Role (Técnico RRHH) Project Role (Directivo RRHH)
Edit All Worklogs Ability to edit all worklogs made on issues.	Project Role (atlassian-addons-project-access) Project Role (Directivo RRHH)
Edit Own Worklogs Ability to edit own worklogs made on issues.	Project Role (atlassian-addons-project-access) Project Role (Técnico RRHH) Project Role (Directivo RRHH)
Work On Issues Ability to log work done against an issue. Only useful if Time Tracking is turned on.	Project Role (atlassian-addons-project-access) Project Role (Directivo RRHH) Project Role (Técnico RRHH)

Figura 37: Permisos de informes de trabajo

g. Automation

Estos son algunos de los automatismos que se han usado en este proyecto y su función.

Los tres primeros sirven para añadir las ciudades, los departamentos y los usuarios. Cuando se cree una incidencia, se activará el automatismo y dependiendo del tipo de incidencia que sea, **“Añadir Ciudad”**, **“Añadir Departamento”** o **“Alta Laboral”** meterá la ciudad, el departamento o el usuario en los Assets.

Automatización HABILITADO [Volver a la l](#)

Añadir Ciudad

① Detalles de la regla
📄 Registro de auditoría

+ When: Incidencia creada
La regla se ejecuta cuando se crea una incidencia.

🔍 If: La incidencia coincide con el JQL issueType = "Añadir Ciudad"

+ Then: Crear Ciudades
Desde Jeremías

○ Añadir componente

Nombre *
Añadir Ciudad

Descripción

Alcance
🌐 Global
Solo se puede modificar el alcance en la administración global.

Permitir desencadenador de regla
☐ Marca esto para permitir que las acciones de otra regla desencadenen esta regla. Habilita la opción únicamente si necesitas que esta regla se ejecute en respuesta a otra regla.

Notificar al producirse un error
No notificar

Propietario *
Eric Quintero Moreno

El propietario recibirá mensajes de correo electrónico cuando la regla genere un error.

Creado:
a day ago

Agente *
Automation for Jira

Las acciones definidas en esta regla las realizará el usuario seleccionado como el actor. [Más información](#) sobre los actores de las reglas de automatización.

¿Quién puede editar esta regla? *
🔒 Todos los administradores

Guardar

Figura 38: Automatismo alta laboral

Automatización

HABILITADO

Añadir Departamento

○ Detalles de la regla

📄 Registro de auditoría

+ When: Incidencia creada
La regla se ejecuta cuando se crea una incidencia.

🔍 If: La incidencia coincide con el JQL
issueType = "Añadir Departamento"

+ Then: Crear Departamentos
Desde Jeremías

○ Añadir componente

Detalles de la regla

Nombre *

Añadir Departamento

Descripción

Alcance

🌐 Global

Solo se puede modificar el alcance en la administración global.

Permitir desencadenador de regla

☐ Marca esto para permitir que las acciones de otra regla desencadenen esta regla. Habilita la opción únicamente si necesitas que esta regla se ejecute en respuesta a otra regla.

Notificar al producirse un error

No notificarme

Propietario *

👤 Eric Quintero Moreno

El propietario recibirá mensajes de correo electrónico cuando la regla genere un error.

Creado

a day ago

Agente *

🤖 Automation for Jira

Las acciones definidas en esta regla las realizará el usuario seleccionado como el actor. [Más información sobre los actores de las reglas de automatización.](#)

¿Quién puede editar esta regla? *

🔒 Todos los administradores

Guardar

Figura 39: Automatismo añadir departamento

Automatización

HABILITADO

Añadir Usuario

○ Detalles de la regla

📄 Registro de auditoría

🔄 When: Transición de incidencia realizada
HASTA
VALIDADO

🔍 If: La incidencia coincide con el JQL
issueType = "Alta Laboral"

+ Then: Crear Personas
Desde Jeremías

○ Añadir componente

Detalles de la regla

Nombre *

Añadir Usuario

Descripción

Alcance

🌐 Global

Solo se puede modificar el alcance en la administración global.

Permitir desencadenador de regla

☐ Marca esto para permitir que las acciones de otra regla desencadenen esta regla. Habilita la opción únicamente si necesitas que esta regla se ejecute en respuesta a otra regla.

Notificar al producirse un error

Enviar un mensaje de correo electrónico al propietario de la regla una v...

Propietario *

👤 Eric Quintero Moreno

El propietario recibirá mensajes de correo electrónico cuando la regla genere un error.

Creado

2 days ago

Actualizado

a day ago

Agente *

🤖 Automation for Jira

Las acciones definidas en esta regla las realizará el usuario seleccionado como el actor. [Más información sobre los actores de las reglas de automatización.](#)

¿Quién puede editar esta regla? *

🔒 Todos los administradores

Figura 40: Automatismo añadir usuario

Los dos siguientes automatismos sirven para que cuando el usuario seleccione un usuario de los Assets, los campos se rellenen automáticamente a la hora de crear o editar la incidencia. El automatismo se iniciará cuando se cree una incidencia o se edite una

incidencia, si el tipo de incidencia es “**Obtención de Nomina**” o “**Baja Laboral**”, rellenará los campos automáticamente.

Proyectos / Jeremías RRHH / Configuración del proyecto

Automatización HABILITADO [Volver a](#)

Añadir datos de usuario al seleccionarlo en la creación de la incidencia

○ Detalles de la regla
■ Registro de auditoría

When: Incidencia creada
La regla se ejecuta cuando se crea una incidencia.

If: La incidencia coincide con el JQL
issuetype IN ("Obtención de Nómina", "Baja Laboral")

Then: Editar campos de incidencia
Nombre **Apellido 1** **Apellido 2** **Puesto** **Nómina**

○ Añadir componente

Detalles de la regla

Nombre*
Añadir datos de usuario al seleccionarlo en la creación de la incidencia

Descripción

Alcance
Jeremías RRHH
Solo se puede modificar el alcance en la administración global.

Permitir desencadenador de regla
☐ Marca esto para permitir que las acciones de otra regla desencadenen esta regla. Habilita la opción únicamente si necesitas que esta regla se ejecute en respuesta a otra regla.

Notificar al producirse un error
Enviar un mensaje de correo electrónico al propietario de la regla una ...

Propietario*
Eric Quintero Moreno
El propietario recibirá mensajes de correo electrónico cuando la regla genere un error.

Creado
2 days ago

Agentes*
Automation for Jira
Las acciones definidas en esta regla las realizará el usuario seleccionado como el actor. Más información sobre los actores de las reglas de automatización.
¿Quién puede editar esta regla?*
Todos los administradores

Figura 41: Automatismo campos al crear

Proyectos / Jeremías RRHH / Configuración del proyecto

Automatización HABILITADO [Volver a la lista](#)

Añadir datos de usuario al seleccionarlo en la edición de la incidencia

○ Detalles de la regla
■ Registro de auditoría

When: Se cambia el valor para Usuario

If: La incidencia coincide con el JQL
issuetype IN ("Obtención de Nómina", "Baja Laboral")

Then: Editar campos de incidencia
Apellido 1 **Apellido 2** **Nómina** **Puesto**

○ Añadir componente

Detalles de la regla

Nombre*
Añadir datos de usuario al seleccionarlo en la edición de la incidencia

Descripción

Alcance
Jeremías RRHH
Solo se puede modificar el alcance en la administración global.

Permitir desencadenador de regla
☐ Marca esto para permitir que las acciones de otra regla desencadenen esta regla. Habilita la opción únicamente si necesitas que esta regla se ejecute en respuesta a otra regla.

Notificar al producirse un error
Enviar un mensaje de correo electrónico al propietario de la regla una v...

Propietario*
Eric Quintero Moreno
El propietario recibirá mensajes de correo electrónico cuando la regla genere un error.

Creado
2 days ago

Actualizado
2 days ago

Agentes*
Automation for Jira
Las acciones definidas en esta regla las realizará el usuario seleccionado como el actor. Más información sobre los actores de las reglas de automatización.
¿Quién puede editar esta regla?*
Todos los administradores

Figura 42: Automatismo campos al editar

El último automatismo es para dar de baja a un usuario, es muy parecido a los tres primeros, el automatismo se ejecutará cuando se cree una incidencia y si es del tipo “**Baja Laboral**”, editará el usuario cambiando el campo del usuario. Los usuarios tienen un campo de tipo Booleano (Verdadero o Falso) que sirven para saber si el usuario está activo en la empresa o no y no tener que eliminarlo y perder sus datos.

Proyectos / Jeremías RRHH / Configuración del proyecto

Automatización

HABILITADO

Baja de usuario

ⓘ Detalles de la regla

📄 Registro de auditoría

🔍 When: Transición de incidencia realizada

HASTA

VALIDADO

🔍 If: La incidencia coincide con el JQL

issueType ="Baja Laboral"

👤 Para AQL

Nombre_Usuario =

{(issue."Usuario").Nombre_Usuario}}

📄 Then: Editar atributos del objeto

Empleado

⊖ Añadir componente

⊖ Añadir componente

Detalles de la regla

Nombre *

Baja de usuario

Descripción

Alcance

Global

Solo se puede modificar el alcance en la [administración global](#).

Permitir desencadenador de regla

☐ Marca esto para permitir que las acciones de otra regla desencadenen esta regla. Habilita la opción únicamente si necesitas que esta regla se ejecute en respuesta a otra regla.

Notificar al producirse un error

Enviar un mensaje de correo electrónico al propietario de la regla una v...

Propietario *

Eric Quintero Moreno

El propietario recibirá mensajes de correo electrónico cuando la regla genere un error.

Creado

2 days ago

Agente *

Automation for Jira

Las acciones definidas en esta regla las realizará el usuario seleccionado como el actor. [Más información sobre los actores de las reglas de automatización.](#)

¿Quién puede editar esta regla? *

Todos los administradores

Guardar

Figura 43: Automatismo baja de usuario

Cabe recalcar que, para acceder a los campos de las incidencias a la hora de meterlos en los Assets, se accede poniendo “**issue.NOMBRE_DEL_CAMPO**”.

h. Scripts

En el proyecto también se usa programación, para ello se usa una aplicación llamada ScriptRunner. Esta aplicación es muy útil ya que nos permite crear scripts y asociarlos a los workflows, programarlos o incluso tener unos listeners que harán que se active el script cuando pase algo determinado.

En este proyecto, todos los scripts están dentro de los workflows, esto quiere decir que cuando una incidencia pase de un determinado estado a otro, el script se activará.

El primer script es para las incidencias de tipo **“Alta Laboral”**, **“Baja Laboral”**, **“Añadir Ciudad”** y **“Añadir Departamento”** y se activará cuando el directivo de RRHH valide la incidencia y pase al estado **“Validada”**. La función de dicho script es para que cuando se cree una incidencia de los tipos mencionados anteriormente y se añadan los datos, cree una incidencia en la instancia de IT avisando de que se ha hecho un cambio en la base de datos. Para ello se necesita el correo del usuario, un token API y la URL de la instancia y acabada con lo que pone en la imagen. La incidencia se creará en la otra instancia mediante el uso de las APIs de Jira y JSON. En el JSON hay que poner los campos que vamos a rellenar de la incidencia, en este caso, se copiarán de la incidencia ya existente.

```
1 def issueKey = issue.key
2
3 def result = get('/rest/api/2/issue/' + issueKey)
4     .header('Content-Type', 'application/json')
5     .asObject(Map)
6
7 def url = "https://grupofjeremias2.atlassian.net/rest/api/2/issue/"
8 def token = "AtIAT13xfGf0-F9j0zH2irctoM0bLfdivsGV5G1co_q0MpgVRE_R_vl-Adgitz"
9 def email = "equintero@inlogiq.com"
10
11 try {
12     def response = post(url)
13         .basicAuth(email, token)
14         .queryString("overrideScreenSecurity", Boolean.TRUE)
15         .header('Content-Type', 'application/json')
16         .body([
17             fields: [
18                 project: [
19                     key: "J1"
20                 ],
21                 summary: result.body.fields.summary,
22                 description: result.body.fields.description,
23                 issuetype: [
24                     id: "10015"
25                 ],
26                 customfield_10066: result.body.fields.customfield_10039,
27                 customfield_10067: result.body.fields.customfield_10040,
28                 customfield_10070: result.body.fields.customfield_10041,
29                 customfield_10068: result.body.fields.customfield_10042,
30                 customfield_10069: result.body.fields.customfield_10043
31             ]
32         ]) .asString()
33     logger.info("Response: " + response)
34 } catch (Exception e){
35     logger.info("Error creating issue: " + e.getMessage())
36 }
```

Figura 44: Script 1

El segundo script se activará de la misma manera y con los mismos tipos de incidencia que el anterior, pero esta vez su función consiste en cerrar automáticamente esa incidencia y pasarla al estado **“Cerrado”**. Para ello será también con el api, pero esta vez con la URL, pero acabada en **“/transitions”**, de esta manera la podremos transicionar poniendo en el JSON y el id del estado.

```
1 def issueKey = issue.key
2
3 def result = get('/rest/api/2/issue/' + issueKey)
4   .header('Content-Type', 'application/json')
5   .asObject(Map)
6
7 if (result.body.fields.status.name == "VALIDADO"){
8   post('/rest/api/2/issue/' + issueKey + "/transitions")
9     .header('Content-Type', 'application/json')
10    .body([
11      transition: [id: 21]
12    ])
13    .asString().body
14 }
```

Figura 45: Script 2

El tercer script se activará de la misma manera y con los mismos tipos de incidencia que los dos anteriores. La función de este script es para que cuando se suban los datos a los Assets, deje un comentario indicando que se ha añadido con éxito la ciudad, el departamento, el usuario o la baja dependiendo de que tipo de incidencia sea.

```
def issueKey = issue.key
def result = get('/rest/api/2/issue/' + issueKey)
  .header('Content-Type', 'application/json')
  .asObject(Map)

def comment = "Se ha dado de alta al usuario"
def comment2 = "Se ha dado de baja al usuario"
def comment3 = "Se ha añadido la ciudad"
def comment4 = "Se ha añadido el departamento"

if(result.body.fields.issueType.name == "Alta Laboral"){
  put('/rest/api/2/issue/' + issueKey)
    .header('Content-Type', 'application/json')
    .body([
      update:[
        comment:[
          add:[
            body: comment
          ]
        ]
      ]
    ])
  .asString()
} else if(result.body.fields.issueType.name == "Baja Laboral"){
  put('/rest/api/2/issue/' + issueKey)
    .header('Content-Type', 'application/json')
    .body([
      update:[
        comment:[
          add:[
            body: comment2
          ]
        ]
      ]
    ])
  .asString()
} else if(result.body.fields.issueType.name == "Alta Ciudad"){
  put('/rest/api/2/issue/' + issueKey)
    .header('Content-Type', 'application/json')
    .body([
      update:[
        comment:[
          add:[
            body: comment3
          ]
        ]
      ]
    ])
  .asString()
} else if(result.body.fields.issueType.name == "Alta Departamento"){
  put('/rest/api/2/issue/' + issueKey)
    .header('Content-Type', 'application/json')
    .body([
      update:[
        comment:[
          add:[
            body: comment4
          ]
        ]
      ]
    ])
  .asString()
}
```

Figura 46: Script 3

El último script se activará cuando se cree una incidencia de tipo “**Tarea**” o “**Error**” ya que los dos comparten el mismo workflow. Su función es para si la incidencia es de tipo “**Error**” deje un comentario diciendo que puede pedir ayuda al departamento de IT.

```
1 def issueKey = issue.key
2
3 def result = get('/rest/api/2/issue/' + issueKey)
4   .header('Content-Type', 'application/json')
5   .asObject(Map)
6
7 def comment = "Recuerda que si es algún error relacionado con el hardware o software, puedes ponerte en contacto con IT mediante el portal: https://grupofjermlas2.atlassian.net/service desk/customer/portal"
8
9 if(result.body.fields.issueType.name == "Bug"){
10   put('/rest/api/2/issue/' + issueKey)
11     .header('Content-Type', 'application/json')
12     .body([
13       update:[
14         comment:[
15           add:[
16             body: comment
17           ]
18         ]
19       ]
20     ])
21   .asString()
22 }
23 }
```

Figura 47: Script 4

i. Enlace al proyecto

[JRRHH board - Tablero ágil - Jira \(atlassian.net\)](#)

11.2. ANEXO 2: PROYECTO JIRA SERVICE MANAGMENT

a. Tipos de incidencias

Tipo de incidencia	Solicitudes
Solicitud	Material nueva incorporación
Incidencia	Fallo hardware Fallo software
Consulta	Consulta genérica
Alta/Baja de usuario	No tiene solicitudes y está oculto del portal, sirve para que cuando se haga la copia de las incidencias mencionadas anteriormente en RRHH, se asignen a este tipo de incidencia.

Tipos de incidencias

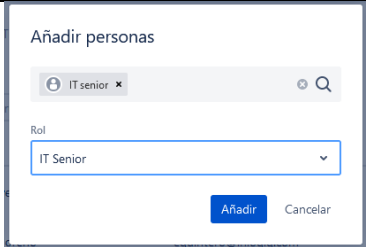
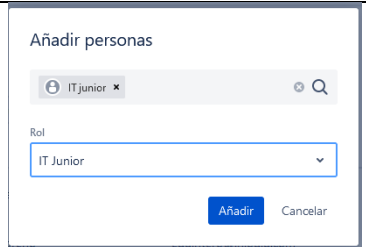
Nota: Aquí se ven los tipos de incidencias

b. Roles y grupos

Los roles y grupos de este proyecto se distribuyen de la misma manera que el anterior.

Los IT Senior son los administradores del proyecto, ya que llevan más tiempo y tienen más experiencia en Jira y los IT Junior solo podrán trabajar con las incidencias ya que al ser nuevos pueden cometer algún error crítico.

En cuanto a los grupos, también hay dos, uno para los IT Junior y otro para los IT Senior.

Rol	Grupo Correspondiente	Asignación
IT Senior	IT Senior	 <i>Figura 48: Asignando el rol al grupo</i>
IT Junior	IT Junior	 <i>Figura 49: Asignando el rol al grupo</i>

Grupos y Roles

Nota: Aquí se ven los roles y grupos que existen

c. Workflows

De la misma manera que en el proyecto anterior, este proyecto tiene cuatro workflows, uno para cada tipo de incidencia.

Proyectos / Jeremías IT / Configuración del proyecto

Flujos de trabajo

Jeremías IT

Agregar Flujo de Trabajo ▼ Cambiar el Esquema

Flujo de trabajo	Tipos de incidencia	Acciones
Alta/Baja Usuario (Ver como texto / diagrama)	Alta/Baja Usuario (Asignar)	
Jeremías IT - Consulta (Ver como texto / diagrama)	Consulta (Asignar)	
Jeremías IT - Incidencia (Ver como texto / diagrama)	Incidencia (Asignar)	
Jeremías IT - Solicitud (Ver como texto / diagrama)	Solicitud (Asignar)	

Figura 50: Tipos de incidencias con sus workflows

Para el tipo de incidencia de **“Solicitud”**, cuando se cree la incidencia pasará al estado **“Abierta”**, cuando se esté trabajando en ella estará **“en progreso”**, si se pausa se quedará en **“pendiente”**, si hace falta más información se pondrá en **“esperando al cliente”** y una vez resuelta se transicionará al estado **“Cerrado”**. Al abrirse la incidencia podría darse el caso de que se cancelará, en ese caso la incidencia pasará a **“Cancelada”**.



Figura 51: Workflow de Solicitud

El otro workflow es el de **“Incidencia”**, de la misma forma que el anterior cuando se crea la incidencia se queda en **“Abierta”**, si se está

trabajando en ella pasa a **"en curso"**, en caso de que se quiera pasar esa incidencia al equipo de desarrollo para que trabajen en ella se pasará al estado **"escalado"**, una vez se terminé pasará a **"resuelto"** y por último cuando el IT Senior la revisé entrará en **"Cerrada"**.



Figura 52: Workflow de Incidencia

El tercer workflow es para el tipo de incidencia **"Alta/Baja Usuario"**, este es el más sencillo de todos, ya que, al ser un aviso, la incidencia se abrirá y solo se podrá cerrar porque no se va a trabajar en ella.



Figura 53: Workflow de alta/baja usuario

El último workflow es para **"Consulta"**, es el mismo que el primero solamente que la incidencia no se puede cancelar.



Figura 54: Workflow de Consulta

d. Permisos

Los permisos se distribuyen de dos maneras, IT Senior será el administrador de todo el proyecto e irá guiando al equipo, mientras que IT Junior se encargará de ir completando las incidencias que vayan surgiendo.

Por lo tanto, IT Senior tendrá todos los permisos del proyecto mientras que IT Junior solo los necesarios para poder trabajar con las incidencias.

Project Permissions	
Permission	Users / Groups / Project Roles
Administer Projects Ability to administer a project in Jira.	Project Role (atlassian-addons-project-access) Project Role (IT Senior)
Browse Projects Ability to browse projects and the issues within them.	Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)
Service Project Agent Allows users to interact with customers and access Jira Service Management features of a project.	Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)
View Development Tools Allows users in a software project to view development-related information on the issue, such as commits, reviews and build information.	Project Role (atlassian-addons-project-access) Project Role (IT Senior)
View Read-Only Workflow Users with this permission may view a read-only version of a workflow.	Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)

Figura 55: Permisos de proyecto

Issue Permissions	
Permission	Users / Groups / Project Roles
Assignable User Users with this permission may be assigned to issues.	Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)
Assign Issues Ability to assign issues to other people.	Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)
Close Issues Ability to close issues. Often useful where your developers resolve issues, and a QA department closes them.	Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)
Create Issues Ability to create issues.	Project Role (atlassian-addons-project-access) Service Project Customer - Portal Access Project Role (IT Junior) Project Role (IT Senior)
Delete Issues Ability to delete issues.	Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)
Edit Issues Ability to edit issues.	Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)
Link Issues Ability to link issues together and create linked issues. Only useful if issue linking is turned on.	Project Role (atlassian-addons-project-access) Project Role (IT Senior) Project Role (IT Junior)
Modify Reporter Ability to modify the reporter when creating or editing an issue.	Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)

Figura 56: Permisos de incidencias

Move Issues Ability to move issues between projects or between workflows of the same project (if applicable). Note the user can only move issues to a project they have the create permission for.	Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)
Resolve Issues Ability to resolve and reopen issues. This includes the ability to set a fix version.	Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)
Schedule Issues Ability to view or edit an issue's due date.	Project Role (atlassian-addons-project-access) Project Role (IT Senior) Project Role (IT Junior)
Set Issue Security Ability to set the level of security on an issue so that only people in that security level can see the issue.	Project Role (atlassian-addons-project-access) Project Role (IT Senior) Project Role (IT Junior)
Transition Issues Ability to transition issues.	Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)
Voters & Watchers Permissions	
Permission	Users / Groups / Project Roles
Manage Watchers Ability to manage the watchers of an issue.	Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)
View Voters and Watchers Ability to view the voters and watchers of an issue.	Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)

Figura 57: Permisos de incidencias 2 y votaciones

Comments Permissions	Users / Groups / Project Roles
Permission	
Add Comments Ability to comment on issues.	Service Project Customer - Portal Access Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)
Delete All Comments Ability to delete all comments made on issues.	Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)
Delete Own Comments Ability to delete own comments made on issues.	Service Project Customer - Portal Access Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)
Edit All Comments Ability to edit all comments made on issues.	Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)
Edit Own Comments Ability to edit own comments made on issues.	Service Project Customer - Portal Access Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)

Figura 58: Permisos de comentarios

Attachments Permissions	Users / Groups / Project Roles
Permission	
Create Attachments Users with this permission may create attachments.	Project Role (Administrators) Service Project Customer - Portal Access Project Role (Service Desk Team) Project Role (atlassian-addons-project-access)
Delete All Attachments Users with this permission may delete all attachments.	Project Role (Administrators) Project Role (atlassian-addons-project-access) Project Role (Service Desk Team)
Delete Own Attachments Users with this permission may delete own attachments.	Project Role (Administrators) Service Project Customer - Portal Access Project Role (Service Desk Team) Project Role (atlassian-addons-project-access)

Figura 59: Permisos de adjuntos

Time Tracking Permissions	
Permission	Users / Groups / Project Roles
Delete All Worklogs Ability to delete all worklogs made on issues.	Project Role (atlassian-addons-project-access) Project Role (IT Senior)
Delete Own Worklogs Ability to delete own worklogs made on issues.	Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)
Edit All Worklogs Ability to edit all worklogs made on issues.	Project Role (atlassian-addons-project-access) Project Role (IT Senior)
Edit Own Worklogs Ability to edit own worklogs made on issues.	Project Role (atlassian-addons-project-access) Project Role (IT Junior) Project Role (IT Senior)
Work On Issues Ability to log work done against an issue. Only useful if Time Tracking is turned on.	Project Role (atlassian-addons-project-access)

Figura 60: Permisos de informes de trabajo

e. **Notificaciones**

En este proyecto se han configurado las notificaciones para no sobrecargar a los de IT. Estos solo recibirán notificaciones en estos casos:

- Cuando se cree una incidencia solamente recibirán las notificaciones el grupo de IT Junior, ya que solamente ellos van a trabajar en las incidencias.
- Cuando un IT Senior le asigne a alguien una incidencia o tarea, se le notificará al usuario al que se le ha asignado.
- Por último, cuando alguien comente en una incidencia, se notificará al usuario que tenga asignada esa incidencia.

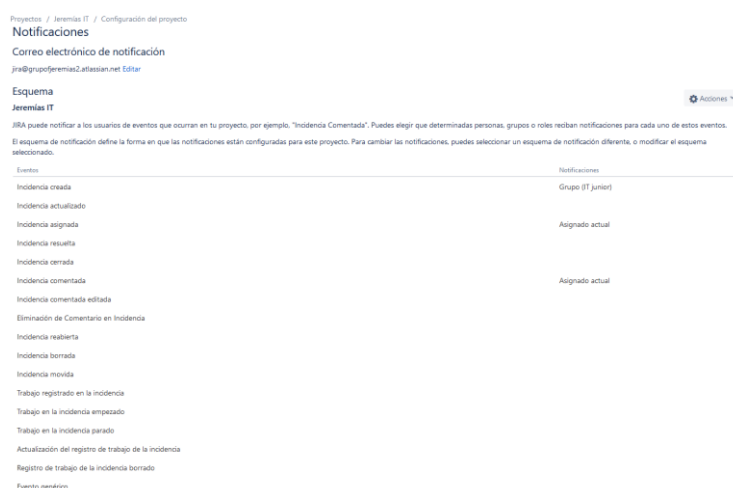



Figura 61: Configuración de notificaciones

f. SLAs

En cuanto a los SLAs tenemos dos tipos, uno para el primer tiempo de respuesta y otro para el tiempo de resolución. Estos SLA evidentemente, se cumplen dentro del horario laboral.




En primer lugar, tenemos el primer tiempo de respuesta, este SLA sirve para definir cuánto tiempo tardará el IT Junior en responder para empezar a trabajar en la incidencia. Para los tipos de incidencia “**Solicitud**” e “**Incidencia**” tenemos un tiempo máximo de 1 hora, mientras que para el tipo de “**Consulta**” horas. Este SLA empieza cuando se crea la incidencia y termina cuando comentamos al usuario que se ha resuelto la incidencia.

 Time to first response

Edit ...

Goals

Issues will be checked against this list, top to bottom, and assigned a time goal based on the first matching JQL statement.

Time Goal	Calendar	Issues to display (in JQL)
1h	 Sample 9-5 Calendar	issueType in (Solicitud, Incidencia)
2h	 Sample 9-5 Calendar	issueType = Consulta
No target	 Sample 9-5 Calendar	All remaining issues

Conditions

Time will be measured between the start and stop conditions below.

Start counting time when...

Issue Created

Pause counting time during...

No conditions

Finish counting time when...

Comment: For Customers

SLA format display





Due date centric e.g. Today 11:30  Today 07:00  24 Feb 17:00 

Figura 62: SLA Time to First Response

Para el segundo SLA, el de tiempo de resolución, para los tipos de incidencia “**Solicitud**” y “**Consulta**” tenemos un tiempo máximo de 8 horas y para el tipo “**Incidencia**” 3 horas. La función de esta SLA es para saber cuánto tiempo tardaran los de IT en solucionar esa incidencia.

El SLA iniciará cuando la incidencia se cree y puede pausarse cuando falta información para poder solucionar esa incidencia, ya sea porque, por ejemplo, tenemos que esperar a que el cliente nos responda a una pregunta.




Por último, el SLA se cerrará cuando la incidencia sea cancelada, cerrada o resuelta.

 Time to resolution

Edit ...

Goals

Issues will be checked against this list, top to bottom, and assigned a time goal based on the first matching JQL statement.

Time Goal	Calendar	Issues to display (in JQL)
8h	 Sample 9-5 Calendar	issueType in (Solicitud, Consulta)
3h	 Sample 9-5 Calendar	issueType = Incidencia
No target	 Sample 9-5 Calendar	All remaining issues

Conditions

Time will be measured between the start and stop conditions below.

Start counting time when...

Issue Created

Pause counting time during...

Status: Esperando a cliente

Status: Pendiente

Finish counting time when...

Entered Status: Cancelado

Entered Status: Cerrado

Entered Status: Resuelto

SLA format display




Due date centric e.g. Today 11:30  Today 07:00  24 Feb 17:00 

Figura 63: SLA Time to Resolution

g. Automation

En este proyecto solo hay 2 automatismos. Los dos automatismos, tienen la misma función, informar a los IT Junior cuando los SLA se han infringido, solo que uno es para el primer tiempo de respuesta y el otro para el tiempo de resolución. Cuando el tiempo se agote y se infringen los SLA, dejará un comentario en esa incidencia para los IT Junior diciendo que se ha infringido el SLA.

Proyectos / Jeremías IT / Configuración del proyecto

Automatización

HABILITADO

[Volver a la lista](#)



Copy of J1: Poner un comentario en la incidencia si se rompe el SLA (Time to First Response)

- ① Detalles de la regla
- 📄 Registro de auditoría

⌚ When: Umbral de SLA incumplido
Time to first response se haya incumplido

🔧 Then: Añadir comentario a la incidencia
Se ha incumplido el SLA de "Time to First Response"

○ Añadir componente

Detalles de la regla

Nombre *

Copy of J1: Poner un comentario en la incidencia si se rompe el SLA (Time to Fi

Descripción

Alcance

Jeremías IT

Solo se puede modificar el alcance en la [administración global](#).

Permitir desencadenador de regla

☐ Marca esto para permitir que las acciones de otra regla desencadenen esta regla. Habilita la opción únicamente si necesitas que esta regla se ejecute en respuesta a otra regla.

Notificar al producirse un error

Enviar un mensaje de correo electrónico al propietario de la regla una ...

Propietario *

Eric Quintero Moreno

El propietario recibirá mensajes de correo electrónico cuando la regla genere un error.

Creado

7 days ago

Figura 64: Automatismo de cuando se infringe el SLA Time to First Response

Proyectos / Jeremías IT / Configuración del proyecto

Automatización

HABILITADO

[Volver a la lista](#)



J1: Poner un comentario en la incidencia si se rompe el SLA (Time to resolution)

- ① Detalles de la regla
- 📄 Registro de auditoría

⌚ When: Umbral de SLA incumplido
Time to resolution se haya incumplido

🔧 Then: Añadir comentario a la incidencia
Se ha incumplido el SLA de "Time to resolution"

○ Añadir componente

Detalles de la regla

Nombre *

J1: Poner un comentario en la incidencia si se rompe el SLA (Time to resolution)

Descripción

Alcance

Jeremías IT

Solo se puede modificar el alcance en la [administración global](#).

Permitir desencadenador de regla

☐ Marca esto para permitir que las acciones de otra regla desencadenen esta regla. Habilita la opción únicamente si necesitas que esta regla se ejecute en respuesta a otra regla.

Notificar al producirse un error

Enviar un mensaje de correo electrónico al propietario de la regla una ...

Propietario *

Eric Quintero Moreno

El propietario recibirá mensajes de correo electrónico cuando la regla genere un error.

Creado

Figura 65: Automatismo de cuando se infringe el SLA Time to Resolution

h. Scripts

Al igual que en el anterior proyecto, en este también se usa la aplicación ScriptRunner para hacer los scripts. En este caso, hay cuatro scripts que están dentro de los workflows, como ya se ha mencionado en los anteriores, se ejecutarán cuando un tipo de incidencia concreto se transicione a un estado en concreto.

El primer script es para que cuando llegue una copia de alguna incidencia del proyecto RRHH, se deje un comentario en la copia indicando que es una copia del proyecto RRHH. Para ello se utiliza todo lo mencionado anteriormente en los otros scripts.

```
1 def issueKey = issue.key
2
3 def result = get('/rest/api/2/issue/' + issueKey)
4   .header('Content-Type', 'application/json')
5   .asObject(Map)
6
7 put('/rest/api/2/issue/' + issueKey)
8   .header('Content-Type', 'application/json')
9   .body([
10    update:[
11      comment:[
12        [
13          add:[
14            body: "Copia de la incidencia de RRHH"
15          ]
16        ]
17      ]
18    ]
19  ])
20  .asString()
```

Figura 66: Script 5

El segundo script se activará de la misma forma que el anterior y su función es para que cuando se cree la incidencia se cierre automáticamente ya que no se va a trabajar en ella, simplemente es un aviso. Como se hace ya se ha explicado anteriormente en otro script.

```

1  def issueKey = issue.key
2
3  def result = get('/rest/api/2/issue/' + issueKey)
4    .header('Content-Type', 'application/json')
5    .asObject(Map)
6
7  if (result.body.fields.issuetype.name == "Alta/Baja Usuario"){
8    post('/rest/api/2/issue/' + issueKey + "/transitions")
9      .header('Content-Type', 'application/json')
10     .body([
11       transition: [id: 11]
12     ])
13     .asString().body
14 }

```

Figura 67: Script 6

El tercer script es muy sencillo, simplemente es una condición para el workflow del tipo de incidencia **“Incidencia”**, sirve para que aparezca un estado o no dependiendo de si la solicitud es tipo hardware o software. Si es de tipo software se podrá cambiar el estado a **“Escalado”**.

```

1  issue.customfield_10010.requestType.name == "Fallo software"

```

Figura 68: Script 7

El cuatro script se activa cuando la incidencia del tipo **“Incidencia”** y con el tipo de solicitud **“Fallo software”** se escala. La función de este script es para que cuando se escale la incidencia, deje un comentario diciendo que se ha escalado con éxito.

```

1  def issueKey = issue.key
2
3  def result = get('/rest/api/2/issue/' + issueKey)
4    .header('Content-Type', 'application/json')
5    .asObject(Map)
6
7  def comment = 'Se ha escalado con éxito'
8
9  put('/rest/api/2/issue/' + issueKey)
10    .header('Content-Type', 'application/json')
11    .body([
12      update:[
13        comment:[
14          [
15            add:[
16              body: comment
17            ]
18          ]
19        ]
20      ]
21    ])
22    .asString()

```

Figura 69: Script 8

i. Portal y Outlook

Para los usuarios de RRHH, pueden crear incidencias a IT desde el propio portal de IT o desde el propio Outlook instalando el complemento de Jira.

Para crearlas desde el portal es sencillo, simplemente accedes al portal, inicias sesión y ya se podrán crear las incidencias a IT.

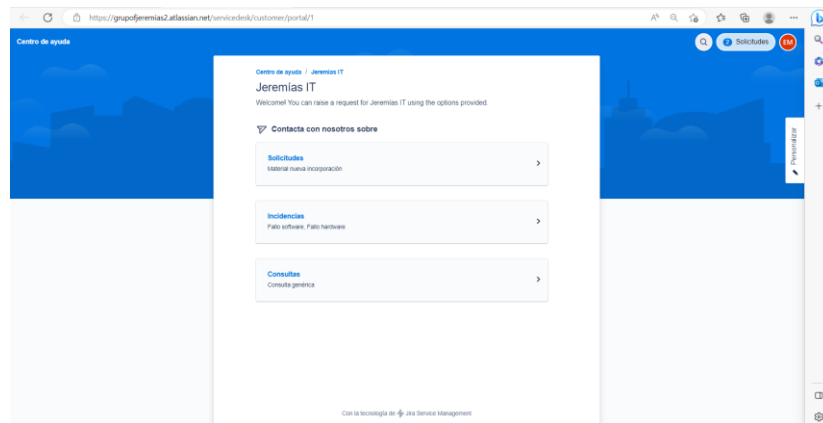


Figura 70: Portal

Para crearlas desde el Outlook también es de lo más sencillo, abres Outlook y en la parte superior en “**New Issue**”, aparecerá una pantalla donde se tendrá que iniciar sesión y a partir de aquí, se siguen los mismos pasos que en el portal.

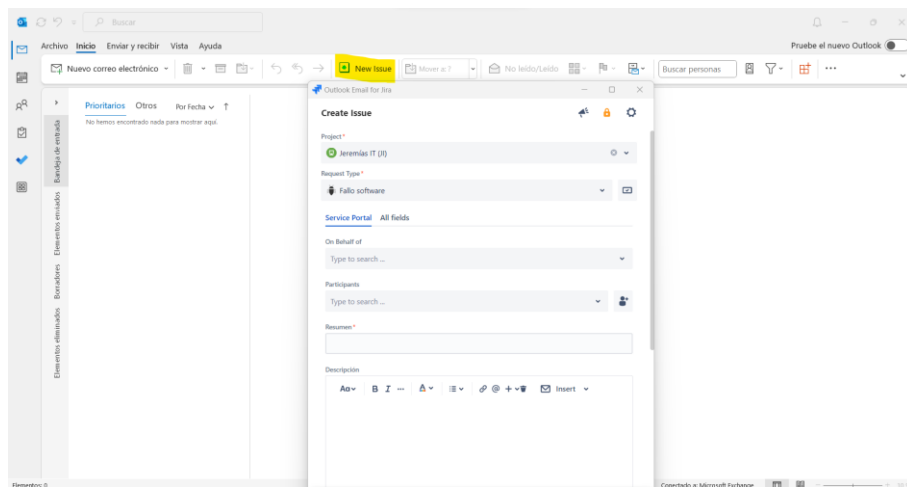


Figura 71: Crear incidencia desde outlook

j. Escalado

La opción del escalado es muy sencilla, se hace utilizando una aplicación llamada “**Exalate**”, ya mencionada anteriormente. Las incidencias del tipo “**Incidencia**” y con el tipo de solicitud “**Fallo software**” tendrán la opción de poner el estado “**Escalado**” y ya la incidencia se creará en el proyecto de desarrollo.

Estás incidencias están sincronizadas entre sí, esto quiere decir que compartirán todos los campos, entonces cuando un campo cambie o se haga un comentario en la incidencia aparecerá también en la otra incidencia. Esta es una manera de repartir el trabajo entre IT y desarrollo para que puedan trabajar entre ambos. Para que los campos se compartan se utiliza programación.

Para que la aplicación funcione correctamente hay que conectar las dos instancias con las URL de cada una.

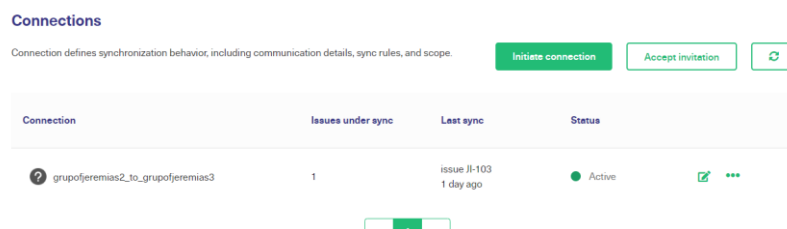


Figura 72: Conexiones de Exalate

▼ Outgoing sync ①

```

1 replica.key = issue.key
2 replica.type = issue.type
3 replica.assignee = issue.assignee
4 replica.reporter = issue.reporter
5 replica.summary = issue.summary
6 replica.description = issue.description
7 replica.labels = issue.labels
8 replica.comments = commentHelper.mergeComments(replica, issue)
9 replica.resolution = issue.resolution
10 replica.parentId = issue.parentId
11 replica.priority = issue.priority
12 replica.attachments = issue.attachments
13 replica.project = issue.project
14
15 //Comment these lines out if you are interested in sending the full list of versions and components of the source project.
16 replica.project.versions = []
17 replica.project.components = []
18
19 /*
20 Custom Fields
21
22 replica.customFields."CF Name" = issue.customFields."CF Name"
23 */

```

Copy outgoing sync processor to olipboard

▼ Incoming sync ①

```

1 if(firstSync){
2   issue.projectKey = "31"
3   // Set type name from source issue, if not found set a default
4   issue.typeName = nodeHelper.getIssueType(replica.type?.name, issue.projectKey)?.name ?: "Consulta"
5 }
6 issue.summary = replica.summary
7 issue.description = replica.description
8 issue.comments = commentHelper.mergeComments(issue, replica)
9 issue.attachments = attachmentHelper.mergeAttachments(issue, replica)
10 issue.comments = replica.comments
11 issue.resolution = replica.resolution
12 issue.attachments = replica.attachments
13 issue.priority = replica.priority
14
15 /*
16 User Synchronization (Assignee/Reporter)
17
18 Set a Reporter/Assignee from the source side, if the user can't be found set a default user
19 You can use this approach for custom fields of type User
20 def defaultUser = nodeHelper.getUserByEmail("default@idalko.com")
21 issue.reporter = nodeHelper.getUserByEmail(replica.reporter?.email) ?: defaultUser
22 issue.assignee = nodeHelper.getUserByEmail(replica.assignee?.email) ?: defaultUser
23 */
24
25 /*
26 Comment Synchronization
27
28 Sync comments with the original author if the user exists in the local instance
29 Remove original Comments sync line if you are using this approach
30 issue.comments = commentHelper.mergeComments(issue, replica){ it.executor = nodeHelper.getUserByEmail(it.author?.email) }
31 */
32
33 /*
34 Status Synchronization
35
36 Sync status according to the mapping [remote issue status: local issue status]
37 If statuses are the same on both sides don't include them in the mapping
38 def statusMapping = ["Open":"Now", "To Do":"Backlog"]
39 def remoteStatusName = replica.status.name
40 issue.setStatus(statusMapping[remoteStatusName] ?: remoteStatusName)
41 */
42
43 /*
44 Custom Fields
45
46 This line will sync Text, Option(s), Number, Date, Organization, and Labels CFs
47 For other types of CF check documentation
48 issue.customFields."CF Name".value = replica.customFields."CF Name".value
49 */
50

```

Copy incoming processor to olipboard

Figura 73: Campos sincronizados de Exalate

k. Enlace al Proyecto

PROYECTO:

[Jeremías IT - Colas - Proyecto de asistencia - Jira Service Management \(atlassian.net\)](#)

PORTAL:

<https://grupofjeremias2.atlassian.net/servicedesk/customer/portal/1>

11.3. ANEXO 3: PROYECTO JIRA SOFTWARE: DESARROLLO

a. Tipos de incidencia

Tipo de incidencia	Campos
Historia/Tarea/Sub-Tarea	Resumen* Tipo de Incidencia* Nivel de Seguridad Adjunto Descripción Informador Incidencias Enlazadas Versiones corregidas Responsable Prioridad Etickettas Sprint
Epic	Epic Name* Resumen* Tipo de Incidencia* Nivel de Seguridad

	<div>Adjunto</div> <div>Descripción</div> <div>Informador</div> <div>Incidencias Enlazadas</div> <div>Versiones corregidas</div> <div>Responsable</div> <div>Prioridad</div> <div>Etickettas</div>
Error	<div>Resumen</div> <div>Tipo de Incidencia</div> <div>Descripción</div> <div>Informador</div> <div>Versiones corregidas</div> <div>Prioridad</div> <div>Etickettas</div> <div>Nivel de Seguridad</div> <div>Eliminar</div> <div>Entorno</div> <div>Adjunto</div> <div>Versiones afectadas</div>

	Incidentes Enlazadas
	Responsable
	Epic Link
	Sprint

Tipos de incidencias

Nota: Aquí se ven los tipos de incidencias

b. Roles y grupos

Este proyecto también tiene solamente 2 roles asociados y 2 grupos creados. Un grupo está asociado a los seniors y el otro a los juniors. Los roles también son dos y se dividen de la misma manera.

Los seniors se encargarán de administrar todo el proyecto y comprobar que todo funcione correctamente, mientras que los juniors se encargarán de resolver todas las incidencias.

Rol	Grupo Correspondiente	Asignación
Programador Senior	Programadores Senior	<div><div>Añadir personas</div><div><div>Name, email or group</div><div><div><div>Programadores Senior</div><div>Introducir más</div><div></div></div></div><div>Rol</div><div><div>Programador Senior</div><div></div></div><div><div>Cancelar</div><div>Añadir</div></div></div><div>Figura 74: Asignando el rol al grupo</div></div>
Programador Junior	Programadores Junior	<div><div>Añadir personas</div><div><div>Name, email or group</div><div><div><div>Programadores Junior</div><div>Introducir más</div><div></div></div></div><div>Rol</div><div><div>Programador Junior</div><div></div></div><div><div>Cancelar</div><div>Añadir</div></div></div><div>Figura 75: Asignando el rol al grupo</div></div>

c. Workflows

En este proyecto solo se utilizan dos workflows, uno para las incidencias de tipo “**Historia**”, “**Error**”, “**Tarea**” y “**Sub-Tarea**” y el otro para las incidencias de tipo “**Épica**”.

Proyectos / Jeremías Desarrollo / Configuración del proyecto

Flujos de trabajo

JD: Software Simplified Workflow Scheme (1)

Agregar Flujo de Trabajo ▼ Cambiar el Esquema

Flujo de trabajo	Tipos de incidencia	Acciones
Flujo (Ver como texto / diagrama)	<div><div>■ Historia</div><div>■ Error</div><div>■ Tarea</div><div>■ Subtarea (Asignar)</div></div>	<div><div>✎</div><div>✕</div></div>
Flujo de Epicas (Ver como texto / diagrama)	<div><div>■ Epic (Asignar)</div></div>	<div><div>✎</div><div>✕</div></div>

Figura 76: Tipos de incidencias con sus workflows

Para el primer flujo de trabajo cuando se cree la incidencia tendrá el estado de “**Abierta**”, cuando se empiece a trabajar en ella se quedará en el estado “**En Progreso**”, cuando se termine la incidencia se pondrá en el estado “**En Revisión**” para que los QA puedan comprobar que todo funcione correctamente, cuando se revise si hay algo mal, se volverá a abrir, si no, pasará al estado “**Resuelta**” para que el programador Senior haga una revisión. Por último, cuando se cierre se pondrá en “**Cerrada**”.

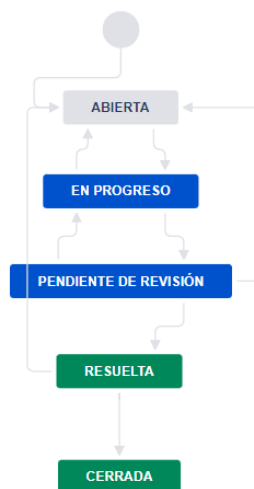


Figura 77: Workflow de historia, error, tarea y sub-trea

El segundo workflow es bastante más sencillo, ya que este tipo de incidencia solo se suele usar para agrupar otros tipos de incidencias, por lo tanto, esta incidencia no se cerrará hasta que todo su grupo las haya cerrado. Solo se compone de tres estados, **“Abierta”** para cuando se cree, **“En Progreso”** para cuando se empiecen a trabajar en las otras y por último **“Cerrado”** para que cuando acaben las otras incidencias, se pueda cerrar.



Figura 78: Workflow de épica

d. Permisos

Los permisos se distribuyen de dos maneras, el rol de Programador Senior será el administrador de todo el proyecto e irá guiando al equipo, mientras que el de Programador Junior se encargará de ir completando las incidencias que vayan surgiendo.

Por lo tanto, el Programador Senior tendrá todos los permisos del proyecto mientras que Programador Junior solo los necesarios para poder trabajar con las incidencias.

Permisos de proyecto	Usuarios / Grupos / Roles de proyecto
Permisos	
Administrar proyectos Capacidad para administrar un proyecto de JIRA.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Senior)
Buscar proyectos Capacidad para navegar por los proyectos y las incidencias de los mismos.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Junior) Rol de proyecto (Programador Senior)
Administrar sprints Capacidad para administrar sprints.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Senior)
Ver datos agregados NOUVO Los usuarios con este permiso podrán ver los datos combinados y resumidos del proyecto, independientemente de sus permisos individuales. Obtén más información.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Senior)
Ver herramientas de desarrollo Permite a los usuarios de un proyecto de software ver información vinculada con el desarrollo acerca de la incidencia, como por ejemplo confirmaciones, revisiones e información de compilaciones.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Senior)
Visualizar Flujo de Trabajo Los usuarios con este permiso pueden ver una versión de solo lectura del flujo de trabajo.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Senior) Rol de proyecto (Programador Junior)

Figura 79: Permisos de proyecto

Permisos en incidencias	Usuarios / Grupos / Roles de proyecto
Permisos	
Usuario asignable A usuarios con este permiso se les puede asignar una incidencia.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Junior)
Asignar incidencias Capacidad para asignar incidencias a otros usuarios.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Senior) Rol de proyecto (Programador Junior)
Cerrar incidencias Capacidad de cerrar incidencias. A menudo es útil que tus desarrolladores resuelvan incidencias y un departamento de calidad sea el encargado de realizar el cierre.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Senior) Rol de proyecto (Programador Junior)
Crear incidencias Capacidad para crear incidencias.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Senior) Rol de proyecto (Programador Junior)
Borrar incidencias Capacidad para borrar incidencias.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Senior)
Editar incidencias Capacidad para editar incidencias.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Junior) Rol de proyecto (Programador Senior)
Incidencias enlazadas Capacidad para enlazar las incidencias entre sí y crear incidencias enlazadas. Solo es útil si está habilitado elazar incidencias.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Junior) Rol de proyecto (Programador Senior)
Modificar Informador Capacidad para cambiar el creador al crear o editar una incidencia.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Senior) Rol de proyecto (Programador Junior)
Mover incidencias Capacidad de mover incidencias entre proyectos o entre flujos de trabajo dentro del mismo proyecto (si procede). Ten en cuenta que el usuario solo puede mover incidencias a un proyecto en que tenga permisos de creación de incidencias.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Junior) Rol de proyecto (Programador Senior)

Figura 80: Permisos de incidencias

Resolver incidencias Capacidad para resolver y reabrir incidencias. Esto incluye la capacidad para asignar una versión como versión con la solución de la incidencia.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Junior) Rol de proyecto (Programador Senior)
Programar incidencias La capacidad de ver o editar la fecha de entrega de una incidencia.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Junior) Rol de proyecto (Programador Senior)
Configurar la seguridad de la incidencia Capacidad de asignar el nivel de seguridad de una incidencia para que solo la gente con ese nivel de seguridad pueda ver la incidencia.	Rol de proyecto (Programador Senior) Rol de proyecto (Programador Junior) Rol de proyecto (atlassian-addons-project-access)
Desplazar incidencias Capacidad para desplazar incidencias.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Junior) Rol de proyecto (Programador Senior)
Permisos de Votación & Seguimiento	
Gestionar intereses Capacidad para gestionar los interesados de una incidencia.	Usuarios / Grupos / Roles de proyecto Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Junior) Rol de proyecto (Programador Senior)
Ver votantes y seguimientos Capacidad para ver los votantes y los interesados de una incidencia.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Junior) Rol de proyecto (Programador Senior)

Figura 81: Permisos de incidencias 2 y votación y seguimiento

Permisos en comentarios Permisos Añadir comentario Capacidad para comentar incidencias.	Usuarios / Grupos / Roles de proyecto Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Junior) Rol de proyecto (Programador Senior)
Borrar todos los comentarios Capacidad para borrar todos los comentarios.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Senior)
Borrar tus comentarios Capacidad para borrar tus comentarios.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Senior)
Editar todos los comentarios Capacidad para editar todos los comentarios.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Senior)
Editar tus comentarios Capacidad para editar tus comentarios.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Junior) Rol de proyecto (Programador Senior)
Permisos en adjuntos	
Permisos Crear archivos adjuntos Usuarios con este permiso pueden crear adjuntos.	Usuarios / Grupos / Roles de proyecto Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Junior) Rol de proyecto (Programador Senior)
Borrar todos los ficheros adjuntos Los usuarios con este permiso pueden borrar todos los adjuntos.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Senior)
Borrar tus adjuntos Los usuarios con este permiso pueden borrar sus propios adjuntos.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Junior) Rol de proyecto (Programador Senior)

Figura 82: Permisos de comentarios y adjuntos

Permisos de seguimiento del tiempo Permisos	Usuarios / Grupos / Roles de proyecto
Borrar todos los informes de trabajo Capacidad para eliminar todos los registros de trabajo.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Senior)
Borrar tus informes de trabajo Capacidad para borrar tus informes de trabajo.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Junior) Rol de proyecto (Programador Senior)
Editar todos los informes de trabajo Capacidad para editar todos los registros de trabajo.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Senior)
Editar tus informes de trabajo Capacidad para editar tus propios registros de trabajo.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Junior) Rol de proyecto (Programador Senior)
Trabajar en incidencias Capacidad para crear informes del trabajo realizado respecto a una incidencia. Solo es útil si el control del tiempo está habilitado.	Rol de proyecto (atlassian-addons-project-access) Rol de proyecto (Programador Senior) Rol de proyecto (Programador Junior)

Figura 83: Permisos de seguimiento del tiempo

e. *Enlace al proyecto*

[Tablero JD - Tablero ágil - Jira \(atlassian.net\)](#)

12. BIBLIOGRAFÍA

- [Learn Jira, Confluence, Trello, Jira Service Management, Jira Align, and more Atlassian software with Atlassian University : Atlassian](#)
- [Atlassian Developer](#)
- [The Jira Software Cloud REST API \(atlassian.com\)](#)