



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

DSIC
DEPARTAMENT DE SISTEMES
INFORMÀTICS I COMPUTACIÓ

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Dpto. de Sistemas Informáticos y Computación

Desarrollo de una aplicación web para la administración y
comunicación de dispositivos SAI

Trabajo Fin de Máster

Máster Universitario en Ingeniería y Tecnología de Sistemas
Software

AUTOR/A: Caicedo Palacios, Alejandro

Tutor/a: Penadés Gramage, María Carmen

CURSO ACADÉMICO: 2021/2022

Dedicatoria

A mis padres, Alejandro Caicedo y Martha Palacios por su empeño en formar el hombre que soy hoy día, llenando mi ser de valores como el amor, respeto, empatía, honestidad, gratitud y responsabilidad.

Y a mis hermanas que nunca han faltado un solo día en mi vida.

Dios bendiga mi familia.

Agradecimientos

A Dios por la vida que me ha dado, a mi familia por el amor con el que me ha criado, a mi tía Cristina Loffroy por el cariño que siempre me ha brindado.

Y a dos personas importantes en todo este proceso, las cuales confiaron en mí, Javier Salavert Fernández y María Carmen Penadés Gramage.

Tabla de Contenido

Índice de Figuras.....	9
Índice de Tablas.....	11
CAPÍTULO 1: Introducción.....	13
1.1. Motivación.....	13
1.2. Objetivos	14
1.2.1. Objetivo general.....	14
1.2.2. Objetivos específicos.....	14
1.3. Planificación	14
1.4. Estructura del documento.....	16
CAPÍTULO 2: Antecedentes	17
2.1. Infraestructura DSIC.....	17
2.1.1. SAI	18
2.1.2. Máquinas virtuales.....	20
2.1.3. Máquinas físicas.....	21
2.2. Problemática general	21
2.2.1. Problemática de apagado	22
2.2.2. Problemática de encendido	22
2.2.3. Problemática de tiempo limite.....	23
2.2.4. Problemática de infraestructura dividida	24
CAPÍTULO 3: Ges-SAI.....	25
3.1. Descripción	25
3.1.1. Funcionalidades	25
3.1.2. Usuarios	27
3.2. Arquitectura del sistema	28
3.3. Metodología	30
3.4. Tecnologías Frontend.....	31
3.5. Tecnologías Backend.....	31
3.6. Herramientas	32
3.7. Backlog	33
CAPÍTULO 4: Sprint-01	35
4.1. Antecedentes.....	35
4.1.1. Protocolo SNMP	35
4.1.2. Interrogar dispositivos SAI	38
4.1.1. Pools.....	39
4.1.2. Apagar y encender máquinas virtuales	40

4.1.3.	Apagar y encender máquinas físicas	42
4.2.	Backlog de historias de usuario	44
4.3.	Funcionalidades	44
4.3.1.	CRUD Modelo SAI	44
4.3.2.	CRUD Modelo Pool	48
4.3.3.	CRUD Modelo Virtual Machine	52
4.3.4.	Consumir API XenAPI	54
4.3.5.	Conexión con protocolo SNMP	55
CAPÍTULO 5: Sprint-02		57
5.1.	Antecedentes.....	57
5.1.1.	Máquinas con dependencia	57
5.1.2.	Máquinas sin dependencia	60
5.1.3.	Proceso de Multihilos.....	60
5.1.4.	Proceso de apagado de Máquinas en hilos.....	61
5.1.5.	Proceso de encendido de Máquinas en hilos.....	63
5.2.	Backlog de historias de usuario	65
5.3.	Funcionalidades	65
5.3.1.	CRUD Modelo Host	65
5.3.2.	CRUD Modelo Dependencias	69
5.3.3.	CRUD Modelo User	71
5.3.4.	Funcionalidades auxiliares	75
5.3.5.	Login API.....	76
CAPÍTULO 6: Sprint-03		77
6.1.	Backlog de historias de usuario	77
6.2.	Funcionalidades	78
6.2.1.	Login.....	78
6.2.2.	Interfaz Módulo SAI	79
6.2.3.	Interfaz Módulo Pool	80
6.2.4.	Interfaz Módulo Máquinas.....	82
6.2.5.	Interfaz Módulo Usuarios.....	84
6.2.6.	Interfaz Módulo Dependencias.....	86
6.2.7.	Interfaz Módulo Baterías	87
6.2.8.	Interfaz Módulo Instrucciones	88
CAPÍTULO 7: Pruebas		91
7.1.	Pruebas del Backend API.....	91
7.2.	Pruebas del Frontend.....	92
7.3.	Pruebas de API XenAPI.....	92
CAPÍTULO 8: Conclusión		95
8.1.	Conclusiones	95
8.2.	Trabajos futuros.....	96
Bibliografía		97
Anexos		99
Anexo A: Casos de uso		100

Anexo B: Manual de usuario Ges-SAI..... 113

Índice de Figuras

Figura 1 - Cebolla de planificación (Miguel, 2016)	14
Figura 2 - Cronograma de alto nivel	15
Figura 3 - Infraestructura general DSIC	17
Figura 4 - Interfaz de usuario Easy UPS 3S (Easy UPS 3S Manuals, s.f.).....	18
Figura 5 - Componentes de un Pool.....	20
Figura 6 - Forma incorrecta y correcta de apagado	22
Figura 7 - Forma incorrecta y correcta de encendido	23
Figura 8 - Diagrama de apagado no secuencial	23
Figura 9 - Diagrama de división de sedes	24
Figura 10 - Ges-SAI como capa intermedia	25
Figura 11 - Diagrama de contexto	28
Figura 12 - Arquitectura del sistema.....	30
Figura 13 - Backlog general del proyecto.....	34
Figura 14 - Diagrama básico de comunicación SNMP (ManageEngine, 2022).....	36
Figura 15 - Diagrama de árbol de MIB (ManageEngine, 2022).....	37
Figura 16 – Función para interrogar dispositivo SAI	39
Figura 17 - Diagrama ejemplo de alto nivel de un Pool	40
Figura 18 – Diagrama de comunicación con XenAPI	41
Figura 19 - Diagrama envío de paquete mágico (WoL)	43
Figura 20 - Sprint Backlog de historias de usuario, Sprint-01.....	44
Figura 21 - Endpoint Crear SAI.....	45
Figura 22 - Endpoint Actualizar SAI	46
Figura 23 - Endpoint Listar SAIS	47
Figura 24 - Endpoint Eliminar SAI.....	48
Figura 25 - Endpoint Crear Pool.....	49
Figura 26 - Endpoint Actualizar Pool	50
Figura 27 - Endpoint Listar Pools.....	51
Figura 28 - Endpoint Sincronizar Pool	51
Figura 29 - Endpoint Eliminar Pool.....	52
Figura 30 - Endpoint Crear máquina virtual	53
Figura 31 - Endpoint Eliminar máquina virtual.....	54
Figura 32 - Endpoint Interrogar al SAI con protocolo SNMP	55
Figura 33 - Dependencia uno a uno	58
Figura 34 - Dependencia uno a muchos.....	58
Figura 35 - Árboles de dependencia y dependencia entre sedes.....	59
Figura 36 - Árbol de dependencias	59
Figura 37 - Hilos y procesos	60
Figura 38 - Diagrama de flujo, Apagar máquina	61
Figura 39 - Árbol de dependencia, flujo de apagado de máquinas.....	62
Figura 40 - Diagrama de flujo, Encender máquina.....	63
Figura 41 - Árbol de dependencia, flujo de encendido de máquinas.....	64
Figura 42 - Sprint Backlog de historias de usuario, Sprint-02.....	65
Figura 43 - Endpoint Crear máquina física.....	66
Figura 44 - Endpoint Actualizar máquina física	67
Figura 45 - Endpoint Listar máquinas físicas	68
Figura 46 - Endpoint Eliminar máquina física.....	69

Figura 47 - Endpoint Crear dependencia	69
Figura 48 - Endpoint Eliminar dependencia	70
Figura 49 - Endpoint Listar dependencias	71
Figura 50 - Endpoint Crear usuario	72
Figura 51 - Endpoint Actualizar usuario.....	73
Figura 52 - Endpoint Listar usuarios	74
Figura 53 - Endpoint Eliminar usuario	75
Figura 54 - Endpoint Consultar máquinas dependientes de otra	75
Figura 55 - Endpoint Existencia de una máquina	76
Figura 56 - Sprint Backlog de historias de usuario, Sprint-03.....	77
Figura 57 - Wireframe formulario de login	78
Figura 58 - Wireframe Sai	79
Figura 59 - Wireframe Sais.....	80
Figura 60 - Wireframe Pool	81
Figura 61 - Wireframe Pools.....	82
Figura 62 – Wireframe Máquina.....	83
Figura 63 - Wireframe Máquinas.....	84
Figura 64 - Wireframe Usuario.....	85
Figura 65 - Wireframe Usuarios	86
Figura 66 - Wireframe Dependencias	87
Figura 67 - Wireframe Grafica del nivel restante de baterías.....	88
Figura 68 - Wireframe Instrucciones	89
Figura 69 - Prueba de Backend, Obtener el token de login	91
Figura 70 - Prueba de Backend, Obtener todas las Máquinas	92
Figura 71 - Prueba de XenAPI, Obtener la sesión de Api	93
Figura 72 - Prueba de XenAPI, Obtener todas las máquinas virtuales.....	93

Índice de Tablas

Tabla 1 - Indicadores LED de estado (Easy UPS 3S Manuals, s.f.)	19
Tabla 2 - Lista de OID relevantes (Harman, 2019)	38
Tabla 3 - HU01 Crear SAI	45
Tabla 4 - HU02 Actualizar SAI	46
Tabla 5 - HU03 Listar SAIS	47
Tabla 6 - HU04 Eliminar SAI	47
Tabla 7 - HU05 Crear Pool	48
Tabla 8 - HU06 Actualizar Pool	49
Tabla 9 - HU07 Listar Pools	50
Tabla 10 - HU08 Sincronizar Pool	51
Tabla 11 - HU09 Eliminar Pool	52
Tabla 12 - HU10 Crear máquina virtual	52
Tabla 13 - HU11 Eliminar máquina virtual	53
Tabla 14 - HU12 Encender máquina virtual	54
Tabla 15 - HU13 Apagar máquina virtual	54
Tabla 16 - HU14 Interrogar al SAI con protocolo SNMP	55
Tabla 17 - HU15 Crear máquina física	66
Tabla 18 - HU16 Actualizar máquina física	67
Tabla 19 - HU17 Listar máquinas físicas	68
Tabla 20 - HU18 Eliminar máquina física	68
Tabla 21 - HU19 Crear dependencia	69
Tabla 22 - HU20 Eliminar dependencia	70
Tabla 23 - HU21 Listar dependencias	70
Tabla 24 - HU22 Crear usuario	71
Tabla 25 - HU23 Actualizar usuario	72
Tabla 26 - HU24 Listar usuarios	73
Tabla 27 - HU25 Eliminar usuario	74
Tabla 28 - HU26 Consultar máquinas dependientes de otra	75
Tabla 29 - HU27 Existencia de una máquina	76
Tabla 30 - HU28 Crear login de sesión por token	76
Tabla 31 - HU29 Formulario de login	78
Tabla 32 - HU30 Crear o Editar SAI	79
Tabla 33 - HU31 Listar SAIS	79
Tabla 34 - HU33 Listar Pools	81
Tabla 35 - HU34 Crear o Editar Máquinas físicas	82
Tabla 36 - HU35 Listar Máquinas físicas	83
Tabla 37 - HU36 Crear o Editar Usuario	84
Tabla 38 - HU37 Listar Usuarios	85
Tabla 39 - HU38 Crear dependencia	86
Tabla 40 - HU39 Eliminar dependencia	86
Tabla 41 - HU40 Listar árbol de dependencia	87
Tabla 42 - HU41 Visualizar nivel de baterías	88
Tabla 43 - HU42 Descargar archivo de clave pública	89

CAPÍTULO 1:

Introducción

En este capítulo se expone la motivación que dio inicio al desarrollo del proyecto, acompañada por los objetivos que se desean alcanzar, junto con una planificación para lograrlos, finalmente en el último apartado se presenta la estructura que sigue el documento.

1.1. Motivación

Hoy en día las organizaciones que cuentan con servicios de redes informáticas, es bastante común que dispongan de un Sistema de Alimentación Ininterrumpida (SAI), en inglés llamado *Uninterruptible Power Supply* (UPS). Se trata de un dispositivo que almacena energía en sus baterías y se activa si se produce un corte en el sistema de corriente eléctrica, protegiendo así a todos los dispositivos conectados a él. Como se menciona, estos SAI disponen de un sistema de baterías y otros almacenadores de energía que durante un apagón eléctrico son capaces de proporcionar energía eléctrica durante un tiempo limitado. Estos equipos también tienen la función de mejorar la calidad de energía eléctrica que llega a los diferentes dispositivos conectados, filtrando así las subidas y bajadas de tensión y eliminando los armónicos que se producen en la red en caso de usar corriente alterna. Dicho esto, el papel principal de un SAI es el de suministrar potencia eléctrica si hay una anomalía en la red o un fallo de suministro, en un intervalo de tiempo “corto”.

El autor de este Trabajo Fin de Máster (TFM), además de ser estudiante del Máster en Ingeniería y Tecnología de Sistemas Software (MUITSS), de la Universidad Politécnica de Valencia (UPV), es becario de formación del departamento de sistemas informáticos y computación (DSIC), y mi función es participar junto con los técnicos del departamento en desarrollos web con Django y la gestión de proyectos de desarrollo con DevOps. El DSIC se encarga de ofrecer diferentes Másteres universitarios a alumnos egresados provenientes de carreras informáticas, y cuentan con un personal de docencia, investigación y equipo directivo. Por todo ello, la infraestructura informática que se necesita para soportar los servicios que la organización requiere, es bastante amplia, para así poder responder eficientemente a las necesidades de todo el personal que hace parte del departamento.

La motivación de realizar este TFM surge a raíz de un fallo eléctrico en la red del departamento DSIC en temporada de vacaciones, en este momento el tiempo de pérdida de luz se prolongó tanto que las baterías del SAI llegaron a agotarse ocasionando un apagón forzoso a todas las máquinas conectadas a él. Es importante para el departamento conocer el estado y carga de las baterías para una correcta toma de decisiones, en caso de que el corte de la corriente persista o se repita de forma intermitente en cortos periodos de tiempo, por lo que se ha tomado la decisión de desarrollar una aplicación web la cual permita administrar automáticamente el apagado y encendido de todas las máquinas físicas y máquinas virtuales que se encuentren conectadas a un SAI.

1.2. Objetivos

El proyecto desarrollado en este TFM tiene como finalidad poder prevenir el apagado forzado de los dispositivos del departamento DSIC evitando así daños en el hardware de estos, esto se desea lograr con los siguientes objetivos presentados. Al proyecto lo denominaremos Ges-SAI.

1.2.1. Objetivo general

Desarrollar una aplicación web que permita visualizar el nivel de batería de los dispositivos SAI del DSIC y a su vez gestione la automatización del apagado y arranque de los servidores físicos y virtuales.

1.2.2. Objetivos específicos

- Prevenir daños de hardware en los servidores del DSIC.
- Interrogar al SAI desde la aplicación web por medio del protocolo SNMP.
- Gestionar el apagado y encendido a las máquinas de forma jerárquica con el fin de no interrumpir procesos y/o bloqueos de éstas.
- Implementar Azure DevOps para gestionar y administrar todo el ciclo de vida del proyecto, incluyendo los repositorios del código y su documentación.

1.3. Planificación

El proyecto Ges-SAI se desarrollará utilizando metodologías ágiles, por lo que su planificación está basada en una planificación por capas o también llamada cebolla de planificación agile, se opta por desarrollarse de esta manera, ya que nos permite tener una mayor flexibilidad gracias a los sprints, los cuales aumentan constantemente la retroalimentación, consiguiendo así mejoras continuas para el proyecto. Aparte de contar con estas ventajas, esta planificación también permite elegir el nivel adecuado para empezar con el desarrollo del producto, en la Figura 1 se puede ver un ejemplo de los niveles de planificación para un desarrollo agile.

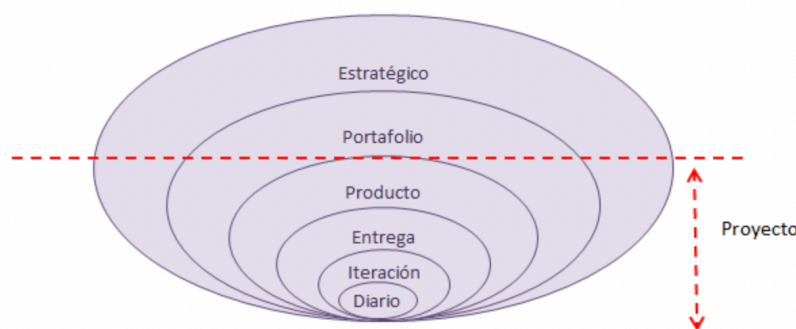


Figura 1 - Cebolla de planificación (Miguel, 2016)

Como se puede ver en la imagen anterior la planificación por capas cuenta con 6 niveles, el desarrollo de este proyecto será enfocado en los últimos 4 niveles que se detallan a continuación:

1. **Producto:** Se realiza el Backlog del producto, el cual es una lista de trabajo que se ordena por prioridades y se obtiene de los requisitos (historias de usuario) del proyecto.

2. **Entrega:** Se establecen las fechas de entrega y se planifica cuando deberían estar disponibles para el cliente final, todo esto basado en la duración de las iteraciones.
3. **Iteración:** Se centra solo en el trabajo que debe realizarse en la siguiente iteración, ésta se hace durante la reunión de planificación del sprint en la que se crea el Sprint Backlog, el cual no es más que una lista de historias de usuario seleccionadas del Backlog del producto para realizarse en el sprint actual.
4. **Diario:** reunión diaria del equipo de desarrollo, donde se discute el progreso que se ha tenido respecto a lo que se planeó en el nivel de iteración.

Con el fin de seguir una hoja de ruta para el desarrollo del proyecto, se ha creado un cronograma de alto nivel el cual presenta la secuencia de entregas a lo largo del tiempo (Figura 2).

Ges-SAI					
Plan de desarrollo del Software					
Color	Numero tarea	Fase / Nombre de la tarea	Fecha de inicio	Fecha de finalización	Duración (días hábiles)
	1	Modelado del proyecto	10/1/22	28/1/22	15
	1.1	Análisis de requisitos	10/1/22	13/1/22	4
	1.2	Negociación de compromisos	14/1/22	18/1/22	3
	1.3	Especificación de los casos de uso	19/1/22	24/1/22	4
	1.4	Planificación del proyecto	25/1/22	28/1/22	4
	2	Sprint-01	1/2/22	18/2/22	14
	2.1	Diseño de base de datos	1/2/22	3/2/22	3
	2.2	Desarrollo Backend	4/2/22	18/2/22	11
	3	Sprint-02	21/2/22	11/3/22	15
	3.1	Desarrollo Backend	21/2/22	8/3/22	12
	3.2	Diseño de interfaz de usuario	9/3/22	11/3/22	3
	4	Sprint-03	14/3/22	1/4/22	15
	4.1	Desarrollo Frontend	14/3/22	1/4/22	15
	5	Pruebas y documentación	1/4/22	29/4/22	21
	5.1	Pruebas unitarias	1/4/22	7/4/22	5
	5.2	Prueba de funcionamiento	8/4/22	14/4/22	5
	5.3	Posibles cambios	15/4/22	22/4/22	6
	5.4	Documentación / manuales	25/4/22	29/4/22	5
	6	Despliegue	2/5/22	31/5/22	22
	6.1	Planificación de despliegue	2/5/22	6/5/22	5
	6.2	Despliegue en producción	9/5/22	31/5/22	17

Figura 2 - Cronograma de alto nivel

1.4. Estructura del documento

Este documento cuenta con 8 capítulos distribuidos de la siguiente manera:

- **Introducción:** En este capítulo se menciona la motivación que surgió para la realización de este proyecto, los objetivos a lograr para un feliz término y la planificación con fechas estimadas para cumplirlos.
- **Antecedentes:** En este capítulo se detalla la infraestructura del DSIC junto con las máquinas físicas, máquinas virtuales y SAIs que la componen. Por último, se describen las diferentes problemáticas del proyecto.
- **Ges-SAI:** En este capítulo se especifica la solución propuesta al proyecto, acompañada por las funcionalidades, usuarios y diagrama de contexto de la aplicación, junto con esto se detalla también la arquitectura del sistema, metodología a seguir y las tecnologías y herramientas usadas para el desarrollo del Backend y Frontend.
- **Sprint-01:** En este capítulo se presentan los antecedentes que aseguran el flujo lógico de las ideas para el desarrollo de las diferentes funcionalidades, el sprint backlog y las historias de usuario agrupadas por funcionalidades y desarrolladas en el sprint-01.
- **Sprint-02:** En este capítulo se presentan los antecedentes que aseguran el flujo lógico de las ideas para el desarrollo de las diferentes funcionalidades, el sprint backlog y las historias de usuario agrupadas por funcionalidades y desarrolladas en el sprint-02.
- **Sprint-03:** En este capítulo se presenta el sprint backlog, las historias de usuario acompañadas de los Wireframes y agrupadas por las diferentes funcionalidades desarrolladas en el sprint-03.
- **Pruebas:** En este capítulo se detallan las pruebas realizadas para validar el correcto funcionamiento de la aplicación.
- **Conclusión:** En este capítulo se presentan las conclusiones del trabajo indicando los objetivos cumplidos y trabajos futuros.

CAPÍTULO 2: Antecedentes

En este capítulo se presentan los antecedentes del proyecto Ges-SAI englobados en dos grandes apartados, el primero de ellos describe la infraestructura del departamento, en el cual se habla de los dispositivos SAI, máquinas virtuales y máquinas físicas, por último, en el segundo apartado se exponen las diferentes problemáticas encontradas, las cuales ayudan a definir el rumbo del proyecto.

2.1. Infraestructura DSIC

El departamento DSIC cuenta con una red compuesta con alrededor de 130 máquinas físicas. Esta infraestructura se encuentra dividida por dos edificios, edificio 1F y edificio 1B, el 1F cuenta con planta baja y primer piso y el 1B cuenta solo con planta baja. Aunque físicamente la infraestructura de red este separada por dos edificios, esto no quiere decir que sea independiente ya que se encuentran comunicados por medio de routers con fibra óptica de 10 GB.

Actualmente el edificio 1F cuenta con 2 dispositivos SAI, el primero ubicado en la planta baja y el segundo en el primer piso, ambos encargados de proveer de energía en caso de un corte eléctrico a las máquinas ubicadas en su piso. Por otro lado, el edificio 1B requiere solo de un dispositivo SAI ubicado en la planta baja (ver Figura 3).

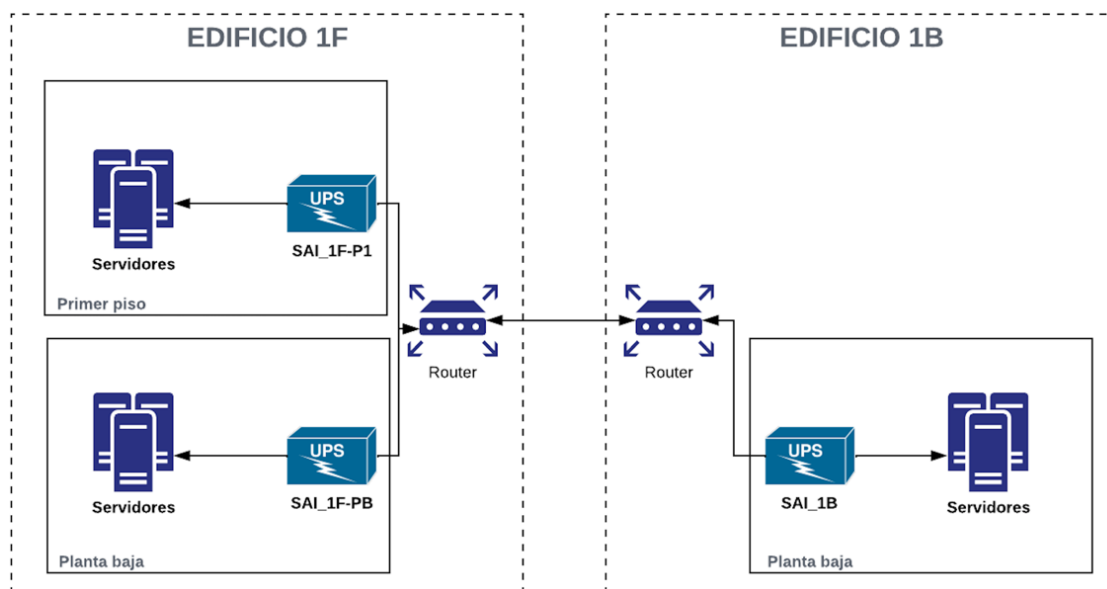


Figura 3 - Infraestructura general DSIC

La infraestructura del DSIC está compuesta por diferentes dispositivos, sin embargo, para este proyecto nos hemos centrado solo en 3 de ellos, los dispositivos SAI, las máquinas físicas y las máquinas virtuales.

2.1.1. SAI

Como se ha mencionado anteriormente los dispositivos SAI son capaces de proporcionar protección contra problemas eléctricos o cortes de luz, el departamento cuenta con 3 de estos dispositivos, los cuales están distribuidos en los 2 edificios anteriormente mencionados.

La principal razón por la que el DSIC cuenta con estos dispositivos, es la de proteger sus equipos, ya que los SAI acumulan energía eléctrica en su interior mediante un conjunto de baterías y esta energía es suministrada a los equipos en cuanto se produce un apagón, cortes de suministro eléctrico u otros problemas, y esto impide así posibles averías en los dispositivos conectados a la red.

Pero este no es el único servicio que puede prestar un SAI, también se encarga de que la energía que llega a un ordenador llegue de una forma constante, sin picos y bajos de intensidad, de esta manera protege componentes como la fuente de alimentación o la placa base. Entonces los fallos que puede prever un SAI son los siguientes:

- Cortes de energía
- Suministro inestable de energía
- Picos y caídas de tensión
- Sobre tensiones prolongadas
- Distorsión de la señal de la corriente

En el mercado actualmente existen diferentes tipos de dispositivos SAI y el departamento cuenta con tres de ellos fabricados bajo la marca Schneider Electric y de modelo Easy UPS 3S. Estos dispositivos disponen de una tarjeta de red SNMP la cual por medio de este puerto se podrá efectuar la comunicación entre la aplicación y el SAI. Además de esto cada uno tiene una interfaz de usuario con indicadores led los cuales se presentan a continuación en caso de que el SAI requiera una administración más directa (Castillo, 2019).

La Figura 4 presenta la de interfaz de usuario para el modelo Easy UPS 3S la cual cuenta con indicadores LED de estado en cada uno de los SAIs.

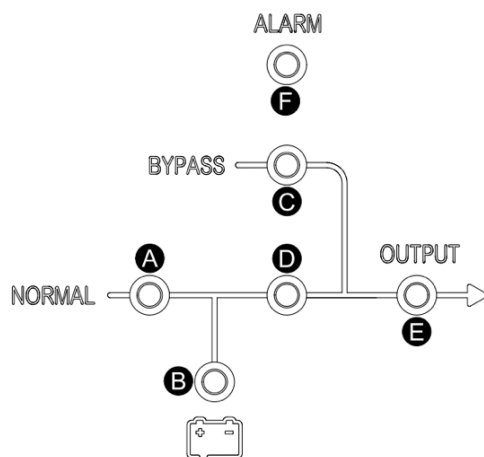


Figura 4 - Interfaz de usuario Easy UPS 3S (Easy UPS 3S Manuals, s.f.)

La Tabla 1 describe cada uno de los LED y los diferentes estados en los que se pueden encontrar el SAI.


























	LED	Estado
A	Rectificador	<p>Verde:  El rectificador funciona correctamente.</p> <p>Verde intermitente:  El rectificador funciona correctamente y la red eléctrica está normal.</p> <p>Rojo:  El rectificador no está operativo.</p> <p>Rojo intermitente:  No hay red eléctrica.</p> <p>OFF:  El rectificador está apagado.</p>
B	Batería	<p>Verde:  La batería se está cargando.</p> <p>Verde intermitente:  La batería se está descargando.</p> <p>Rojo:  La batería no está operativa.</p> <p>Rojo intermitente:  Tensión baja de la batería.</p> <p>OFF:  La batería y el cargador de la batería están normales; la batería no se carga.</p>
C	Derivación	<p>Verde:  Carga suministrada por fuente de derivación.</p> <p>Rojo:  La fuente de derivación no está disponible o el conmutador de derivación estática no está operativo.</p> <p>Rojo intermitente:  La tensión de derivación está fuera de tolerancia.</p> <p>OFF:  La fuente de derivación es normal.</p>
D	Inversor	<p>Verde:  Carga suministrada por el inversor.</p> <p>Verde intermitente:  Encendido, sincronización o inactividad (modo ECO) de al menos un módulo.</p> <p>Rojo:  Carga no suministrada por el inversor; el inversor no está operativo.</p> <p>Rojo intermitente:  Carga suministrada por el inversor, pero existe una alarma de inversor.</p> <p>OFF:  El inversor está apagado.</p>
E	Carga	<p>Verde:  La salida del SAI está encendida.</p> <p>Rojo:  Demasiado tiempo en sobrecarga en la salida del SAI, o bien la salida está en cortocircuito o no hay potencia de salida.</p> <p>Rojo intermitente:  Sobrecarga en la salida del SAI.</p> <p>OFF:  La salida del SAI está apagada.</p>
F	Estado	<p>Verde:  Modo normal.</p> <p>Rojo:  Estado no operativo.</p>

Tabla 1 - Indicadores LED de estado (Easy UPS 3S Manuals, s.f.)

2.1.2. Máquinas virtuales

Actualmente el departamento cuenta con dos sistemas de virtualización, el primero es XCP-ng el cual está basado en el popular hipervisor XenServer y se caracteriza por ser una plataforma de código abierto, permitiendo implementar rápidamente la virtualización de servidores y estaciones de trabajo. El segundo sistema es oVirt (Open Virtualización Manager) también de código abierto y basado principalmente en el hipervisor KVM, dirigido a ser un sistema de virtualización distribuida.

Las máquinas virtuales hacen parte fundamental de la infraestructura del DSIC, estas máquinas son agrupadas por Pools, Serafini (2011) afirma que: “*Un Pool se define como dos o más máquinas Citrix XenServer que componen una misma entidad, de modo que pueden compartir recursos como máquinas virtuales, almacenamiento, etc. de forma centralizada.*”. Las máquinas virtuales se ejecutan en máquinas físicas también llamadas (Host) y estas componen el Pool, a continuación, la Figura 5 muestra una imagen de ejemplo desde el cliente de XCP-ng.

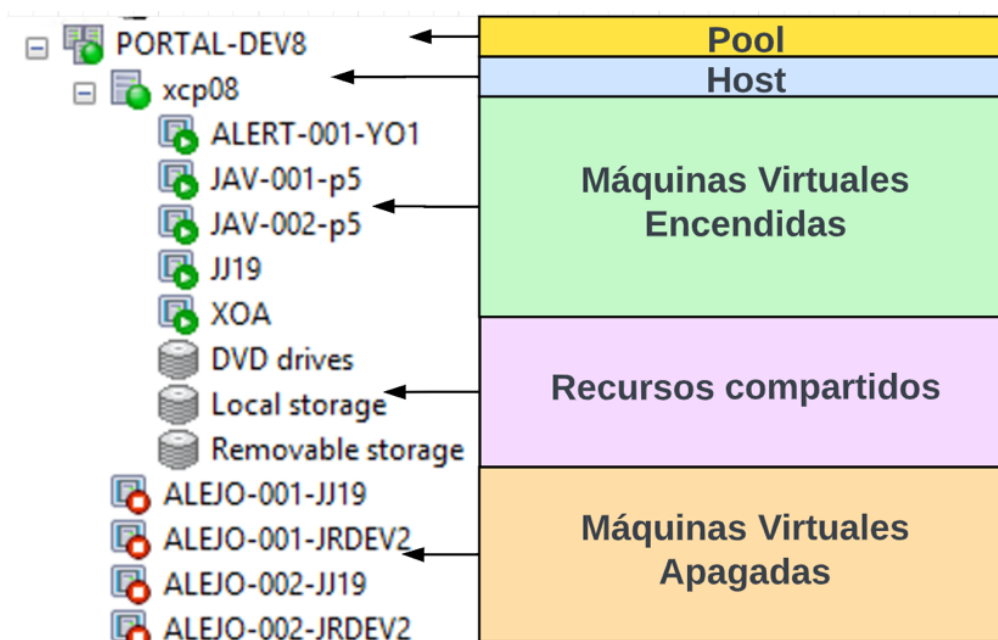


Figura 5 - Componentes de un Pool

Podríamos decir que actualmente en el DSIC existen 2 grupos de Máquinas Virtuales. Por una parte, están las que son permanentes, nunca se eliminan, nunca se apagan y prestan recursos compartidos y, por otra parte, están las que se crean diariamente ya sea para prácticas, pruebas o exámenes, estas últimas se pueden crear o eliminar las veces que se desee. EL departamento cuenta con diferentes pools ya establecidos, y la unión de estos puede llegar a tener entre 1.000 a 1.500 máquinas virtuales permanentes y no permanentes, los más relevantes son:

- **Pool DSIC:** Tiene alrededor de 30 máquinas virtuales permanentes.
- **Pool Portal:** Tiene alrededor de 6 máquinas virtuales permanentes y se utiliza para prácticas de laboratorio.
- **Pool Evir:** Tiene 2 máquinas virtuales permanentes y se utiliza para exámenes.

2.1.3. Máquinas físicas

Nos referimos a todas las maquinas físicas que pertenecen a la red del departamento como servidores y almacenamientos, estas máquinas se encuentran distribuidas en las diferentes plantas de ambos edificios.

El edificio 1F actualmente cuenta con alrededor de 100 máquinas conectadas al SAI ubicado en el primer piso y 35 máquinas conectadas al SAI ubicado en la planta baja. Y por último el edificio 1B tiene aproximadamente 38 máquinas conectadas al SAI. Estas máquinas físicas son las encargadas de componer los diferentes Pools de máquinas virtuales del DSIC, algunas encargadas del almacenamiento y otras encargadas de prestar diferentes recursos compartidos.

2.2. Problemática general

Aunque los casos de apagones de luz no son muy comunes, si son bastante complejos de evitar ya que estos pueden ser generados por diferentes causas como, por ejemplo:

- Operadores del servicio eléctrico: Internamente estas empresas pueden presentar sus fallos dejando a sus usuarios sin el servicio de electricidad.
- Tormentas eléctricas: Estos rayos pueden impactar en los transformadores, postes de servicios públicos o las diferentes líneas altas.
- Accidentes vehiculares: Estos accidentes ocurridos en calles locales es muy usual que provoquen roturas de postes de servicios públicos, ocasionando daños importantes y fallas en los equipos.
- Árboles y vegetación: También se pueden producir cortes de energía cuando los árboles interfieren o se caen sobre las líneas eléctricas, ya que las ramas pueden romperse como resultado del viento o la antigüedad.

Todos estos casos pueden resultar en un corte de energía, ocasionando daños importantes en la infraestructura. En el DSIC ocurrió un caso de corte de energía en agosto de 2021 que ocasionó daños en diferentes maquinas del departamento y fue debido a que el corte de energía se prolongó en el tiempo hasta agotarse las baterías de los dispositivos SAI los cuales se tienen para prevenir estos casos.

Es muy inusual que un corte de energía se prolongue por un largo tiempo, pero coincidió además que en el momento en que se produjo este apagón, el personal técnico del DSIC se encontraba de vacaciones. En esta ocasión, una vez que los dispositivos SAI detectaron que se había producido un corte en el suministro de energía estos se activaron alimentando los equipos conectados a ellos, proporcionándoles electricidad durante un tiempo limitado, pero puesto que la corriente eléctrica no se reanudó, las baterías se agotaran y dejaran de brindar energía a los equipos, por lo que no se pudo prevenir que los equipos se apagaran abruptamente y de manera forzada.

Aparte de los SAI con los que cuenta el departamento no hay otro método que evite que los equipos se apaguen indebidamente, ya que la batería de estos dispositivos es limitada, por lo que se necesita encontrar una solución con el fin de prevenir que los ordenadores del DSIC sufran daños si vuelve a ocurrir un caso de corte de energía prolongado.

2.2.1. Problemática de apagado

Aunque una solución para prevenir el apagado forzoso se pueda dar, también es necesario pensar en la forma de apagar todos los ordenadores de forma automática, ya que el departamento cuenta con un conjunto amplio de máquinas tanto físicas como virtuales, las cuales comparten recursos entre ellas. Por lo que, aunque tengamos la capacidad de apagar un ordenador de la manera correcta, no es prudente apagarlo si éste comparte recursos a una o más máquinas, lo adecuado sería encontrar la forma de que todos estos ordenadores, los cuales reciben recursos compartidos sean los primeros en apagarse, y por último apagar las máquinas que comparten estos recursos, evitando así que las máquinas dependientes de recursos se queden bloqueadas esperando conexión con la máquina principal. De acuerdo con lo dicho, la Figura 6 muestra el apagado incorrecto y correcto de las máquinas.

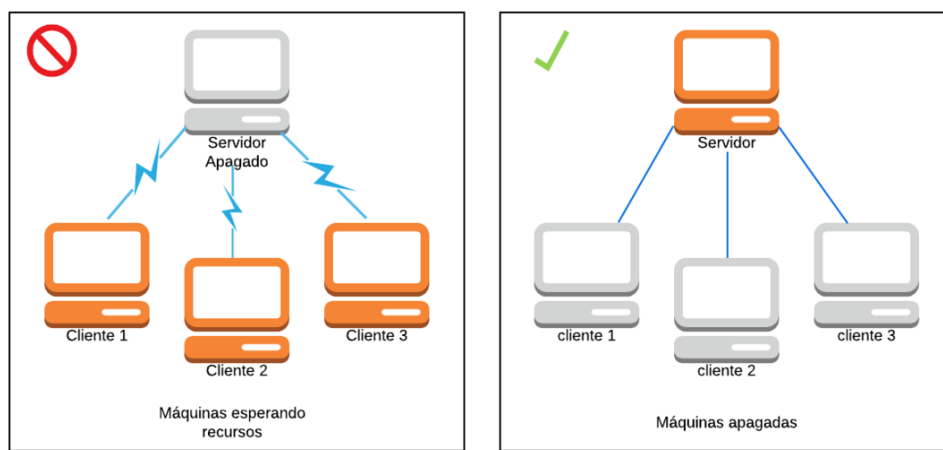


Figura 6 - Forma incorrecta y correcta de apagado

2.2.2. Problemática de encendido

De la misma forma que se desea encontrar una solución para la problemática del apagado de máquinas, también se debe encontrar una para el encendido automático de éstas. Como ya se mencionó anteriormente muchas de estas máquinas comparten recursos con otras, lo cual hace que si deseamos encender nuevamente las máquinas debamos aplicar la misma metodología del apagado pero esta vez de manera inversa.

Esto se debe hacer con el fin de que las máquinas dependientes de otras, no se queden bloqueadas esperando recursos de otras, entonces para este caso contrario se quiere encontrar una solución que permita encender primero todas las máquinas que prestan los recursos, y seguido de esto encender las máquinas dependientes para que el proceso de arranque concluya correctamente. A continuación, la Figura 7 presenta la forma incorrecta y correcta del encendido de las máquinas.

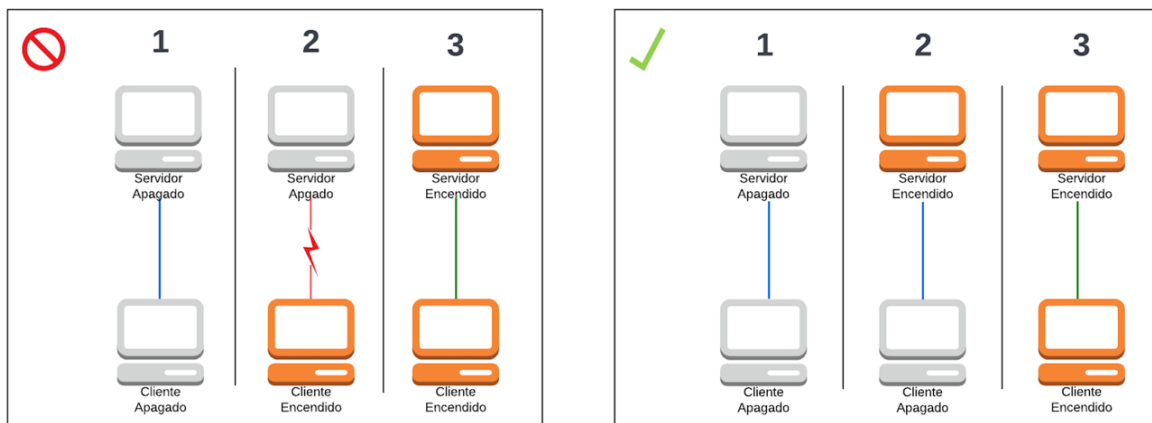


Figura 7 - Forma incorrecta y correcta de encendido

2.2.3. Problemática de tiempo limite

Como sabemos ya, los dispositivos SAI tienen un sistema de baterías el cual alimenta a las máquinas conectadas a él, sin embargo, estas baterías no son infinitas y cuentan con un límite de carga, una vez que estas se agotan el SAI no tiene cómo proveer de energía a las máquinas y éstas se apagarán.

En este caso, el DSIC es un departamento de tamaño medio-grande, contando con alrededor de 100 máquinas físicas que lo conforman, cada uno de estos ordenadores en un entorno normal tarda uno 15 a 20 segundos en apagarse, esto sin contar que la máquina caiga en un estado de actualización lo cual aumentará más el tiempo de espera del apagado. Por lo que no sería conveniente realizar un apagado secuencial a las máquinas ya que esto hará que el proceso tarde mucho más y se llegue al límite de tiempo permitido por las baterías, así que se desea encontrar una solución la cual permita apagar todas las máquinas antes de que las baterías se agoten.

Para entender esto (ver Figura 8) en la cual cada círculo representa una máquina y cada color representa un estado de la máquina, el ejemplo se presenta en 3 pasos diferentes, en el que se puede ver que las máquinas que se encuentren al mismo nivel de jerarquía pueden iniciar su apagado siempre y cuando la máquina con la que tengan una dependencia ya se encuentre apagada.

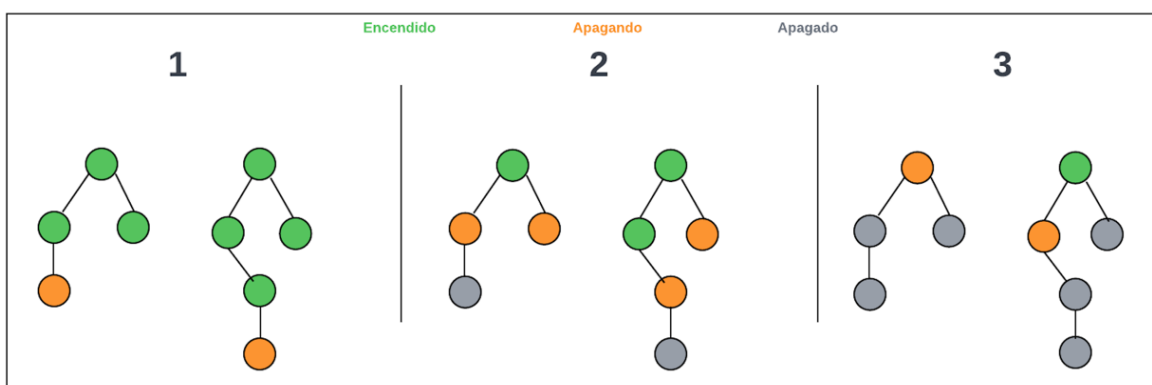


Figura 8 - Diagrama de apagado no secuencial

2.2.4. Problemática de infraestructura dividida

La infraestructura de TI del departamento DSIC se encuentra dividida en 2 sedes, sin embargo, todos sus dispositivos están conectados a la misma red, con lo cual, las máquinas están separadas físicamente en 2 edificios, pero no son independientes uno del otro, cada uno de estos edificios cuenta con su propio dispositivo SAI, el cual se encarga de proveer de energía a las máquinas conectadas a él en el caso de un apagón.

Esto hace que se genere una problemática importante para cubrir, ya que existen máquinas de un edificio que ofrecen recursos compartidos a máquinas del otro edificio (los edificios se denominan, como se ha dicho anteriormente, edificio 1B y edificio 1F). Para entender un poco mejor este caso es necesario explicarlo con un pequeño ejemplo.

Imaginemos que en el edificio 1F ocurre un corte de energía prolongado haciendo que salte la alarma para iniciar el proceso de apagado de máquinas, todos los ordenadores conectados al SAI-1F deben apagarse, sin embargo 2 de estos brindan recursos de almacenamiento a 3 máquinas que se encuentran en el edificio 1B conectadas al SAI-1B, estas 3 máquinas deberán también ser apagadas ya que si no se hace, estas quedarán de alguna forma bloqueadas esperando conexión de 2 máquinas que ya se han apagado, sin embargo, no sólo deberán apagarse estas 3 máquinas si no también todas las máquinas que dependan de ellas. La Figura 9 muestra una gráfica que representa este caso, cada círculo es una máquina y todas las máquinas del edificio 1F han sido apagadas junto con las máquinas dependientes del edificio 1B.

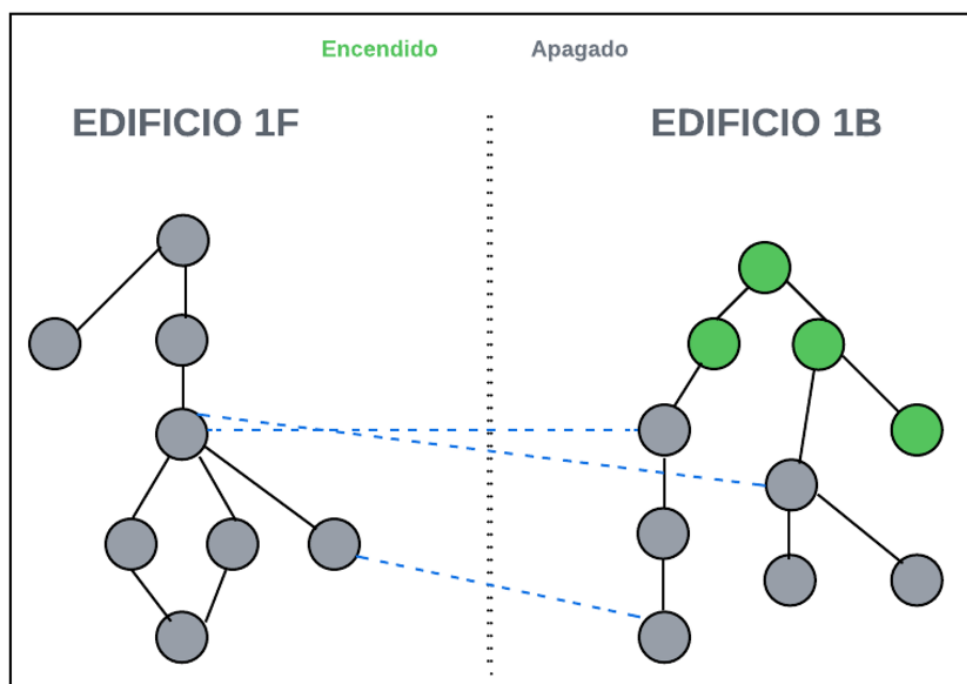


Figura 9 - Diagrama de división de sedes

CAPÍTULO 3:

Ges-SAI

Este capítulo presenta la solución propuesta, dando a conocer la funcionalidad principal del proyecto a desarrollar junto con los requisitos propuestos, casos de uso y arquitectura del sistema, por último, la metodología y tecnología utilizados para llegar satisfactoriamente a la entrega final del software.

3.1. Descripción

La funcionalidad principal de Ges-SAI es poder prevenir un apagado forzado a las máquinas físicas y virtuales conectadas a los diferentes SAIs. Para lograr esto, Ges-SAI funcionara como una capa intermedia entre los dispositivos SAI y las máquinas físicas y virtuales (ver Figura 10), interrogando constantemente al SAI por su nivel restante de carga y una vez la aplicación detecte que la batería que queda es mínima, procederá al apagado de los servidores.

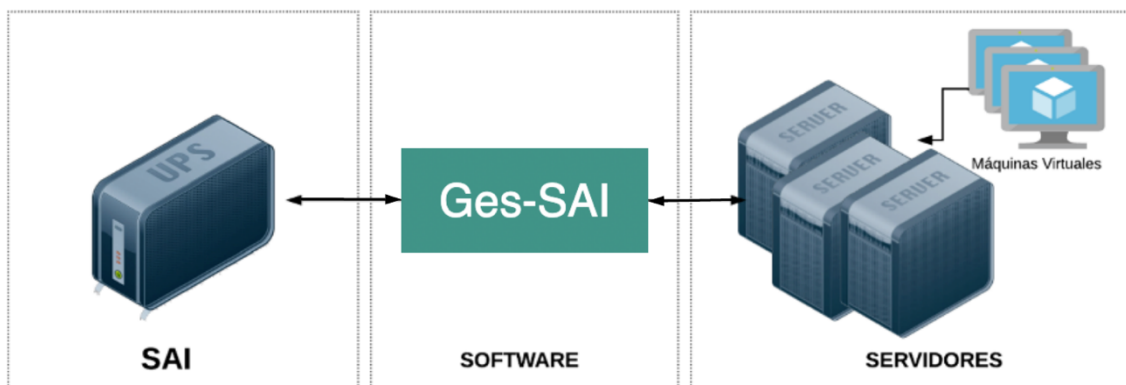


Figura 10 - Ges-SAI como capa intermedia

Ges-SAI será un software de administración y comunicación de los diferentes dispositivos SAI del DSIC, el cual gestionará la automatización del apagado y encendido de los servidores físicos y virtuales, mostrando en tiempo real el estado del SAI e indicando el nivel de las baterías y el tiempo de duración de estas. Aparte de poder administrar los SAIs y Pools, la aplicación también le permitirá al usuario administrar las diferentes dependencias entre las máquinas que desee, con el fin de incluirlas en la automatización del encendido y apagado.

3.1.1. Funcionalidades

Se presenta un listado con las funcionalidades agrupadas que han sido tomadas como requisitos de las diferentes reuniones con la tutora y los técnicos del DSIC. Se identifican 6 elementos destacables y para cada uno de ellos se enumeran las funcionalidades necesarias.

➤ Elemento SAI

Este elemento representa el modelo del objeto físico SAI del cual nos interesa conocer su nombre, credenciales de conexión, IP y Mac. Las funciones que este modelo tendrá son las siguientes.

- Crear
- Editar
- Listar
- Eliminar
- Consultar nivel de batería

➤ **Elemento Pool**

Este elemento representa el modelo del objeto físico Pool del cual también nos interesa conocer su nombre, usuario, contraseña, IP y Mac. Las funciones que este modelo tendrá son las siguientes.

- Crear
- Sincronizar
- Editar
- Listar
- Eliminar

➤ **Elemento Máquina física**

Este elemento representa el modelo del objeto Máquina física del cual nos interesa conocer su nombre, IP y Mac y sistema operativo. Las funciones que este modelo tendrá son las siguientes.

- Crear
- Editar
- Listar
- Eliminar
- Encender
- Apagar

➤ **Elemento Máquina virtual**

Este elemento representa el modelo del objeto Máquina virtual del cual nos interesa conocer su nombre, IP y Mac y Pool al que pertenece. Las funciones que este modelo tendrá son las siguientes.

- Crear
- Edita
- Listar
- Eliminar
- Encender
- Apagar

➤ **Elemento Dependencias**

Este elemento representa el modelo del objeto Dependencias del cual nos interesa conocer la máquina padre y la máquina hijo. Las funciones que este modelo tendrá son las siguientes.

- Crear
- Listar
- Eliminar

➤ **Elemento Usuario**

Este elemento representa el modelo del objeto Usuario del cual nos interesa conocer su nombre completo, nombre de usuario, contraseña, y rol de usuario. Las funciones que este modelo tendrá son las siguientes.

- Crear
- Editar
- Listar
- Eliminar

3.1.2. Usuarios

Ges-SAI contará con 2 roles de usuario, el técnico y el genérico, estos serán usuarios locales. Inicialmente se pensó tener un usuario por persona, con lo cual pudiera iniciar sesión en la aplicación siempre y cuando estuviera dado de alta en el sistema de autenticación del departamento, con esto cada usuario podría administrar únicamente sus máquinas.

Sin embargo, esto no es conveniente para los grupos de investigación, ya que estos comparten las mismas máquinas físicas y virtuales, es por esta razón que Ges-SAI cuenta con usuarios locales, los cuales tienen la ventaja de ser compartidos, dando acceso a la aplicación con un mismo usuario a diferentes personas que pertenecen al mismo grupo de investigación.

- **Rol Técnico:**

Este rol será asignado para los administradores del sistema, los cuales contarán con acceso a todos los módulos de la aplicación. Las funcionalidades que podrá tener son las siguientes:

- Administración de Sais
- Administración de Pools
- Administración de Máquinas
- Administración de Dependencias
- Administración de Usuarios
- Visualización de Baterías
- Instrucciones de integración de máquinas

- **Rol Genérico:**

Este rol será asignado a los diferentes grupos de investigación que requieran la administración del apagado y encendido de sus máquinas. Las funcionalidades que podrá tener son las siguientes:

- Administración de Máquinas
- Administración de Dependencias
- Administración de Pools
- Instrucciones de integración de máquina

En la Figura 11 se presentan estos roles como diagrama de contexto en UML, identificando los límites y entorno del sistema Ges-SAI, junto con los actores externos que van a interactuar con el sistema.

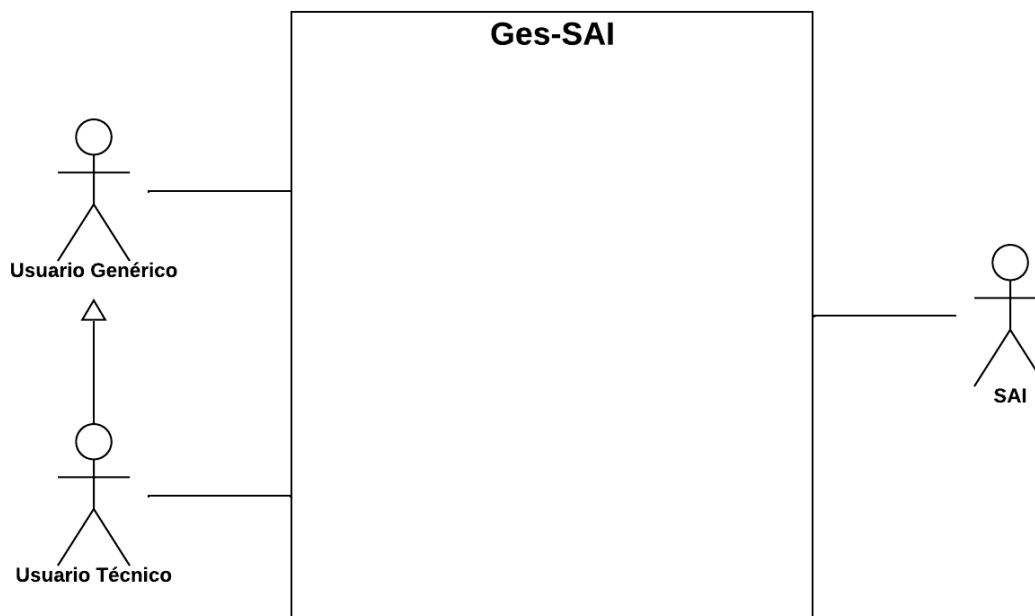


Figura 11 - Diagrama de contexto

3.2. Arquitectura del sistema

Ges-SAI será una aplicación web y estará basada en una arquitectura cliente/servidor. Esta aplicación solo será utilizada por miembros del DSIC y será ejecutada desde la propia intranet o mediante acceso VPN al departamento. Este software estará compuesto por dos aplicaciones, la primera será una aplicación web encargada de toda la funcionalidad del Frontend y la segunda una API REST como servicio Backend encargada de devolver los datos y conectarse a otros servicios.

La arquitectura estará conformada por los siguientes elementos:

➤ **Frontend**

Se encargará de ser la parte visible con la que los usuarios interactuarán con la aplicación, esta se ejecutará en el navegador web del cliente y se comunicará con el Backend por medio de peticiones GET y POST.

➤ **Backend**

Será la parte oculta al usuario y se encargará de ser la capa de acceso a los datos, se construirá como una API REST con el fin de ser consumida por la aplicación Frontend.

➤ **Script de interrogación al SAI**

Dentro del Backend se encontrarán 2 Scripts los cuales se ejecutarán como cron cada cierto tiempo, estos se encargarán de interrogar el nivel de batería restante de los SAIS, con el fin de iniciar el proceso de apagado o encendido de las máquinas virtuales y físicas.

➤ **Base de datos local**

Ges-SAI contará con una base de datos local, la cual se encargará de almacenar toda la información de los diferentes modelos de la aplicación.

- **Base de datos externa**
Con el fin de ser utilizada como una base de datos de sólo consulta, se hará uso de la información de una base de datos ya existente que funciona para toda la UPV a la que se consultará una única tabla, la cual contiene información de las máquinas físicas dadas de alta.
- **Api XenAPI**
Por medio de esta API, la cual será consumida por el Backend, se podrá tener acceso a los diferentes Pools y máquinas virtuales del departamento. Esta API fue desarrollada por el proyecto xapi y facilita la comunicación con máquinas virtuales que se ejecutan sobre el hipervisor XenServer.
- **Máquinas físicas**
Serán todas aquellas máquinas que pertenezcan al ecosistema de encendido y apagado. El usuario en la aplicación dará de alta información relevante de una máquina física la cual permita al sistema poder acceder a ella.
- **Pools**
Son los diferentes grupos de máquinas virtuales existentes en el departamento. Al igual que las máquinas físicas, el usuario también podrá dar de alta información relevante de un Pool, la cual le permita al sistema tener comunicación con el mismo, para posteriormente poder conocer y dar de alta a las máquinas virtuales que éste contiene.
- **Sais**
Los dispositivos SAI son los que proporcionan protección a problemas o cortes de luz en el departamento, previniendo daños en las máquinas conectadas a él. El usuario también dará de alta la información relevante para la comunicación del sistema con el SAI.

La Figura 12, presenta un diagrama general de la arquitectura del sistema, en el cual puede ver la distribución de todos los elementos anteriormente mencionados.

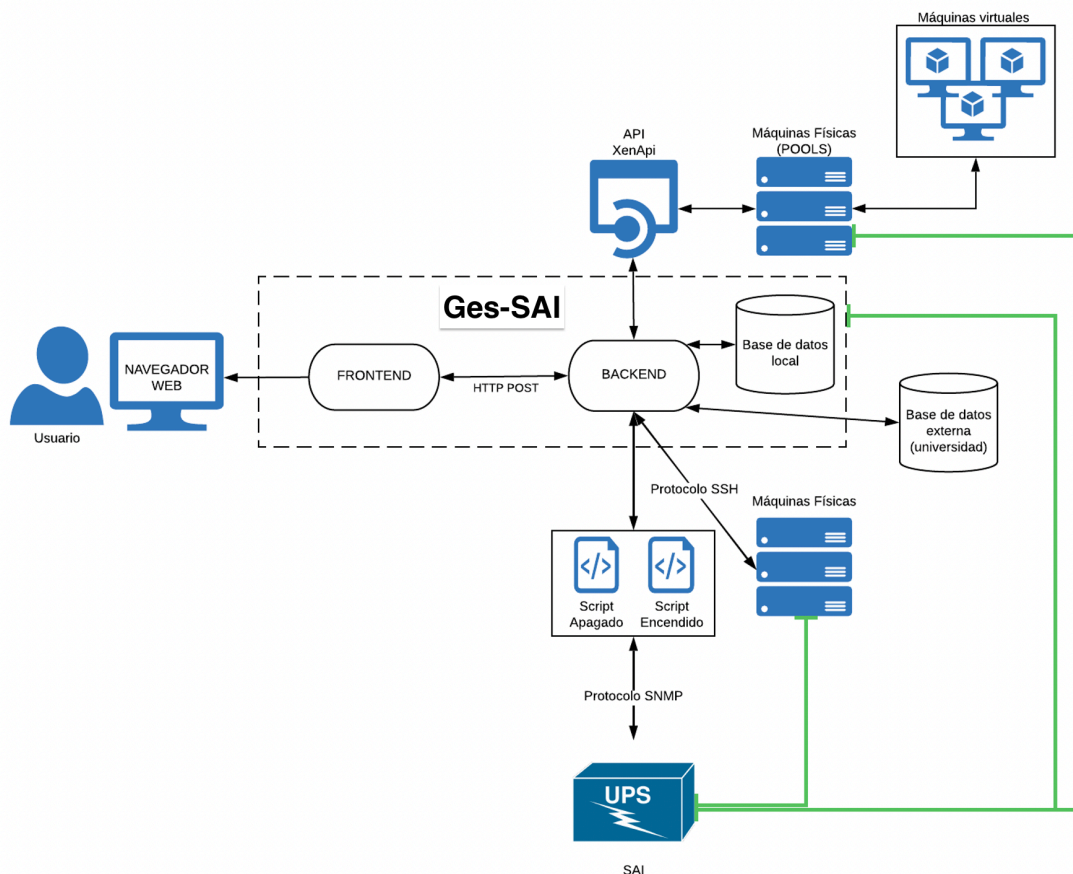


Figura 12 - Arquitectura del sistema

3.3. Metodología

Para el desarrollo del proyecto se ha elegido la metodología Scrumban, ya que esta nos permitirá una planificación flexible, una priorización adecuada de las tareas y un control de seguimiento continuo respecto a la evolución del proyecto.

Scrumban es una metodología para la gestión de proyectos la cual combina dos estrategias ágiles llamadas Scrum y Kanban. Inicialmente Scrumban fue desarrollado para dar soporte a los equipos de desarrollo al momento de hacer una transición de una metodología a la otra. Aunque Scrumban nació para ayudar a los equipos a migrar de una estrategia a otra, se descubrió que la combinación de ambas metodologías era bastante interesante y beneficiosa, sin embargo, esta metodología funciona mucho mejor si los miembros de los equipos ya conocen al menos una de las estrategias, ya sea scrum o Kanban (Laoyan, 2022).

La motivación a utilizar esta metodología se debe a la combinación de 2 experiencias obtenidas en los últimos tres años, la primera de ellas en mi anterior trabajo, en el cual me desempeñe como desarrollador junior, utilizando la metodología Scrum, y la segunda, en diferentes proyectos del máster en los cuales se utilizó la metodología Kanban.

Por lo que, con los dos técnicos del departamento se toma la decisión de utilizar la mezcla de estas dos metodologías, con el fin de poder adaptar la forma de trabajo con las condiciones del proyecto y así mejorar la calidad de este.

Con esta estrategia vamos a poder aprovechar los famosos Sprint de Scrum, los cuales permiten establecer el desarrollo por fases y priorizar las tareas, junto con esto, también se aprovechará el tablero Kanban, el cual facilitará la creación y visualización de las historias de usuario tomadas como requisitos del proyecto. El equipo de este proyecto está formado por cuatro

personas, dos técnicos del departamento, la tutora del TFM y por último el autor del TFM como estudiante del máster. A continuación, se mencionan los roles de cada uno de los miembros:

- **Primer técnico**
Desempeñará el papel de Product Owner, con el fin de validar que el desarrollo y los requisitos del proyecto sean los correctos.
- **Tutora y segundo técnico**
Desempeñarán el cargo de Stakeholders, con el fin de ser fuente de información para el equipo a la hora de tomar decisiones sobre el desarrollo del producto.
- **Autor del TFM**
Desempeñará el papel de desarrollador de todo el producto final.

3.4. Tecnologías Frontend

Durante este apartado se enumerarán las diferentes tecnologías, librerías o frameworks que se utilizarán durante el desarrollo del Frontend, las siguientes tecnologías fueron escogidas por su facilidad de aprendizaje e innovación.

- **React**

Con el fin de construir una aplicación que funcione únicamente como Frontend se ha decidido utilizar React o también llamado ReactJs, actualmente es una de las librerías más populares de **JavaScript**, la cual permite crear interfaces de usuario y hacer llamadas a APIs REST para obtener los datos en formato JSON. React ha sido creado por Facebook y una de sus ventajas es el fácil aprendizaje, a grandes rasgos la construcción de las interfaces se crea por medio de piezas aisladas de código reutilizables que se llaman componentes (Deyimar, 2022).

- **Material UI**

Material UI es una de las bibliotecas de componentes más populares actualmente para el desarrollo con React, ya que contiene una colección de componentes preconstruidos y listos para implementar en la aplicación que se esté realizando, junto con esto Material UI implementa Material Design de Google, dándole a los componentes un toque de diseño bastante bueno y permitiendo modificar sus opciones para tener una personalización acorde a lo que deseemos.

- **JavaScript**

JavaScript es un robusto lenguaje de programación que permite añadir características interactivas a un sitio web, lo que se conoce hoy en día como webs dinámicas, aparte de esto es el único lenguaje de programación que funciona en los navegadores de forma nativa, lo que quiere decir que puede ser interpretado sin necesidad de una compilación, este lenguaje fue inventado por Brendan Eich, cofundador del proyecto Mozilla.

3.5. Tecnologías Backend

Durante este apartado encontrara las tecnologías, librerías o framework que se escogieron para realizar el desarrollo del Backend, estas han sido seleccionadas teniendo en cuenta la facilidad de aprendizaje e innovación.

- **Django REST Framework**

Para el desarrollo del Backend se utilizará una herramienta de **Django** llamada Django REST Framework, el cual nos va a permitir crear una API REST utilizando lenguaje **Python**, las principales ventajas de DRF es la integración que tiene con la autenticación basada en OAuth1 y OAuth2, permite serializar los datos a partir de ORM y la amplia documentación que existe al ser una herramienta open source.

- **XenAPI**

XenAPI hace parte del proyecto Xen Project y es una API utilizada para la administración de máquinas virtuales, funciona como Interfax para controlar o configurar de forma remota máquinas virtuales que se estén ejecutando en una máquina física habilitada para el hipervisor Xen, básicamente esta API le permite al usuario que la use, crear, eliminar, apagar, encender, entre otras funcionalidades, todo esto para las máquinas virtuales.

- **PySNMP**

PySNMP es creado por Ilya Etingof, y funciona como una librería SNMP para Python totalmente gratuita y de código abierto, el código para su descarga está alojado en GitHub, y su principal función es facilitar la comunicación con dispositivos que cuenten con un agente de administración SNMP, para este proyecto esta es una de las herramientas más importantes ya que nos permitirá interrogar el estado de los dispositivos SAI.

- **Python**

Python es un lenguaje de programación fuertemente tipado, desarrollado bajo una licencia de código abierto por Guido Van Rossum, la principal ventaja por la cual se eligió Python para el desarrollo fue su fácil aprendizaje y la gran cantidad de documentación que existe gracias a su amplia comunidad.

3.6. Herramientas

En este apartado se enumeran las diferentes herramientas que se emplearan en el desarrollo del producto software, se escogieron estas herramientas debido a su gran utilidad a la hora de desarrollar, implementar o integrar una aplicación web.

- **Azure DevOps**

Azure DevOps es una tecnología desarrollada por Microsoft, la cual se creó con el fin de que los equipos de desarrollo y los de operaciones de TI, se colaboren entre sí para ofrecer a los clientes finales un producto confiable y de calidad, una de las principales ventajas de DevOps es que mejora el rendimiento del equipo. Se ha decidido utilizar esta herramienta, ya que nos permitirá mantener una documentación del proyecto, crear diferentes Sprints y gestionar su tablero de historias de usuario.

El desarrollo se dividirá en 3 Sprint, cada uno con una duración de tres semanas, el Sprint-01 iniciara el 1 de febrero y finalizará el 18 de febrero, el Sprint-02 iniciara el 21 de febrero y finalizará el 11 de marzo, por último, el Sprint-03 iniciara el 14 de marzo y finalizará el 1 de abril. Se utilizará el módulo Boards de Azure DevOps basado en el tablero Kanban, con el fin de mantener un orden de los diferentes requisitos, todos los ítems del Product Backlog se agruparán por 3 niveles, el nivel superior llamado “Épicas”, será encargado de contener el nivel de las “Funcionalidades”, y a su vez cada una de estas funcionalidades contendrá el tercer y último nivel llamado “Historias de usuario”.

- **PyCharm**

PyCharm es un IDE desarrollado por JetBrains, que les permite a los desarrolladores crear simples o complejos programas escritos en lenguaje Python, con la ventaja de que el desarrollador únicamente se centre en su código, algunas de las características principales con las que PyCharm cuenta son las siguientes:

- **Marcos de trabajo web Python**

Tiene una gran compatibilidad con marcos de trabajo utilizados para el desarrollo web como Django, Flask, Google App, por solo nombrar algunos. Aparte de esto también incluye depurador de plantillas para Django.

- **Editor de código inteligente**

PyCharm facilita el autocompletado de código para agilizar el desarrollo y ayudar a prevenir errores.

- **Herramientas para bases de datos**

Directamente desde PyCharm se puede acceder y ejecutar consultas a distintas bases de datos como Oracle, SQL Server, MySQL, entre otras.

- **MySQL Workbench**

Es una herramienta visual para el diseño de bases de datos, la cual le permite a los desarrolladores, modelar, gestionar y generar bases de datos de manera grafica. Su entorno grafico permite la modificación de todos los aspectos de la base de datos, facilitando la configuración de las tablas, como sus índices, columnas, opciones o permisos.

- **GitLab**

En el desarrollo siempre es importante poder administrar las diferentes versiones que se van generando del código, a esto se le conoce como repositorios Git, y una de las herramientas que permite hacer esto es GitLab, la cual facilita la total gestión de estos repositorios y también la integración con otras aplicaciones.

3.7. Backlog

El desarrollo se realizó en 3 Sprints diferentes, Sprint-01, Sprint-02 y Sprint-03. Respecto a los dos primeros Sprints (Sprint-01 y Sprint-02) estos han sido centrados únicamente para cumplir los requisitos del desarrollo del Backend, con lo cual también se detallará la documentación de la API y sus diferentes Endpoints empleados. En el caso del (Sprint-03) el desarrollo fue centrado únicamente para cumplir los requisitos del Frontend, con lo cual en este se presentarán los diferentes Wireframe de Ges-SAI, a continuación, se presenta el Backlog general de todo

el proyecto, en el que se puede ver las diferentes funcionalidades agrupadas en el Backend y Frontend e indicando el sprint al que pertenecieron.

GesSAI Team | + New Work Item | View as Board | Column Options

Order	Work Item Type	Title	State	Iteration Path
1	Epic	Backend	Resolved	GesSAI\Spring 01
	Feature	Consumir API Xenapi	Resolved	GesSAI\Spring 01
	Feature	CRUD Modelo Host	Resolved	GesSAI\Spring 01
	Feature	CRUD Modelo SAI	Resolved	GesSAI\Spring 01
	Feature	CRUD Modelo Dependencies	Resolved	GesSAI\Spring 01
	Feature	CRUD Modelo Pool	Resolved	GesSAI\Spring 02
	Feature	CRUD Modelo User	Resolved	GesSAI\Spring 02
	Feature	CRUD Modelo Virtual Machine	Resolved	GesSAI\Spring 02
	Feature	Funcionalidades auxiliares	Resolved	GesSAI\Spring 02
	Feature	Login Api	Resolved	GesSAI\Spring 02
	Feature	Conexión con protocolo SNMP	Resolved	GesSAI\Spring 02
2	Epic	Frontend	Resolved	GesSAI\Spring 03
	Feature	Interfaz Modulo Máquinas	Resolved	GesSAI\Spring 03
	Feature	Interfaz Modulo Pools	Resolved	GesSAI\Spring 03
	Feature	Interfaz Modulo Sais	Resolved	GesSAI\Spring 03
	Feature	Interfaz Modulo Usuarios	Resolved	GesSAI\Spring 03
	Feature	Login	Resolved	GesSAI\Spring 03
	Feature	Interfaz Modulo Dependencias	Resolved	GesSAI\Spring 03
	Feature	Interfaz Modulo Baterías	Resolved	GesSAI\Spring 03
	Feature	Interfaz Modulo Instrucciones	Resolved	GesSAI\Spring 03

Figura 13 - Backlog general del proyecto

CAPÍTULO 4:

Sprint-01

Este capítulo es el primero de tres capítulos en dar inicio a la presentación de todo el desarrollo hecho en cada uno de los Sprints, en este se encontrarán tres puntos fundamentales a presentar, el primero de ellos serán los antecedentes, esto con el fin de tener una síntesis conceptual de todo el desarrollo realizado en el Sprint-01, como segundo punto podrá ver el Sprint Backlog como un resumen de las tareas realizadas, y por último en el tercer punto, se detallarán las funcionalidades y sus historias de usuario.

4.1. Antecedentes

Sin duda alguna uno de los retos más grandes y el cual dio iniciativa a desarrollar este proyecto, es el preguntarnos ¿Cómo nos podemos comunicar con un dispositivo SAI? A partir de aquí, el encontrar información en internet sobre esto no es una tarea sencilla, sin embargo, nos dimos cuenta de que en el mercado existen a la venta “Tarjetas de comunicación LAN para SAI/UPS”. Estas tarjetas disponen de un conector RJ45 para realizar conexiones LAN y son compatibles con los siguientes protocolos de comunicación: TCP/IP, UDP, SNMP, SMTP, HTTP, HTTPS, SSL, SSH, IPV4, IPV6, y DHCP. Con esta tarjeta es posible realizar un seguimiento del tiempo de actividad y el nivel de tráfico, entre otras cosas. Por suerte, en el departamento del DSIC 2 de los 3 dispositivos SAI cuentan con estas tarjetas integradas en su sistema, se hará uso del protocolo SNMP con el fin de poder interrogar el estado de estos dispositivos.

4.1.1. Protocolo SNMP

SNMP son las siglas de “Protocolo simple de administración de redes” o en inglés “Simple Network Management Protocol”. Tal como lo indica la organización (ManageEngine, 2022) este protocolo se encuentra en el séptimo nivel del modelo OSI ubicado en la capa de aplicación, sirve para intercambiar información entre dispositivos de red, en la actualidad la mayoría de los elementos de red que son de nivel profesional cuentan con un agente SNMP incluido, el agente es un programa que esta empaquetado dentro del elemento de red, sin embargo, estos agentes deben estar habilitados y configurados para comunicarse con el sistema de administración de red. Su utilidad proviene del hecho que permite recuperar la información de los dispositivos que se encuentran conectados a la red, todo esto de una forma estandarizada. Se puede decir que SNMP cuenta con cuatro componentes básicos que permiten su correcto funcionamiento, estos se especifican a continuación:

➤ **Administrador SNMP**

El sistema administrador es una entidad separada capaz de comunicarse con los diferentes dispositivos que se encuentran conectados a la red, en este caso la aplicación Ges-SAI será la encargada de realizar la comunicación con los dispositivos SAI.

➤ **Dispositivos administrados**

Esto hace referencia a los dispositivos que se encuentran en la red, los cuales requieren algún tipo de monitorización y administración, pueden ser servidores, switches, routers, impresoras, etc. En este caso será el dispositivo SAI.

➤ **Agente SNMP**

Como se mencionó anteriormente el agente es un programa empaquetado dentro del dispositivo conectado a la red, este debe estar habilitado con el fin de poder recopilar información ubicada en la base de datos de administración del dispositivo localmente, la respuesta es enviada al administrador SNMP cuando este la solicita, en este caso retornada a la aplicación Ges-SAI. La Figura 14 presenta la distribución de los componentes que interfieren en el agente SNMP.

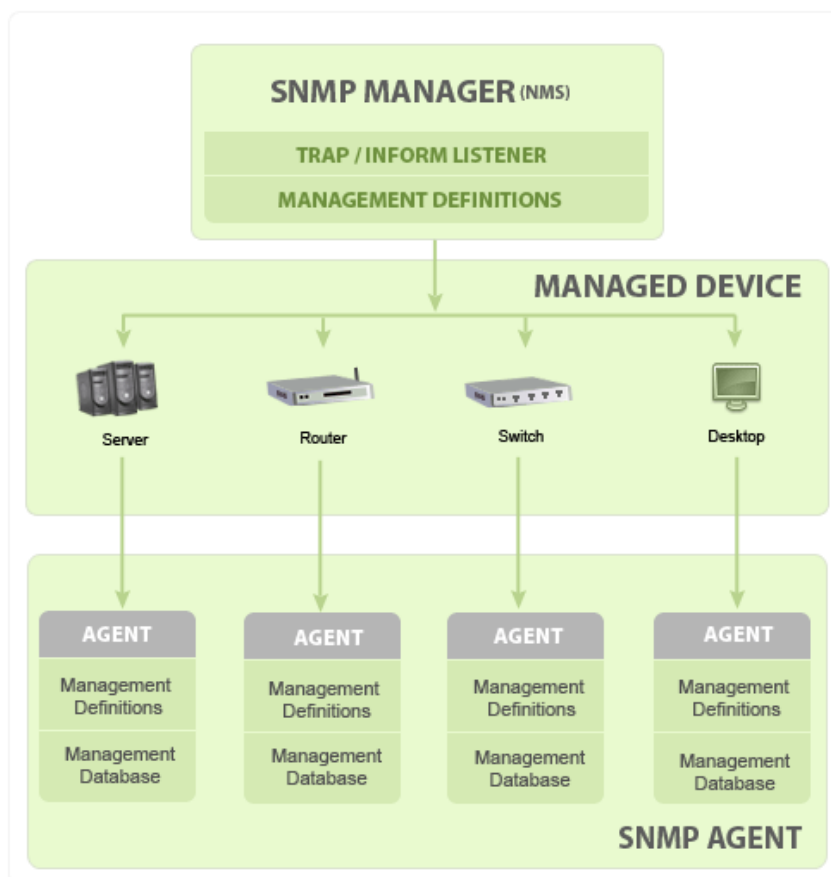


Figura 14 - Diagrama básico de comunicación SNMP (ManageEngine, 2022)

➤ **Base de información de administración (MIB)**

El agente SNMP del dispositivo contiene una base de datos de información en la que se describen los parámetros del dispositivo administrado, esta base de datos se denomina Base de información de administración (MIB). Estos archivos son un conjunto estándar de valores estadísticos que están definidos para los nodos del hardware de una red, se puede decir que los archivos MIB son un conjunto de preguntas que el administrador SNMP conoce y puede hacerle a un agente.

Los MIB contienen OID de una forma correctamente definida, OID significa “Identificador de objetos” y son estos los que identifican a un objeto dentro de un dispositivo, por lo general cada dispositivo tiene objetos típicos que se desean

administrar, por ejemplo, un switch tiene objetos de interés a administrar, como lo pueden ser, el tráfico entrante y saliente o la tasa de pérdida de paquetes, para nuestro caso un dispositivo SAI cuenta con diferentes objetos para supervisar como lo pueden ser, estado de la batería, duración restante de la batería, temperatura de la batería, voltaje de la batería, etc. Estos objetos OID se representan con un árbol jerárquico con diferentes niveles iniciando desde la raíz hasta cada una de las hojas, se puede decir que cada OID es una dirección que sigue los niveles del árbol. La Figura 15 presenta una estructura de ejemplo de un OID:

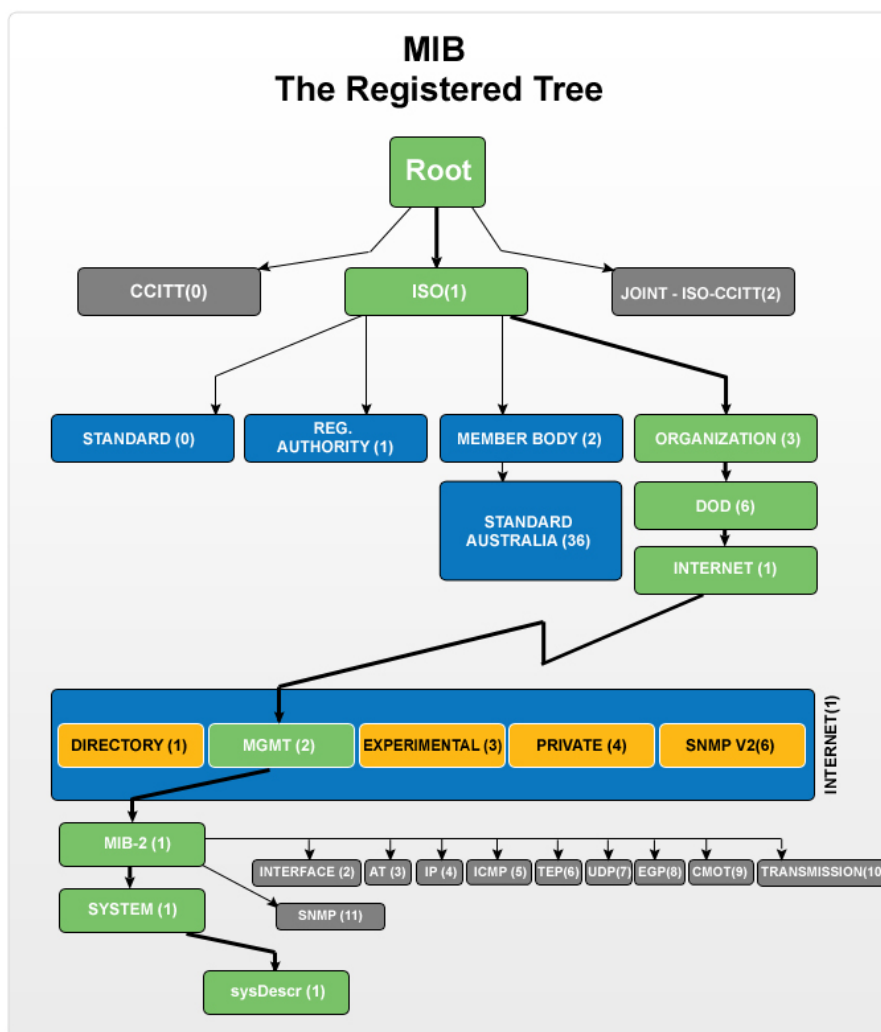


Figura 15 - Diagrama de árbol de MIB (ManageEngine, 2022)

Si tomamos el anterior ejemplo presentado en la página de ManageEngine, podríamos decir que la dirección OID del objeto sería la siguiente: (1.3.6.1.2.1.1.1).

Una vez obtenida esta información y haber entendido para que sirve un agente SNMP, lo primero que se tuvo que hacer fue buscar un listado de los diferentes OID utilizados para la administración de dispositivos UPS/SAI, afortunadamente encontramos la información recopilada por (Hartman, 2019), en la cual nos presenta un listado de OID con su respectiva descripción. A continuación, se presenta un fragmento de esta lista la cual incluye el OID que nos interesa para consultar la duración restante de la batería de un SAI.

OID	DESCRIPCIÓN
SNMPv2-SMI::mib-2.33.1.2.1.0	Estado de la batería del SAI, 2 = Normal
SNMPv2-SMI::mib-2.33.1.2.2.0	Segundos de batería
SNMPv2-SMI::mib-2.33.1.2.3.0	Duración restante de la batería en minutos
SNMPv2-SMI::mib-2.33.1.2.4.0	Porcentaje restante de carga de la batería
SNMPv2-SMI::mib-2.33.1.2.5.0	Voltaje de la batería (falta el decimal después del primer dígito)

Tabla 2 - Lista de OID relevantes (Harman, 2019)

4.1.2. Interrogar dispositivos SAI

Teniendo en cuenta la información presentada en el apartado anterior, se concateno el código MIB general del protocolo SNMP (1.3.6.1.2.1), con el código OID específico para obtener la duración restante de la batería en minutos (33.1.2.3.0), con esto obtenemos el OID final (1.3.6.1.2.1.33.1.2.3.0), a partir de esto la tarea ya fue comunicarnos con los dispositivos SAI utilizando la biblioteca para Python llamada PySNMP, esta librería cuenta con un motor SNMP completamente funcional el cual es capaz de actuar en roles de agente, administrador o proxy, su versión estable actualmente es la 4.4.

Como se mencionó anteriormente, la arquitectura del sistema cuenta con 2 Scripts que se encargan de interrogar al SAI cada cierto tiempo, dependiendo de esto, el sistema iniciara con el encendido o apagado de máquinas, solo si el tiempo restante de la batería es mayor o menor respectivamente a los parámetros establecidos en base de datos, el objeto SAI cuenta con atributos llamados “**value_off**” y “**value_on**”, estos valores contienen el tiempo mínimo de apagado en minutos y el tiempo máximo de encendido en minutos respectivamente, ejemplo:

- **Value_off = 20**, esto significa que, si al SAI le quedan 20 minutos de batería, el sistema empezara a apagar todas las máquinas.
- **Value_on = 55**, esto significa que, si el SAI cuenta con 55 minutos de batería, el sistema empezara a encender todas las máquinas.

A continuación, se presenta el código utilizado para obtener el tiempo restante en minutos de las baterías de un dispositivo SAI.

Parámetros:

- **User:** Usuario de autenticación para el SAI
- **AuthKey:** Clave de autenticación
- **PriveKey:** Clave privada de autenticación
- **Host:** Ip o nombre del SAI
- **Oid:** Id del objeto a consultar. (Tiempo restante de baterías)

```
from pysnmp.hlapi import *

def batteryLevel(user, authKey_sai, priveKey_sai, host, oid):
    level = ''
    iterator = getCmd(
        SnmpEngine(),
        UsmUserData(user, authKey=authKey_sai, privKey=priveKey_sai),
        UdpTransportTarget((host, 161)),
        ContextData(),
        # ObjectType(ObjectIdentity('SNMPv2-MIB', 'sysDescr', 0))
        ObjectType(ObjectIdentity(oid))
    )

    errorIndication, errorStatus, errorIndex, varBinds = next(iterator)

    if errorIndication:
        print(errorIndication)

    elif errorStatus:
        print('%s at %s' % (errorStatus.prettyPrint(), errorIndex and varBinds[int(errorIndex) - 1][0] or '?'))

    else:
        for varBind in varBinds:
            code = str(varBind[0].prettyPrint())
            level = int(varBind[1])
            print('Tiempo restante de batería: ' + str(level) + ' Minutos')
            print('Codigo de MIB: ' + str(code))

    return level
```

Figura 16 – Función para interrogar dispositivo SAI

4.1.1. Pools

Las máquinas virtuales hacen parte fundamental de la infraestructura del DSIC, y son agrupadas por Pools, los cuales están basados en el hipervisor XenServer. “Un hipervisor, conocido también como monitor de máquina virtual (VMM), es un software que crea y ejecuta máquinas virtuales (VM) y que, además, aísla su sistema operativo y recursos de las máquinas virtuales y permite crearlas y gestionarlas” (Red Hat, 2020). Se hace llamar Pool a la agrupación de máquinas físicas, una de estas máquinas es la que se designa como host máster (Máquina maestro) la cual proporciona un único punto de contacto para todas las demás máquinas físicas del pool.

El host máster es el encargado de enrutar toda la comunicación a otras máquinas del grupo según sea necesario, son en estas en las que se ejecutan las diferentes máquinas virtuales. Cuando el host máster se apaga, el resto de las máquinas del pool no estarán disponibles hasta que este se encuentre en línea nuevamente. La Figura 17, presenta una diagrama de alto nivel como ejemplo de la distribución de un Pool de máquinas físicas.

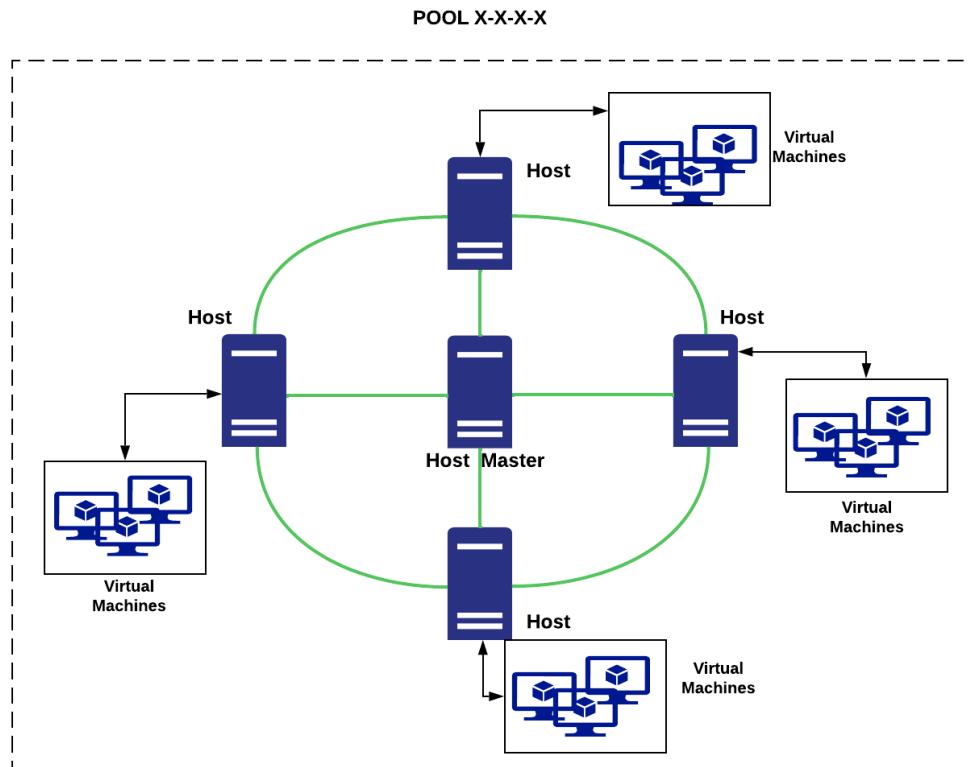


Figura 17 - Diagrama ejemplo de alto nivel de un Pool

4.1.2. Apagar y encender máquinas virtuales

Los sistemas de virtualización suelen contar con una API de administración la cual proveen una interfaz para controlar y configurar de forma remota las máquinas virtualizadas que se ejecutan en un host habilitado. Como se mencionó anteriormente, el sistema de virtualización que utiliza el departamento del DSIC se llama XCP-ng, el cual cuenta con su propia API llamada XenAPI y toda su documentación puede encontrarse en el siguiente enlace <https://xapi-project.github.io/xen-api/>.

Por medio del conjunto de llamadas a procedimientos remotos de esta Api, la aplicación Ges-SAI podrá apagar o encender las máquinas virtuales del departamento, esta comunicación se hace directamente al **Host Máster**, el cual contiene información de todos los **Host esclavos** que pertenecen al Pool. La Figura 18 muestra un ejemplo del envío de la instrucción de encendido o apagado desde Ges-SAI a XenAPI, una vez que el host recibe la instrucción, este

consulta en su diccionario de máquinas virtuales a cuál va dirigida y procede a ejecutarla.

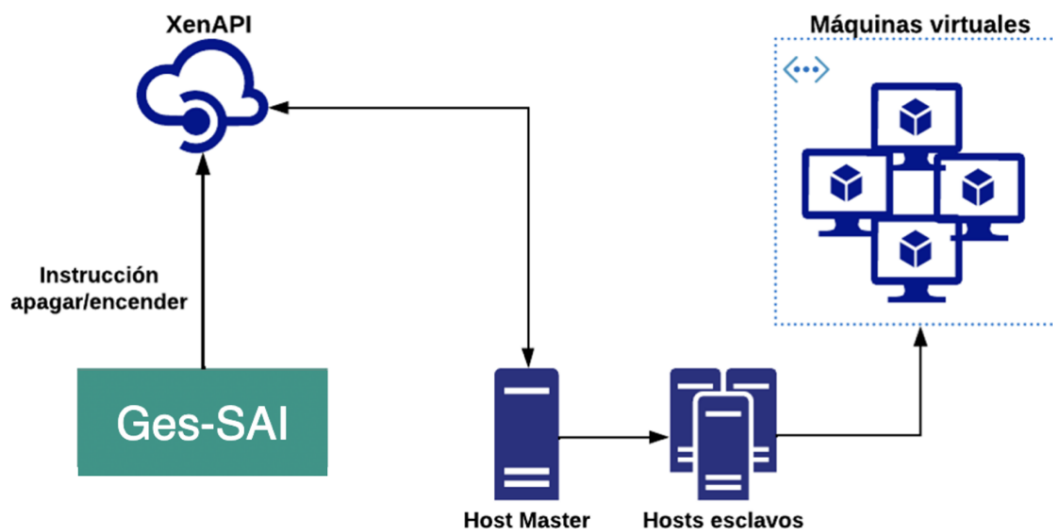


Figura 18 – Diagrama de comunicación con XenAPI

Por medio de la librería XenAPI de Python, se puede hacer uso de la API facilitando la comunicación con los diferentes Pools del departamento. A continuación, se menciona un ejemplo del bloque de código usado para la conexión con un Pool, apagado y encendido de una máquina virtual.

- **Conectarse a un Pool**

```
import XenAPI
session = XenAPI.Session('http://x.x.x.x/')
session.login_with_password('user', 'password')
```

- **Apagar máquina virtual**

```
vm = session.xenapi.VM.get_by_uuid(uuid) #Id virtual machine
session.xenapi.VM.hard_shutdown(vm) #-> Apagar
```

- **Encender máquina virtual**

```
vm = session.xenapi.VM.get_by_uuid(uuid) #Id virtual machine
session.xenapi.VM.start(vm, False, True) #-> Encender
```

- **Listar máquinas virtuales de un Pool**

```
records = session.xenapi.VM.get_all_records_where(
    'field "is_a_template" = "false" and '
    'field "is_a_snapshot" = "false" and '
    'field "is_control_domain" = "false"')
for ref in records:
    vm = records[ref]
    virtual_machines.append(vm)
```


4.1.3. Apagar y encender máquinas físicas

Con la ventaja de que las máquinas se pueden alcanzar por la red podemos apagarlas o encenderlas de forma remota, y para esto ocupamos otros medios diferentes a una API como se hace en las máquinas virtuales.

➤ Apagar Máquina física

Para apagar una máquina física basta con conocer el nombre o dirección IP del equipo y ejecutar un comando SSH desde una terminal del sistema. Sin embargo, existe una librería de Python llamada **Paramiko**, la cual desde el código podemos crear conexiones SSH y ejecutar comandos, este módulo permite la comunicación tanto del lado del servidor como del cliente.

Al momento de querer conectarse a una máquina, es normal que esta nos solicite un usuario y una contraseña para acceder, pero para este caso utilizaremos la clave pública del servidor donde se ejecuta (Ges-SAI), con esto facilitamos la autenticación de la aplicación con las máquinas, en el servidor de Ges-SAI se generarán un par de **claves publica/privada**, estos archivos se crean con el nombre de:

- **id_rsa**: Clave privada, la cual permanecerá en el servidor Ges-SAI.
- **id_rsa.pub**: Clave pública, cada usuario que desee incluir su máquina en el proceso de apagado automático, deberá copiarla en el ordenador. Este archivo estará disponible desde la interfaz de Ges-SAI para su descarga.

A continuación, se menciona un ejemplo del bloque de código usado para la conexión y apagado por SSH de una máquina física.

• Crear sesión de servidor SSH

```
import paramiko
cli = paramiko.SSHClient()
```

• Cargar clave privada del servidor

```
prikey = paramiko.RSAKey.from_private_key_file('/id_rsa')
cli.set_missing_host_key_policy(paramiko.AutoAddPolicy())
```

• Conectarse a la máquina

```
cli.connect(hostname='ip',
            port=22,
            username='user',
            pkey=prikey,
            timeout=2)
```

• Ejecutar comando de apagado (shutdown)

```
stdin, stdout, stderr = cli.exec_command('shutdown now')
```

➤ **Encender Máquina física**

Cuando una máquina se encuentra apagada esta no cuenta con una dirección IP, con lo cual no se podrá acceder a ella por medio del protocolo SSH, para este caso se utilizará una tecnología llamada Wake on LAN (WoL), la cual permite encender un equipo de forma remota por medio de la red. De manera que si una máquina física se encuentra apagada, suspendida o hibernada se puede utilizar este estándar, siempre y cuando la tarjeta de red de la máquina sea compatible con él y se encuentre configurada correctamente en la BIOS.

WoL lo que hará será enviar un paquete mágico utilizando el protocolo UDP a una dirección MAC que le indiquemos, al momento que la tarjeta de red recibe el paquete mágico, esta comprueba si la MAC pertenece a ella, y si es así, la tarjeta envía una señal eléctrica a la placa base para que se encienda la máquina.

Afortunadamente para Python existe un pequeño modulo llamado **wakeonlan** el cual permite enviar de manera sencilla un paquete mágico a una dirección MAC.

A continuación, se menciona un ejemplo del bloque de código usado para arrancar una máquina física.

- **Importar modulo wakeonlan**

```
from wakeonlan import send_magic_packet
```
- **Enviar paquete mágico (MAC)**

```
send_magic_packet('ff.ff.ff.ff.ff.ff')
```

La Figura 19 presenta un diagrama de ejemplo, del envío del paquete mágico por la red.

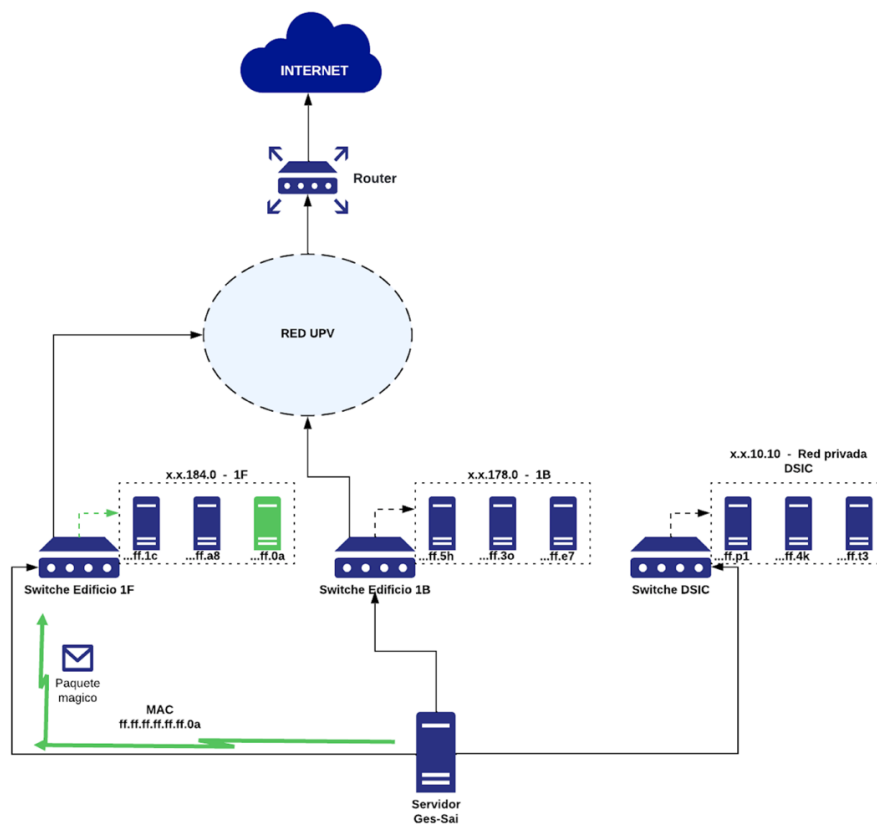


Figura 19 - Diagrama envío de paquete mágico (WoL)

4.2. Backlog de historias de usuario

La Figura 20, presenta el Sprint Backlog documentado con ayuda de la herramienta Azure DevOps en el cual se puede ver el rango de fechas en el que se realizó el Sprint, la cantidad de días trabajados y cada una de las historias de usuario agrupadas por funcionalidad.

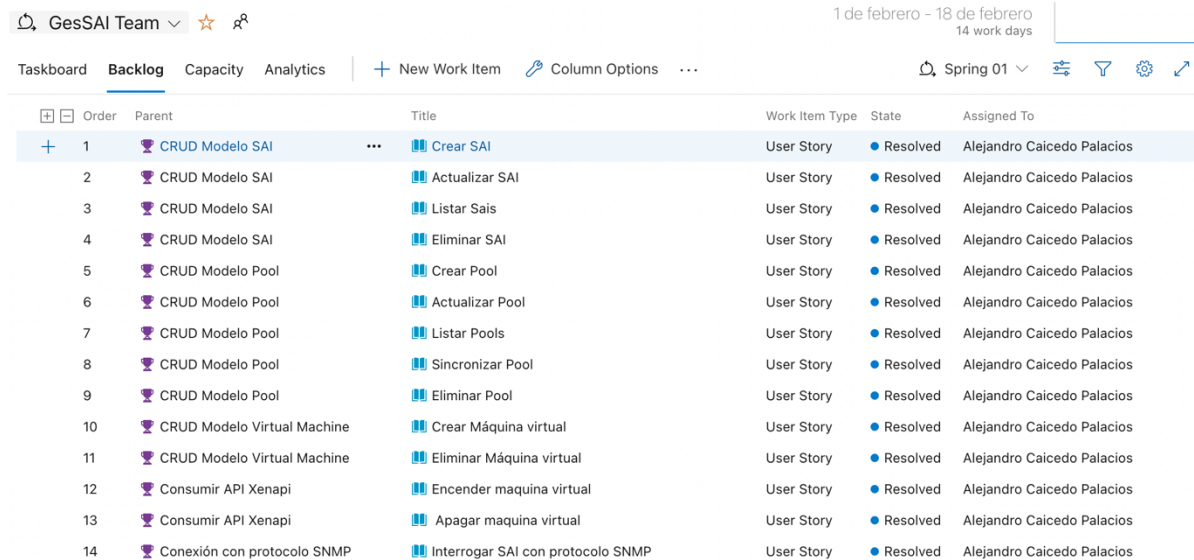


Figura 20 - Sprint Backlog de historias de usuario, Sprint-01

Como se puede apreciar en la imagen anterior, la columna Parent contiene el nombre de la funcionalidad y a su derecha en la columna Title tenemos el nombre de la historia de usuario, cada una de estas HU ya se encuentran en estado resueltas, todas ellas fueron realizadas en el periodo comprendido entre el 1 de febrero y 18 de febrero. A continuación, se agrupan las historias de usuario por funcionalidad y se detallan cada una de ellas.

4.3. Funcionalidades

Con el fin de presentar de manera ordenada las historias de usuario realizadas en el Sprint-01, estas serán agrupadas en las diferentes funcionalidades de la aplicación, si se requiere también estarán acompañadas de la documentación del Endpoint utilizado en la API.

4.3.1. CRUD Modelo SAI

Las máquinas físicas, los Pools y las máquinas virtuales, todas deben tener asociado un SAI, por lo que el modelo SAI fue uno de los principales en realizarse, ya que de este parte prácticamente todos los demás. A continuación, se describen todas las historias de usuario y los Endpoints empleados para su desarrollo.

Historia de usuario	Crear SAI
ID	HU01

Como

Usuario técnico

Quiero

Dar de alta un SAI, solo buscándolo por su nombre DNS

Para

Posteriormente poder asociar las diferentes máquinas físicas o virtuales conectadas a él.

Criterios de aceptación

- El SAI debe tener conexión con el servidor de la aplicación

Tabla 3 - HU01 Crear SAI

create Interact

POST /gessaiapi/v1/sais/

Clase que contiene los metodos genericos para la vista de los Sai ['get', 'post', 'put', 'delete']

Request Body

The request body should be a "application/json" encoded object, containing the following items.

Parameter	Description
name_sai required	
userConnection required	
authKey required	
privKey required	
ip	
mac required	
url	
protocol required	
state required	
code_oid required	
value_off required	
value_on required	
administrator	

Figura 21 - Endpoint Crear SAI

Historia de usuario	Actualizar SAI
ID	HU02
Como	Usuario técnico
Quiero	Editar la información de un SAI
Para	Corregir posibles cambios en el futuro.

Criterios de aceptación

- El SAI debe tener conexión con el servidor de la aplicación.
- Solo se podrá editar los campos – Tiempo de encendido y apagado.

Tabla 4 - HU02 Actualizar SAI

update Interact

PUT `/gessaiapi/v1/sais/{id}/`

Clase que contiene los metodos genericos para la vista de los Sai ['get', 'post', 'put', 'delete']

Path Parameters

The following parameters should be included in the URL path.

Parameter	Description
<code>id</code> required	

Request Body

The request body should be a `"application/json"` encoded object, containing the following items.

Parameter	Description
<code>name_sai</code> required	
<code>userConnection</code> required	
<code>authKey</code> required	
<code>privKey</code> required	
<code>ip</code>	
<code>mac</code> required	
<code>url</code>	
<code>protocol</code> required	
<code>state</code> required	
<code>code_oid</code> required	
<code>value_off</code> required	
<code>value_on</code> required	
<code>administrator</code>	

Figura 22 - Endpoint Actualizar SAI

Historia de usuario	Listar SAIS
ID	HU03
Como	Usuario técnico
Quiero	Visualizar en una lista todos los SAIS que he dado de alta
Para	Llevar un control de la cantidad de SAIS y el detalle de cada uno.

Criterios de aceptación

- El usuario mínimo debe tener un SAI dado de alta.

Tabla 5 - HU03 Listar SAIS

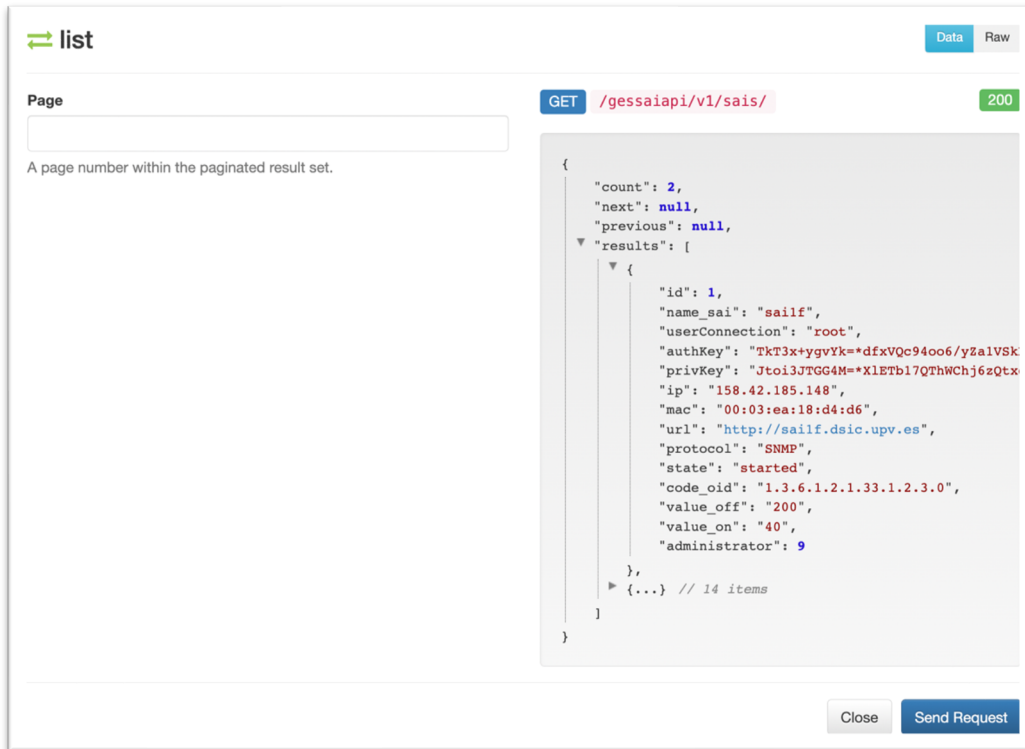


Figura 23 - Endpoint Listar SAIS

Historia de usuario	Eliminar SAI
ID	HU04
Como	Usuario técnico
Quiero	Eliminar un SAI
Para	Corregir cambios y eliminar todas las dependencias creadas y asociadas a él.
Criterios de aceptación	

Tabla 6 - HU04 Eliminar SAI

delete

↔ Interact

DELETE `/gessaiapi/v1/sais/{id}/`

Clase que contiene los metodos genericos para la vista de los Sai ['get', 'post', 'put', 'delete']

Path Parameters

The following parameters should be included in the URL path.

Parameter	Description
<code>id</code> required	

Figura 24 - Endpoint Eliminar SAI

4.3.2. CRUD Modelo Pool

Para poder tener acceso a la información de las máquinas virtuales, primero se debe tener conexión al Pool que las contiene, por lo que este modelo es fundamental para el flujo del sistema, una vez la aplicación cuenta con los datos de conexión de un Pool, el usuario puede dar de alta sus máquinas virtuales. A continuación, se describen todas las historias de usuario y los Endpoints empleados para su desarrollo.

Historia de usuario	Crear Pool
ID	HU05
Como	Usuario técnico o genérico
Quiero	Dar de alta un Pool
Para	Consultar y posteriormente dar de alta a las máquinas virtuales que tiene creadas.
Criterios de aceptación	
<ul style="list-style-type: none"> ○ El Pool debe tener conexión con el servidor de la aplicación. ○ Antes debe haber mínimo un SAI dado de alta para asociarlo al Pool. 	

Tabla 7 - HU05 Crear Pool

create ↔ Interact

POST `/gessaiaapi/v1/pools/`

Clase que contiene los metodos genericos para la vista de los Pools ['get', 'post', 'put', 'delete']

Request Body


The request body should be a `"application/json"` encoded object, containing the following items.

Parameter	Description
<code>name_pool</code>	
<code>ip</code>	
<code>url</code>	
<code>username</code>	
<code>password</code>	
<code>type</code> required	
<code>user</code> required	
<code>sais</code> required	

Figura 25 - Endpoint Crear Pool

Historia de usuario	Actualizar Pool
ID	HU06
Como	Usuario técnico o genérico
Quiero	Editar la información de un Pool
Para	Corregir posibles cambios en el futuro.
Criterios de aceptación	
<ul style="list-style-type: none"> ○ El Pool debe tener conexión con el servidor de la aplicación. ○ Solo se podrá editar los campos – Nombre de Pool y Sai asociado. 	

Tabla 8 - HU06 Actualizar Pool

update 

PUT `/gessaiapi/v1/pools/{id}/`
 Clase que contiene los metodos genericos para la vista de los Pools ['get', 'post', 'put', 'delete']

Path Parameters
 The following parameters should be included in the URL path.

Parameter	Description
<code>id</code> required	

Request Body
 The request body should be a `"application/json"` encoded object, containing the following items.

Parameter	Description
<code>name_pool</code>	
<code>ip</code>	
<code>url</code>	
<code>username</code>	
<code>password</code>	
<code>type</code> required	
<code>user</code> required	
<code>sais</code> required	

Figura 26 - Endpoint Actualizar Pool

Historia de usuario	Listar Pools
ID	HU07
Como	Usuario técnico o genérico
Quiero	Visualizar en una lista todos los Pools que he dado de alta
Para	Llevar un control de la cantidad de Pools y el detalle de cada uno.
Criterios de aceptación	
<ul style="list-style-type: none"> ○ El usuario mínimo debe tener un Pool dado de alta. 	

Tabla 9 - HU07 Listar Pools

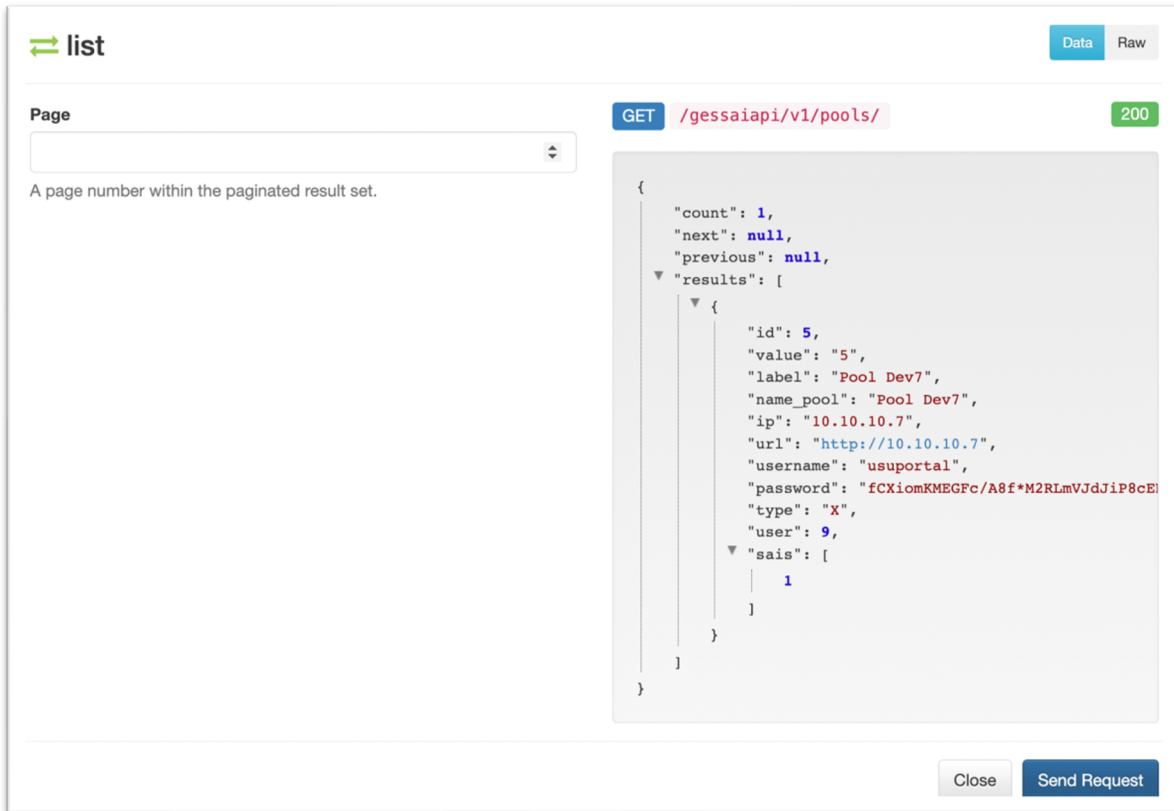


Figura 27 - Endpoint Listar Pools

Historia de usuario	Sincronizar Pool
ID	HU08
Como	Usuario técnico o genérico
Quiero	Sincronizar el Pool
Para	Actualizar automáticamente los cambios realizados en los Host.
Criterios de aceptación	
<ul style="list-style-type: none"> ○ El Pool debe tener conexión con el servidor de la aplicación. 	

Tabla 10 - HU08 Sincronizar Pool



Figura 28 - Endpoint Sincronizar Pool

Historia de usuario	Eliminar Pool
ID	HU09
Como	Usuario técnico o genérico
Quiero	Eliminar un Pool
Para	Corregir cambios y eliminar todas las dependencias y máquinas virtuales asociadas a él.
Criterios de aceptación	

Tabla 11 - HU09 Eliminar Pool

delete Interact

DELETE `/gessaiapi/v1/pools/{id}/`

Clase que contiene los metodos genericos para la vista de los Pools ['get', 'post', 'put', 'delete']

Path Parameters

The following parameters should be included in the URL path.

Parameter	Description
<code>id</code> required	

Figura 29 - Endpoint Eliminar Pool

4.3.3. CRUD Modelo Virtual Machine

A continuación, se presentan los requisitos del modelo de máquinas virtuales, para realizar estas funcionalidades, fue indispensable primero realizar las funcionalidades del modelo Pool, ya que un Pool es quien contiene toda la información de las máquinas virtuales, así el usuario solo tiene que dar de alta el Pool y consultar que máquina virtual desea generarle una dependencia de encendido y apagado.

Historia de usuario	Crear máquina virtual
ID	HU10
Como	Usuario técnico o genérico
Quiero	Dar de alta a una máquina virtual, consultándola en la lista de máquinas virtuales de un Pool.
Para	Posteriormente crearle una dependencia con otra máquina, ya sea virtual o física.
Criterios de aceptación	
<ul style="list-style-type: none"> ○ El Pool que continente la máquina virtual debe estar dado de alta ○ La conexión con el Pool debe ser correcta. 	

Tabla 12 - HU10 Crear máquina virtual

create ↔ Interact

POST `/gessaiapi/v1/virtualmachine/`

Clase que contiene los metodos genericos para la vista de las Virtual Machine ['get', 'post', 'put', 'delete']

Request Body

The request body should be a `"application/json"` encoded object, containing the following items.

Parameter	Description
<code>uuid</code>	
<code>ref</code>	
<code>name</code>	
<code>type</code> required	
<code>ip</code>	
<code>mac</code>	
<code>power_state</code>	
<code>so</code> required	
<code>description</code>	
<code>order</code> required	
<code>pool</code> required	

Figura 30 - Endpoint Crear máquina virtual

Historia de usuario	Eliminar máquina virtual
ID	HU11
Como	Usuario técnico o genérico
Quiero	Dar de baja a una máquina virtual.
Para	Corregir cambios y eliminar toda dependencia asociada a ella.
Criterios de aceptación	
<ul style="list-style-type: none"> ○ El Pool que continente la máquina virtual debe estar dado de alta ○ La conexión con el Pool debe ser correcta. 	

Tabla 13 - HU11 Eliminar máquina virtual

delete Interact

DELETE `/gessaiapi/v1/virtualmachine/{id}/`

Clase que contiene los metodos genericos para la vista de las Virtual Machine ['get', 'post', 'put', 'delete']

Path Parameters

The following parameters should be included in the URL path.

Parameter	Description
<code>id</code> required	

Figura 31 - Endpoint Eliminar máquina virtual

4.3.4. Consumir API XenAPI

XenAPI es la Api que nos provee una conexión directa para el encendido o apagado de las máquinas virtuales, a continuación, se describen las historias de usuario ligadas a las tareas de esta API. En el apartado Apagar y encender máquinas virtuales se ha mencionado el código utilizado para su solución.

Historia de usuario	Encender máquina virtual
ID	HU12
Como	Usuario técnico o genérico
Quiero	Que las máquinas virtuales dadas de alta en el sistema sean encendidas automáticamente.
Para	Continuar con el flujo de trabajo correctamente.
Criterios de aceptación	
<ul style="list-style-type: none"> ○ El Pool debe estar dado de alta ○ La conexión con el Pool debe ser correcta 	

Tabla 14 - HU12 Encender máquina virtual

Historia de usuario	Apagar máquina virtual
ID	HU13
Como	Usuario técnico o genérico
Quiero	Que las máquinas virtuales dadas de alta en el sistema sean apagadas automáticamente, antes de que la corriente sea interrumpida
Para	Prevenir daños o fallos en las mismas.
Criterios de aceptación	
<ul style="list-style-type: none"> ○ El Pool debe estar dado de alta ○ La conexión con el Pool debe ser correcta 	

Tabla 15 - HU13 Apagar máquina virtual

4.3.5. Conexión con protocolo SNMP

Una de las funcionalidades principales de Ges-SAI es el poder interrogar a los diferentes dispositivos SAI dados de alta en el sistema, esto con el fin de validar el estado de las baterías y prevenir que estas se agoten antes de que se apaguen todas las máquinas virtuales o máquinas físicas del departamento. A continuación, se describe su historia de usuario y el endpoint empleado para su desarrollo.

Historia de usuario	Interrogar al SAI con protocolo SNMP
ID	HU14
Como	Usuario técnico o genérico
Quiero	Que la aplicación pueda interrogar a los SAI que he dado de alta.
Para	Iniciar el proceso de Apagado o Encendido de máquinas, solo si salta la alerta de batería baja.
Criterios de aceptación	
<ul style="list-style-type: none"> ○ El SAI debe estar dado de alta ○ La conexión con el SAI debe ser correcta 	

Tabla 16 - HU14 Interrogar al SAI con protocolo SNMP



Figura 32 - Endpoint Interrogar al SAI con protocolo SNMP

CAPÍTULO 5:

Sprint-02

Este capítulo es el segundo de tres capítulos, el cual contiene todo el desarrollo realizado en el Sprint-02, como se indicó en el capítulo anterior, la realización de este Sprint y el Sprint-01 fueron únicamente centrados en el desarrollo del Backend, a continuación, encontrara tres puntos fundamentales a presentar, el primero de ellos serán los antecedentes, esto con el fin de tener una síntesis conceptual de todo el desarrollo realizado en este Sprint, como segundo punto podrá ver el Sprint Backlog como un resumen de las tareas realizadas, y por último en el tercer punto, se detallarán las funcionalidades y sus historias de usuario.

5.1. Antecedentes

El objetivo principal del proyecto es lograr un correcto apagado o encendido de las máquinas virtuales y máquinas físicas del departamento. Para cumplir con esto se debe cubrir diferentes factores, uno de ellos es tener en cuenta las máquinas que comparten recursos con otras, las cuales se han denominado **máquinas con dependencia**, otro de los factores para tener en cuenta son las máquinas que no dependen de ninguna otra para apagarse o encenderse, a estas se denominan **máquinas sin dependencia**.

A continuación, se expone la visión general de la dependencia y la no de dependencia entre máquinas ya sean virtuales o físicas, junto con esto se presenta el correcto apagado y/o encendido de las máquinas y el proceso de multihilos como solución al tiempo límite de descarga de baterías de los SAIS.

5.1.1. Máquinas con dependencia

Como ya sabemos, el proceso de encendido y apagado de máquinas no puede hacerse a la ligera sin previamente tener un orden establecido, dicho esto se opta por dar una solución eficiente a las problemáticas mencionadas anteriormente en los apartados: 2.2.1, 2.2.2 y 2.2.4 abarcando el término **DEPENDENCIAS**.

Por esto el usuario de la aplicación podrá dar de alta a una máquina e incluirla en un grupo de dependencias, con el fin de garantizar que las máquinas se apaguen o inicien en el orden correcto.

Nos referimos con este término a la solución propuesta, ya que una máquina puede ser “dependiente” de los recursos de otra máquina, existen 4 casos en los que se pueda dar esta dependencia, ya sea entre una máquina virtual con otra máquina virtual, una máquina física con otra máquina física o una máquina virtual con una máquina física y viceversa una física con una virtual.

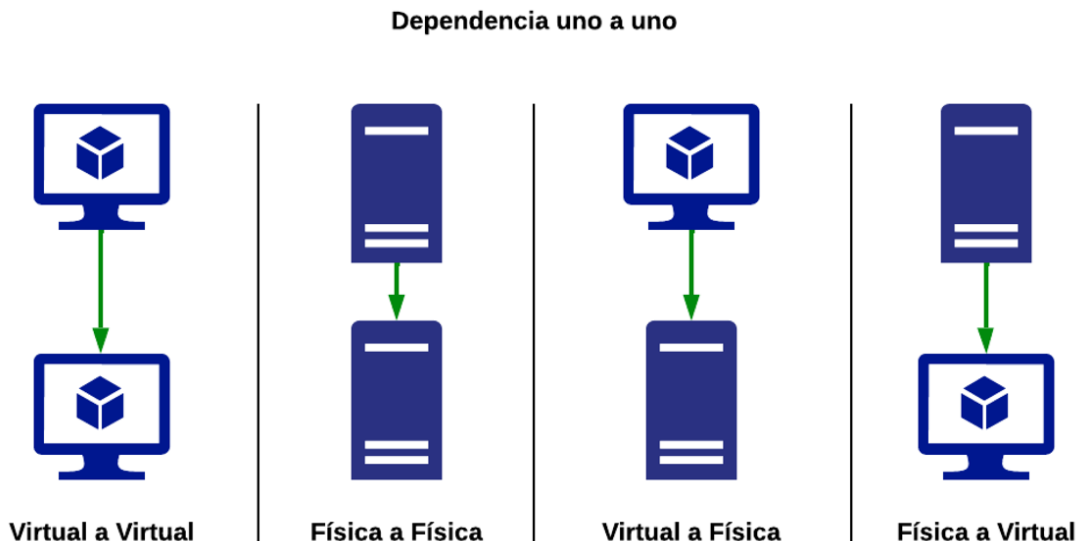


Figura 33 - Dependencia uno a uno

Aparte de la dependencia vista anteriormente, la cual es una dependencia de uno a uno (máquina a máquina) también existe el caso donde una máquina depende de más de una máquina, también llamada dependencia de uno a muchos (máquina a máquinas). Para esto también pueden existir 4 casos diferentes, donde la máquina depende únicamente de máquinas virtuales, o depende únicamente de máquinas físicas, y también donde depende de máquinas físicas y virtuales al mismo tiempo.

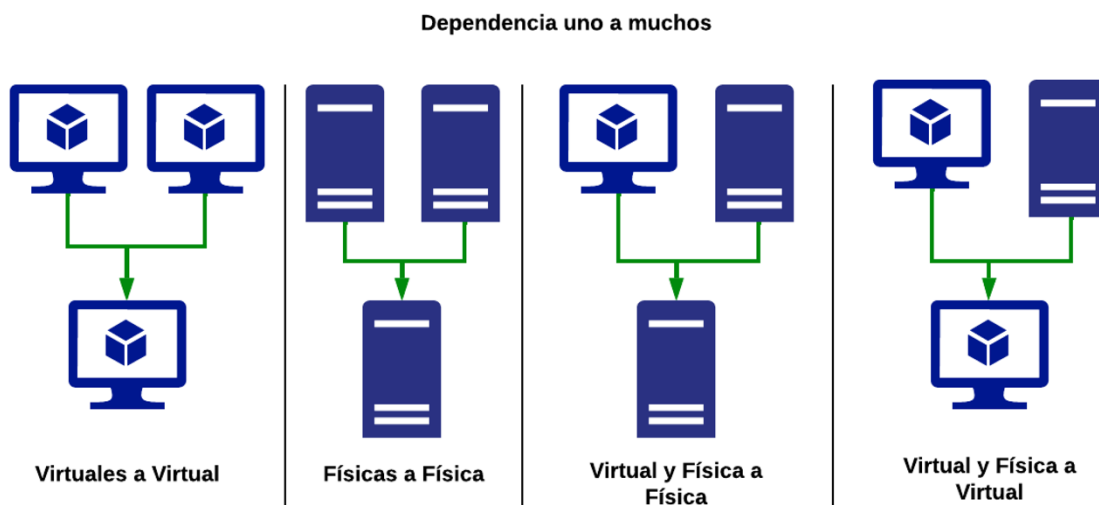


Figura 34 - Dependencia uno a muchos

ÁRBOL DE DEPENDENCIAS

A medida que se va creando la dependencia entre máquinas, se va formando un árbol de dependencias, este árbol puede ser tan grande como se requiera. Recordemos que el departamento DSIC cuenta con 2 edificios y con 3 dispositivos SAI, los cuales están conectados por red, esto quiere decir, que pueden existir máquinas en un edificio que dependan de máquinas ubicadas en el otro edificio.

Sin embargo, un corte de luz no necesariamente puede ocurrir en ambos edificios, cada edificio cuenta con su administración de SAI, por lo que, si el apagón ocurre en el edificio 1F, Ges-SAI dará la instrucción de apagar todas las máquinas del SAI_1F y únicamente las máquinas del edificio 1B que se encuentran en una dependencia con las máquinas del edificio 1F. Esto ha sido explicado más detalladamente en el apartado Problemática de infraestructura dividida. Tomando el mismo ejemplo del apartado mencionado, la Figura 35 representa un ejemplo grafico de árbol de dependencia entre máquinas de ambos edificios.

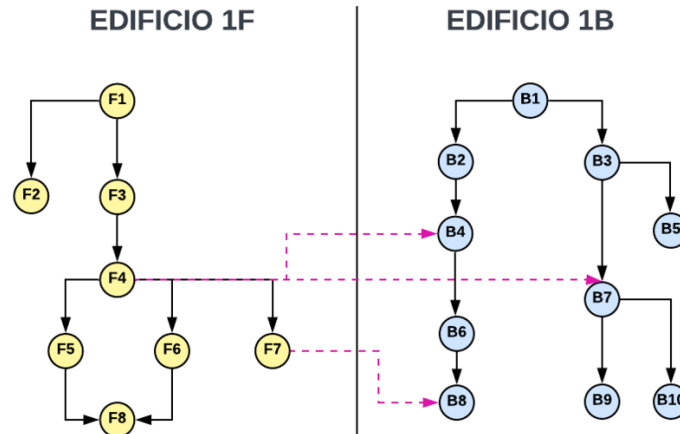


Figura 35 - Árboles de dependencia y dependencia entre sedes

Como vemos las máquinas B4 y B7 son dependientes de la máquina F4 y la máquina B8 es dependiente de la máquina F7. Conociendo esto, la Figura 36 muestra como quedaría el árbol de dependencias del edificio 1F.

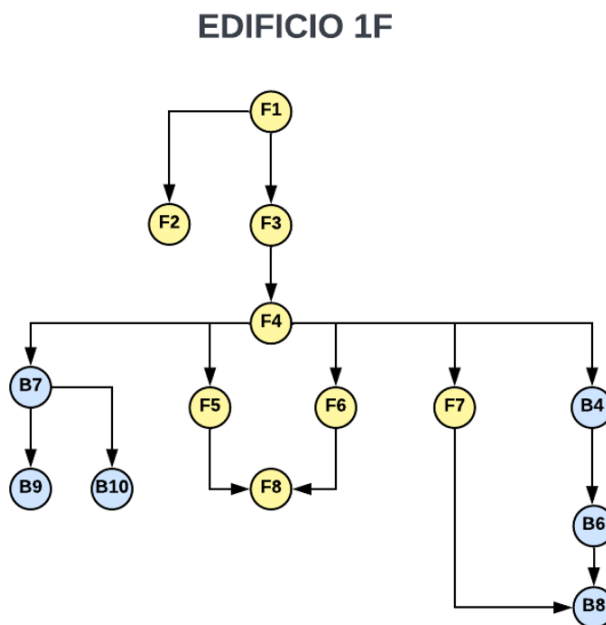


Figura 36 - Árbol de dependencias

La dependencia recae en el padre y sus hijos, por esto vemos que, aunque la dependencia directa sea a las máquinas B7, B4 y B8, si estas a su vez tienen hijos dependientes, esos hijos también harán parte del árbol de dependencia.

5.1.2. Máquinas sin dependencia

Este apartado hace referencia a esas máquinas físicas que no tienen dependencia alguna con otra máquina, para estas no será necesario tener un orden de espera, ya que la instrucción de encendido o apagado será lanzada de forma secuencial para cada una de ellas.

El usuario podrá dar de alta estas máquinas en Ges-SAI simplemente conociendo el nombre DNS y a que SAI está conectada, una vez hecho esto la máquina hará parte de la administración de encendido y apagado de la aplicación. Sin embargo, este caso únicamente funcionara para las **máquinas físicas**, ya que las máquinas virtuales siempre van a tener dependencia porque pertenecen a un Pool de máquinas.

5.1.3. Proceso de Multihilos

El tiempo límite con el que cuentan las baterías de los SAI, también es uno de los factores importantes para tener en cuenta, esto ya se ha mencionado en la problemática del apartado 2.2.3. Cada máquina tarda unos segundos en apagarse, y como sabemos estos segundos pueden aumentar si se encuentran con alguna actualización, la sumatoria de estos tiempos es bastante considerable si hablamos de apagar un poco más de 100 máquinas, por lo que, para dar solución a esta problemática, se ha utilizado la programación en hilos.

HILOS

Cuando se habla en programación de hilo o en inglés Thread, no es más que una tarea que se puede ejecutar al mismo tiempo que otra, sin embargo, los hilos no pueden ejecutarse por sí solos ni vivir fuera de un proceso, y un proceso es quien permite la ejecución de estos, así que se puede decir que un proceso es el supervisor de los hilos en ejecución (Reyes, 2018). Un proceso puede ejecutar uno o más hilos a la vez, siendo cada uno de estos tareas independientes donde pueden o no compartir los mismos datos.

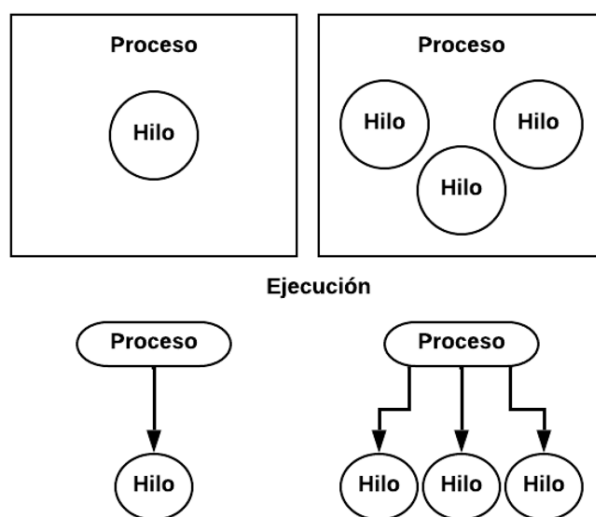


Figura 37 - Hilos y procesos

Principales ventajas de usar Hilos:

- Agiliza los tiempos de retraso entre la comunicación cliente - servidor
- Se puede tener más de un camino de ejecución en un mismo programa
- La creación de un nuevo hilo es más rápida que la creación de un nuevo proceso

5.1.4. Proceso de apagado de Máquinas en hilos

Como se muestra en la Figura 8, el apagado de las máquinas dependientes no puede ser secuencial (una tras otra), esto retrasaría el proceso y hará que se agoten primero las baterías del SAI, por lo que se utilizarán hilos de ejecución y para esto es necesario entender lo siguiente. Una máquina que presta recursos a otra, se llamara máquina **Padre**, y una máquina que recibe recursos de otra, se llamara máquina **Hijo**, sin embargo, también puede existir una máquina que recibe y presta recursos a la vez, a estas se les llamara máquina **Padre-Hijo**.

Entendido esto, a continuación, se describe una serie de pasos que se realizan en el flujo de apagar una máquina.

1. Listar todas las máquinas que únicamente son máquinas (Hijo).
2. Ejecutar un hilo por cada una de estas máquinas.
3. Apagar máquina.
4. Una vez este apagada, se notifica a todos sus padres.
5. Ejecutar un hilo por cada padre notificado.
6. El padre pregunta si todos sus hijos están apagados.
 - SI = El padre se apaga, repitiendo el flujo desde el punto 3.
 - NO = El padre no se apaga y espera que los demás hijos terminen de apagarse y lo notifiquen.

Existirá un flujo único que se muestra a continuación, este flujo servirá para todas las máquinas físicas o virtuales que pertenezcan a un árbol de dependencia.

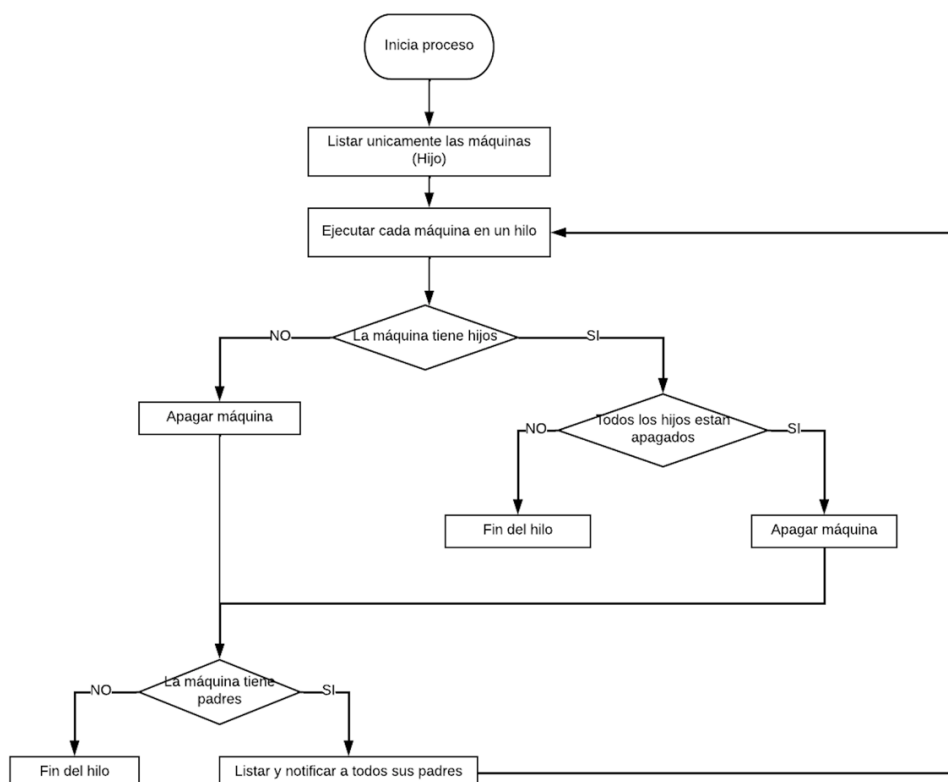


Figura 38 - Diagrama de flujo, Apagar máquina

Como se puede ver, el diagrama representa el flujo del proceso de apagado de máquinas, a continuación, se presenta una simulación con un árbol de dependencia usando los pasos del diagrama de flujo anterior.

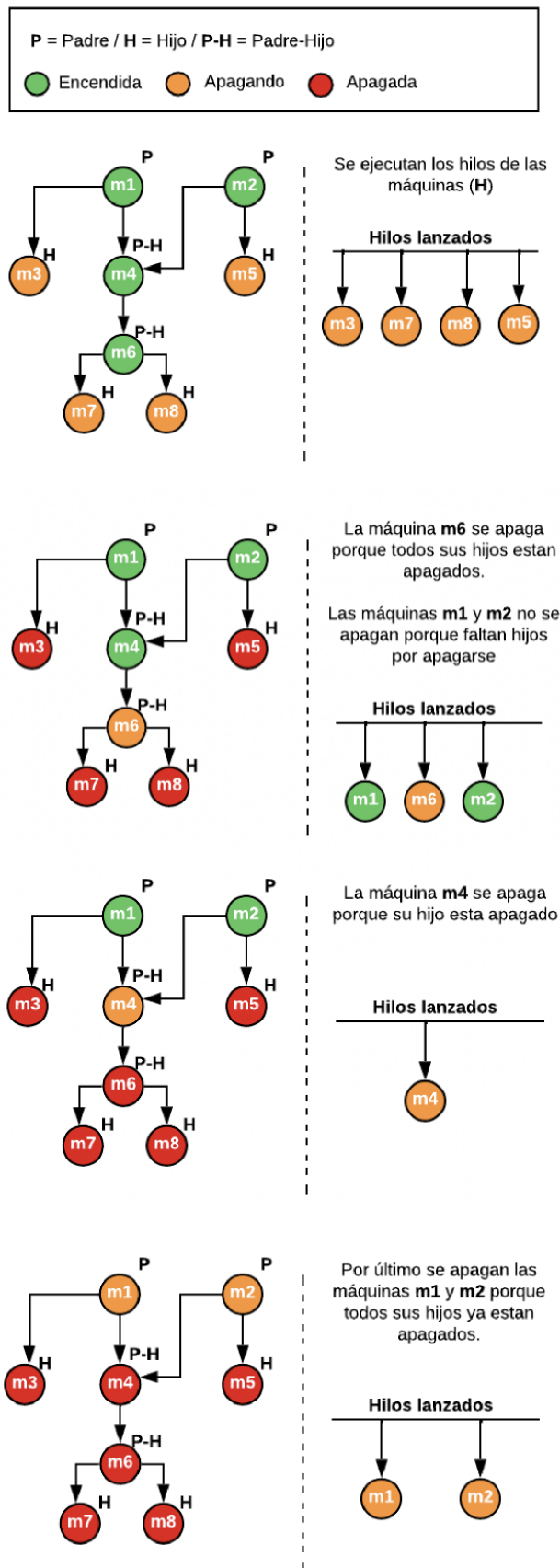


Figura 39 - Árbol de dependencia, flujo de apagado de máquinas

5.1.5. Proceso de encendido de Máquinas en hilos

Para el encendido de máquinas ya no es necesario tener en cuenta el tiempo límite de agotamiento de las baterías, ya que este no es un caso que lo requiera, sin embargo, este proceso se hará de la misma manera que se hace para el apagado de máquinas, pero de forma inversa. Como ya sabemos, una máquina **Hijo** depende de una máquina **Padre**, entonces en este caso se dará prioridad de encendido a los padres antes de sus hijos, a continuación, se describe la serie de pasos que se realizarán en el flujo de encender una máquina.

1. Listar todas las máquinas que únicamente son máquinas (Padre).
2. Ejecutar un hilo por cada una de estas máquinas.
3. Encender máquina.
4. Una vez este encendida, se notifica a todos sus hijos.
5. Ejecutar un hilo por cada hijo notificado.
6. El hijo pregunta si todos sus padres están encendidos.
 - SI = El hijo se enciende, repitiendo el flujo desde el punto 3.
 - NO = El hijo no se enciende y espera que los demás padres terminen de encenderse y lo notifiquen.

A continuación, se presenta el flujo de encendido que servirá para todas las máquinas físicas o virtuales pertenecientes a un árbol de dependencia.

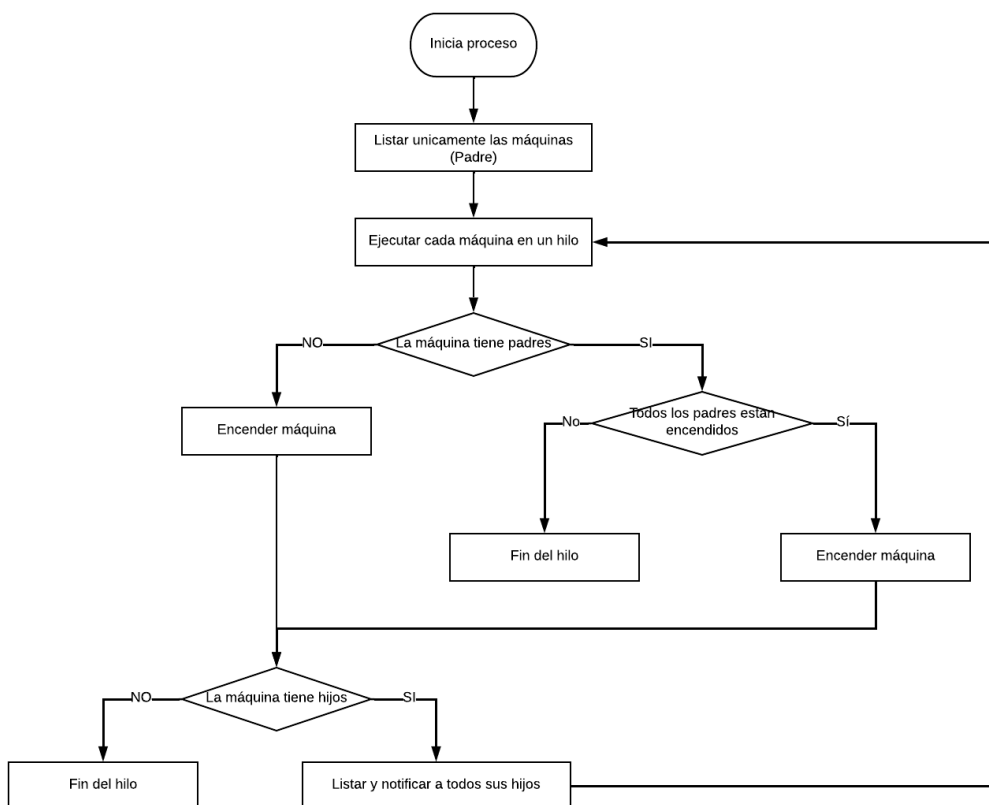


Figura 40 - Diagrama de flujo, Encender máquina

Como se pudo ver, el anterior diagrama representa el flujo del proceso de encendido de máquinas, a continuación, se presenta una simulación con un árbol de dependencia usando el diagrama de flujo anterior.

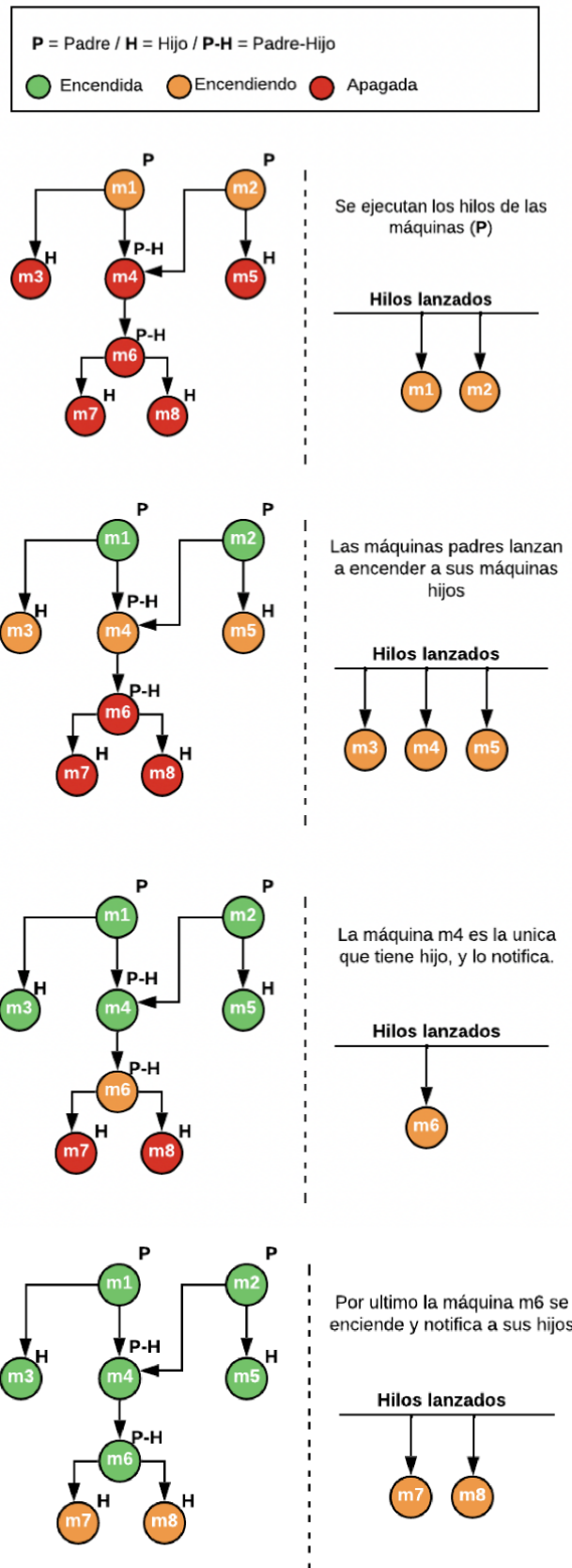


Figura 41 - Árbol de dependencia, flujo de encendido de máquinas

5.2. Backlog de historias de usuario

Esta es la segunda y última parte del desarrollo del Backend, la Figura 42, presenta un resumen del Sprint Backlog realizado durante el proyecto con la ayuda de la herramienta Azure DevOps, en el que se puede ver las historias de usuario agrupadas por funcionalidades las cuales serán detalladas más adelante.

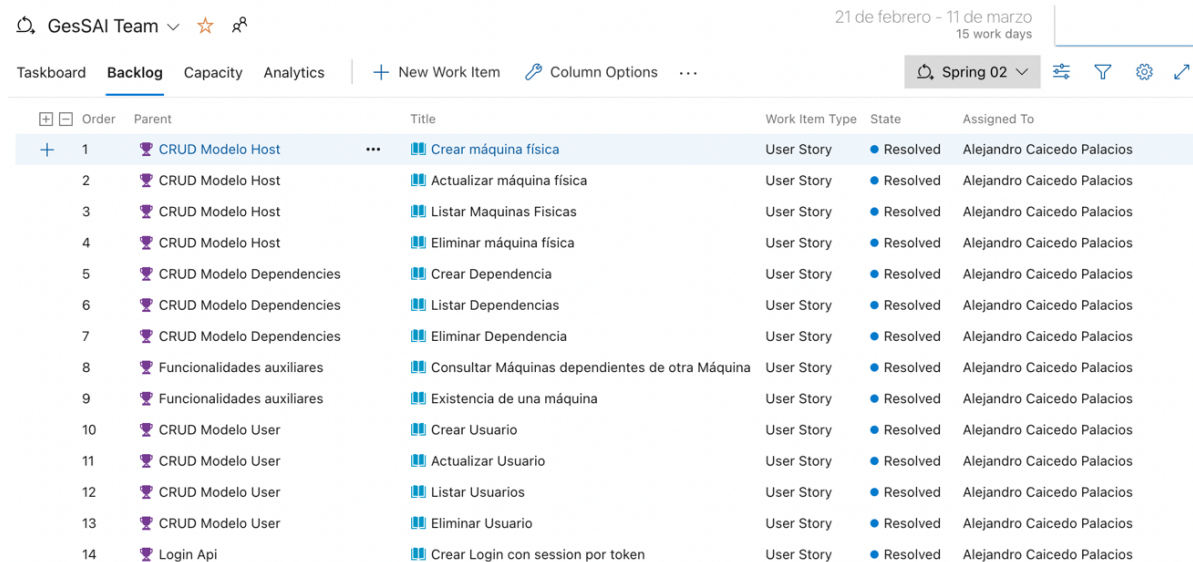


Figura 42 - Sprint Backlog de historias de usuario, Sprint-02

Este Sprint fue realizado en el periodo comprendido entre el 21 de febrero y 11 de marzo, enfocado únicamente al desarrollo del Backend y en el podemos apreciar las funcionalidades en la columna Parent y las historias de usuario en la columna Title, todas y cada una de ellas se encuentran en estado resueltas, en el siguiente apartado, se agrupan las historias de usuario por funcionalidad y se detallan cada una de ellas.

5.3. Funcionalidades

A continuación, se describen todas y cada una de las historias de usuario agrupándose por funcionalidad, si se requiere algunas de ellas estarán acompañadas por la documentación de la Api indicando el Endpoint empleado.

5.3.1. CRUD Modelo Host

El modelo Host se encarga de contener la información de toda máquina registrada en el sistema, y a partir de él, se crean las diferentes dependencias entre máquinas virtuales y físicas. A continuación, se describen los requisitos necesarios para la implementación de toda la funcionalidad del modelo Host.

Historia de usuario	Crear máquina física
ID	HU15
Como	Usuario técnico o genérico
Quiero	dar de alta una máquina física, solo buscándola por el nombre de DNS

Para

posteriormente añadirla en el sistema de dependencias de encendido y apagado.

Criterios de aceptación

- El usuario anteriormente tiene que dar de alta mínimo un SAI.

Tabla 17 - HU15 Crear máquina física

create ↔ Interact

POST /gessaiaapi/v1/hosts/

Clase contenedora de los metodos genericos para la vista de los Hosts [Crear, Actualizar, Eliminar, Listar]

Request Body

The request body should be a "application/json" encoded object, containing the following items.

Parameter	Description
name_host	
ip	
mac	
so	
groupId required	
order	
description	
poolId required	
user required	
type_host	
host_host required	

Figura 43 - Endpoint Crear máquina física

Historia de usuario	Actualizar máquina física
ID	HU16
Como	Usuario técnico o genérico
Quiero	editar la información de una máquina física
Para	corregir posibles cambios en el futuro.
Criterios de aceptación <ul style="list-style-type: none"> ○ La máquina deberá estar asociada a un SAI. ○ El usuario solo podrá editar las máquinas que ha dado de alta. 	

Tabla 18 - HU16 Actualizar máquina física

update ⇒ Interact

PUT `/gessaiapi/v1/hosts/{id}/`

Clase contenedora de los metodos genericos para la vista de los Hosts [Crear, Actualizar, Eliminar, _istar]

Path Parameters

The following parameters should be included in the URL path.

Parameter	Description
<code>id</code> required	

Request Body

The request body should be a `"application/json"` encoded object, containing the following items.

Parameter	Description
<code>name_host</code>	
<code>ip</code>	
<code>mac</code>	
<code>so</code>	
<code>groupId</code> required	
<code>order</code>	
<code>description</code>	
<code>poolId</code> required	
<code>user</code> required	
<code>type_host</code>	
<code>host_host</code> required	

Figura 44 - Endpoint Actualizar máquina física

Historia de usuario	Listar máquinas físicas
ID	HU17
Como	Usuario técnico o genérico
Quiero	Listar todas las maquinas físicas que he dado de alta en la aplicación
Para	Tener un control sobre ellas, eligiendo alguna acción o ver el resumen de su información.
Criterios de aceptación	
<ul style="list-style-type: none"> ○ El usuario debe tener mínimo una máquina física dada de alta. ○ El usuario solo podrá ver sus máquinas. 	

Tabla 19 - HU17 Listar máquinas físicas

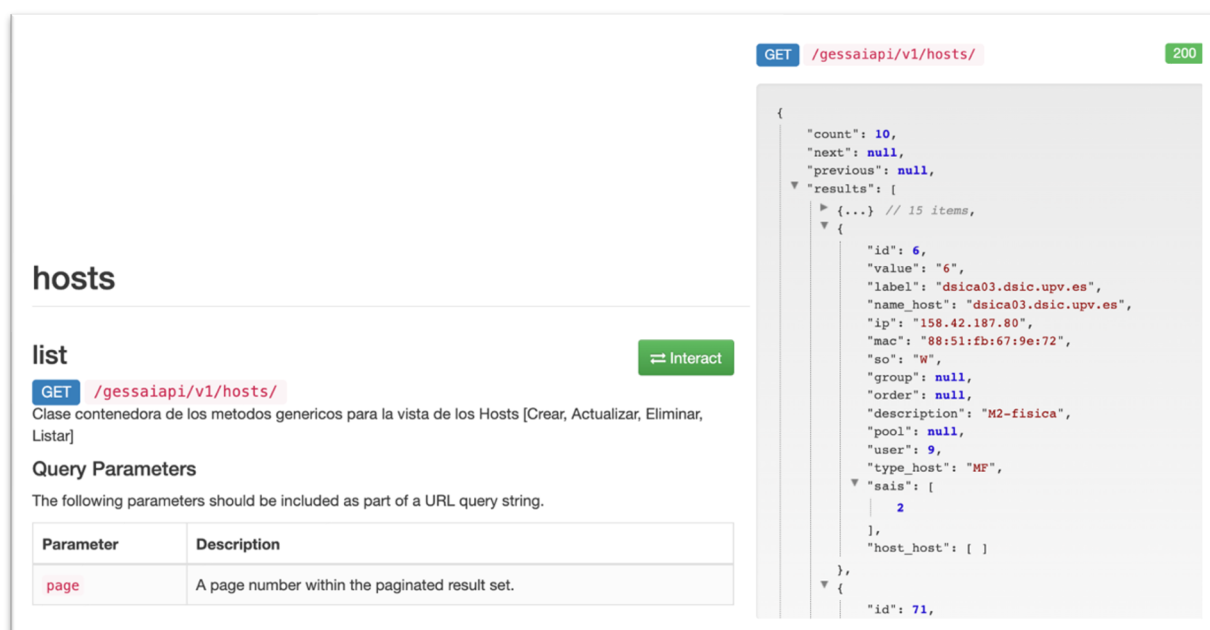


Figura 45 - Endpoint Listar máquinas físicas

Historia de usuario	Eliminar máquina física
ID	HU18
Como	Usuario técnico o genérico
Quiero	poder dar de baja una máquina física
Para	Desvincularla del sistema de apagado y encendido y eliminar toda dependencia.
Criterios de aceptación	
<ul style="list-style-type: none"> ○ El usuario solo podrá eliminar sus máquinas. 	

Tabla 20 - HU18 Eliminar máquina física

delete Interact

DELETE /gessaiapi/v1/hosts/{id}/

Clase contenedora de los metodos genericos para la vista de los Hosts [Crear, Actualizar, Eliminar, Listar]

Path Parameters

The following parameters should be included in the URL path.

Parameter	Description
id required	

Figura 46 - Endpoint Eliminar máquina física

5.3.2. CRUD Modelo Dependencias

Si no existieran dependencias entre máquinas, al momento de realizar el apagado de las mismas todo sería un desorden y muchas de las máquinas se quedarían bloqueadas esperando respuesta de otras máquinas las cuales comparten recursos, es por esto que las dependencias es uno de los requisitos principales para que la aplicación funcione correctamente, su explicación se detalla a profundidad en el apartado 5.1.1, a continuación, se describen todas las historias de usuario y sus endpoint empleados.

Historia de usuario	Crear dependencia
ID	HU19
Como	Usuario técnico o genérico
Quiero	Crear una dependencia entre dos máquinas virtuales o máquinas físicas
Para	poder dar un orden específico de apagado y encendido.
Criterios de aceptación	
<ul style="list-style-type: none"> El usuario deberá haber dado de alta más de una máquina física o un pool. 	

Tabla 21 - HU19 Crear dependencia

createDependence

create Interact

POST /gessaiapi/v1/createDependence/

Metodo que crea una dependencia entre dos Máquinas :Param hostFather = Máquina Padre de la dependencia (Física o Virtual) :Param hostChild = Máquina Hijo de la dependencia (Física o Virtual)

Figura 47 - Endpoint Crear dependencia

Historia de usuario	Eliminar dependencia
ID	HU20
Como	Usuario técnico o genérico
Quiero	poder dar de baja una dependencia
Para	eliminar el orden de apagado/encendido de las maquinas físicas o virtuales que pertenezcan a la dependencia.
Criterios de aceptación	
<ul style="list-style-type: none"> ○ El usuario deberá haber dado de alta más de una máquina física o un pool. ○ La máquina deberá tener mínimo una máquina dependiente. 	

Tabla 22 - HU20 Eliminar dependencia

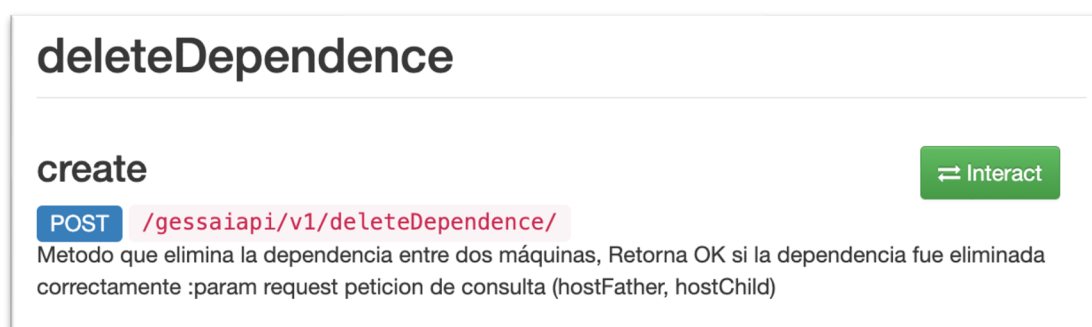


Figura 48 - Endpoint Eliminar dependencia

Historia de usuario	Listar dependencias
ID	HU21
Como	Usuario técnico o genérico
Quiero	poder consultar todas las dependencias que he dado de alta en la aplicación
Para	Tener de manera visual y en forma de árbol jerárquico, el nombre de todas las máquinas físicas y virtuales
Criterios de aceptación	
<ul style="list-style-type: none"> ○ El usuario deberá tener mínimo una dependencia creada. 	

Tabla 23 - HU21 Listar dependencias



Figura 49 - Endpoint Listar dependencias

5.3.3. CRUD Modelo User

Ya que el departamento cuenta con diferentes grupos de investigación los cuales serán usuarios de la aplicación, se tomó la decisión de tener usuarios locales que permiten ser compartidos entre varias personas que pertenezcan al mismo grupo de investigación, a continuación, se detallan los requisitos necesarios para el modelo de usuarios y sus endpoint empleados.

Historia de usuario	Crear usuario
ID	HU22
Como Usuario técnico	
Quiero Dar de alta a un usuario en la aplicación	
Para Asignarlo a las diferentes personas encargadas de las máquinas del departamento.	
Criterios de aceptación	
<ul style="list-style-type: none"> ○ Solo el usuario técnico podrá crear usuarios y asignar roles. 	

Tabla 24 - HU22 Crear usuario

create Interact

POST /gessaiaapi/v1/users/

Clase que contiene los metodos genericos para la vista de los User ['get', 'post', 'put', 'delete']

Request Body

The request body should be a "application/json" encoded object, containing the following items.

Parameter	Description
password required	
last_login	
is_superuser	Indica que este usuario tiene todos los permisos sin asignárselos explícitamente.
username required	Requerido. 150 caracteres como máximo. Únicamente letras, dígitos y @/./+/-/_
first_name	
last_name	
email	
is_staff	Indica si el usuario puede entrar en este sitio de administración.
is_active	Indica si el usuario debe ser tratado como activo. Desmarque esta opción en lugar de borrar la cuenta.
date_joined	
groups	Los grupos a los que pertenece este usuario. Un usuario tendrá todos los permisos asignados a cada uno de sus grupos.
user_permissions	Permisos específicos para este usuario.

Figura 50 - Endpoint Crear usuario

Historia de usuario	Actualizar usuario
ID	HU23
Como Usuario técnico	
Quiero Editar la información de un usuario	
Para Corregir posibles cambios en el futuro.	
Criterios de aceptación	
<ul style="list-style-type: none"> ○ Solo el usuario técnico podrá editar usuarios. 	

Tabla 25 - HU23 Actualizar usuario

update Interact

PUT `/gessaiapi/v1/users/{id}/`

Clase que contiene los metodos genericos para la vista de los User ['get', 'post', 'put', 'delete']

Path Parameters

The following parameters should be included in the URL path.

Parameter	Description
<code>id</code> required	

Request Body

The request body should be a `"application/json"` encoded object, containing the following items.

Parameter	Description
<code>password</code> required	
<code>last_login</code>	
<code>is_superuser</code>	Indica que este usuario tiene todos los permisos sin asignárselos explícitamente.
<code>username</code> required	Requerido. 150 caracteres como máximo. Únicamente letras, dígitos y @/./+/-/_
<code>first_name</code>	
<code>last_name</code>	
<code>email</code>	
<code>is_staff</code>	Indica si el usuario puede entrar en este sitio de administración.
<code>is_active</code>	Indica si el usuario debe ser tratado como activo. Desmarque esta opción en lugar de borrar la cuenta.
<code>date_joined</code>	
<code>groups</code>	Los grupos a los que pertenece este usuario. Un usuario tendrá todos los permisos asignados a cada uno de sus grupos.
<code>user_permissions</code>	Permisos específicos para este usuario.

Figura 51 - Endpoint Actualizar usuario

Historia de usuario	Listar usuarios
ID	HU24
Como Usuario técnico	
Quiero Listar todos los usuarios que están dados de alta en la aplicación	
Para Tener un control sobre ellos, realizando alguna acción o visualizar su información.	
Criterios de aceptación	
<ul style="list-style-type: none"> ○ Solo el usuario técnico podrá ver los usuarios del sistema. 	

Tabla 26 - HU24 Listar usuarios

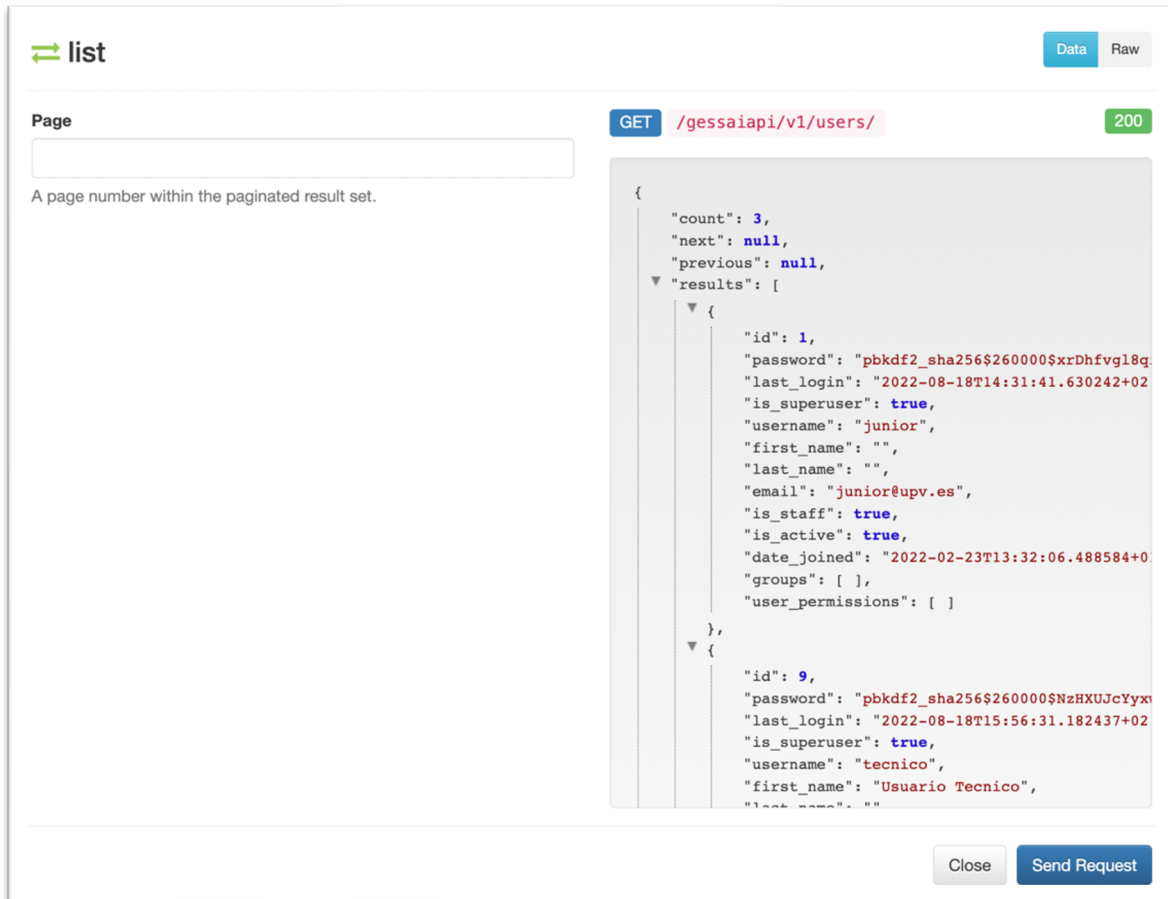


Figura 52 - Endpoint Listar usuarios

Historia de usuario	Eliminar usuario
ID	HU25
Como	Usuario técnico
Quiero	Dar de baja a un usuario en la aplicación
Para	Corregir errores o no permitir el acceso a la aplicación.
Criterios de aceptación	
<ul style="list-style-type: none"> ○ Solo el usuario técnico podrá eliminar los usuarios del sistema. 	

Tabla 27 - HU25 Eliminar usuario

delete Interact

DELETE /gessaiapi/v1/users/{id}/

Clase que contiene los metodos genericos para la vista de los User ['get', 'post', 'put', 'delete']

Path Parameters

The following parameters should be included in the URL path.

Parameter	Description
id required	

Figura 53 - Endpoint Eliminar usuario

5.3.4. Funcionalidades auxiliares

Siempre existen funcionalidades auxiliares que nos ayudan a cumplir con ciertos requisitos del sistema, en este caso contamos con 2, las cuales nos sirven para buscar una máquina por su nombre y para saber cuáles son las máquinas que dependen de él. A continuación, se detallan las historias de usuario y sus endpoint empleados.

Historia de usuario	Consultar máquinas dependientes de otra
ID	HU26
Como	Usuario técnico o genérico
Quiero	Consultar las máquinas hijo dependientes de un padre en especifico
Para	Saber cuáles son las máquinas que se deben apagar antes que el padre.
Criterios de aceptación	
<ul style="list-style-type: none"> ○ La máquina mínimo debe tener una dependencia. 	

Tabla 28 - HU26 Consultar máquinas dependientes de otra

hosts-children

create Interact

POST /gessaiapi/v1/hosts-children/

Retorna las maquinas (Host) Hijos de un Padre :param request peticion de consulta (hostFather)

Figura 54 - Endpoint Consultar máquinas dependientes de otra

Historia de usuario	Existencia de una máquina
ID	HU27
Como	Usuario técnico o genérico
Quiero	consultar la existencia de una máquina que se encuentre en la base de datos de la universidad (Piolín)
Para	recuperar la información como la ip, Mac ... etc.
Criterios de aceptación	
<ul style="list-style-type: none"> o La aplicación debe contar con permisos de acceso a Piolín. 	

Tabla 29 - HU27 Existencia de una máquina

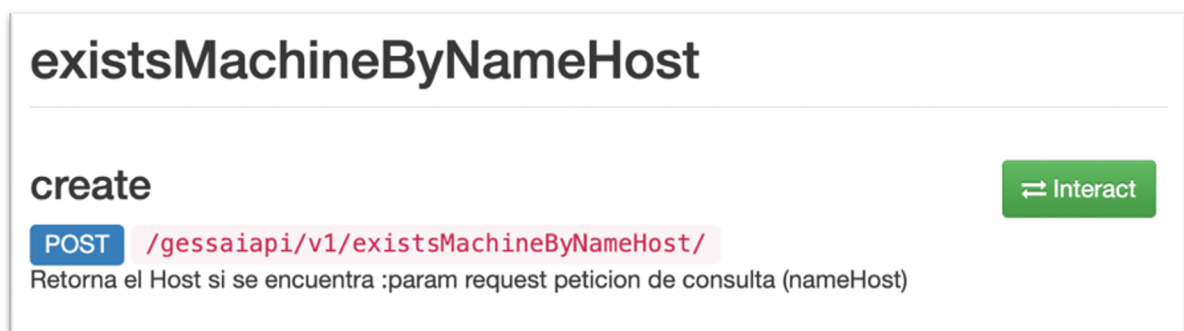


Figura 55 - Endpoint Existencia de una máquina

5.3.5. Login API

Historia de usuario	Crear login de sesión por token
ID	HU28
Como	Usuario técnico o genérico
Quiero	Poder ingresar al sistema realizando un login
Para	Hacer uso de las funcionalidades de la aplicación.
Criterios de aceptación	
<ul style="list-style-type: none"> o El usuario tiene que estar dado de alta anteriormente en el sistema. 	

Tabla 30 - HU28 Crear login de sesión por token

CAPÍTULO 6:

Sprint-03

El Sprint-03 se enfocó en desarrollar una aplicación independiente, completamente hecha en React y dirigida únicamente al Frontend, la cual se encarga de solicitar datos al Backend por medio de solicitudes HTTP. A continuación, se presenta el último de los tres capítulos del desarrollo realizado en Sprints, a diferencia de los dos sprint anteriores los cuales fueron orientados únicamente en el Backend, este último sprint se utilizó para realizar el desarrollo del Frontend, por lo que en este capítulo no será necesario presentar antecedentes, sin embargo encontrara el resumen del Sprint Backlog, las funcionalidades y el detalle de las historias de usuario acompañadas de los diferentes Wireframe de alta fidelidad.

6.1. Backlog de historias de usuario

Este apartado presentara un resumen del último Sprint Backlog realizado durante el proyecto con la ayuda de la herramienta Azure DevOps, ya que este desarrollo fue enfocado al Frontend como aplicación independiente, todas sus funcionalidades requirieron de una interfaz gráfica, en la Figura 56, puede ver las historias de usuario agrupadas por funcionalidades las cuales serán detalladas más adelante.

Order	Parent	Title	Work Item Type	State	Assigned To
1	Login	Formulario de login	User Story	Resolved	Alejandro Caicedo Palacios
2	Interfaz Modulo Sais	Crear o Editar SAI	User Story	Resolved	Alejandro Caicedo Palacios
3	Interfaz Modulo Sais	Listar Sais	User Story	Resolved	Alejandro Caicedo Palacios
4	Interfaz Modulo Pools	Crear o Editar Pool	User Story	Resolved	Alejandro Caicedo Palacios
5	Interfaz Modulo Pools	Listar Pools	User Story	Resolved	Alejandro Caicedo Palacios
6	Interfaz Modulo Máquinas	Crear o Editar máquina física	User Story	Resolved	Alejandro Caicedo Palacios
7	Interfaz Modulo Máquinas	Listar máquinas físicas	User Story	Resolved	Alejandro Caicedo Palacios
8	Interfaz Modulo Máquinas	Listar todas los arboles de dependencias	User Story	Resolved	Alejandro Caicedo Palacios
9	Interfaz Modulo Usuarios	Crear o Editar Usuario	User Story	Resolved	Alejandro Caicedo Palacios
10	Interfaz Modulo Usuarios	Listar Usuarios	User Story	Resolved	Alejandro Caicedo Palacios
11	Interfaz Modulo Dependencias	Crear dependencias	User Story	Resolved	Alejandro Caicedo Palacios
12	Interfaz Modulo Dependencias	Eliminar dependencias	User Story	Resolved	Alejandro Caicedo Palacios
13	Interfaz Modulo Dependencias	Listar árbol de dependencia	User Story	Resolved	Alejandro Caicedo Palacios
14	Interfaz Modulo Baterías	Visualizar nivel de baterías SAI	User Story	Resolved	Alejandro Caicedo Palacios
15	Interfaz Modulo Instrucciones	Descargar archivo de clave publica	User Story	Resolved	Alejandro Caicedo Palacios

Figura 56 - Sprint Backlog de historias de usuario, Sprint-03

Este último Sprint fue realizado en el periodo comprendido entre el 14 de marzo y 1 de abril, enfocado únicamente al desarrollo del Frontend y en el podemos apreciar las funcionalidades en la columna Parent y las historias de usuario en la columna Title, todas y cada una de ellas se encuentran en estado resueltas. En el siguiente apartado, se agrupan las historias de usuario por funcionalidad y se detallan cada una de ellas, indicando también su correspondiente Wireframe.

6.2. Funcionalidades

A continuación, se presentan las diferentes funcionalidades que se requirieron para el desarrollo del Frontend, con el fin de no duplicar información, en algunos casos como el de crear o editar, se utilizara el mismo Wireframe, ya que son funciones que se crearon con interfaces similares.

6.2.1. Login

La aplicación cuenta con manejo de sesiones por lo que es necesario un formulario de login, sin embargo, este será un formulario simple donde se le solicita al usuario que ingrese el usuario y la contraseña para acceder al sistema, si las credenciales son incorrectas arrojará una alerta indicándole al usuario el error que ha ocurrido.

Historia de usuario	Formulario de login
ID	HU29
Como Usuario técnico o genérico	
Quiero Poder ingresar mi usuario y contraseña por medio de un formulario de login	
Para Acceder con mi propia sesión y hacer uso de las funcionalidades de la aplicación.	
Criterios de aceptación	
<ul style="list-style-type: none"> ○ Las credenciales del usuario tienen que estar dadas de alta anteriormente en el sistema. 	

Tabla 31 - HU29 Formulario de login

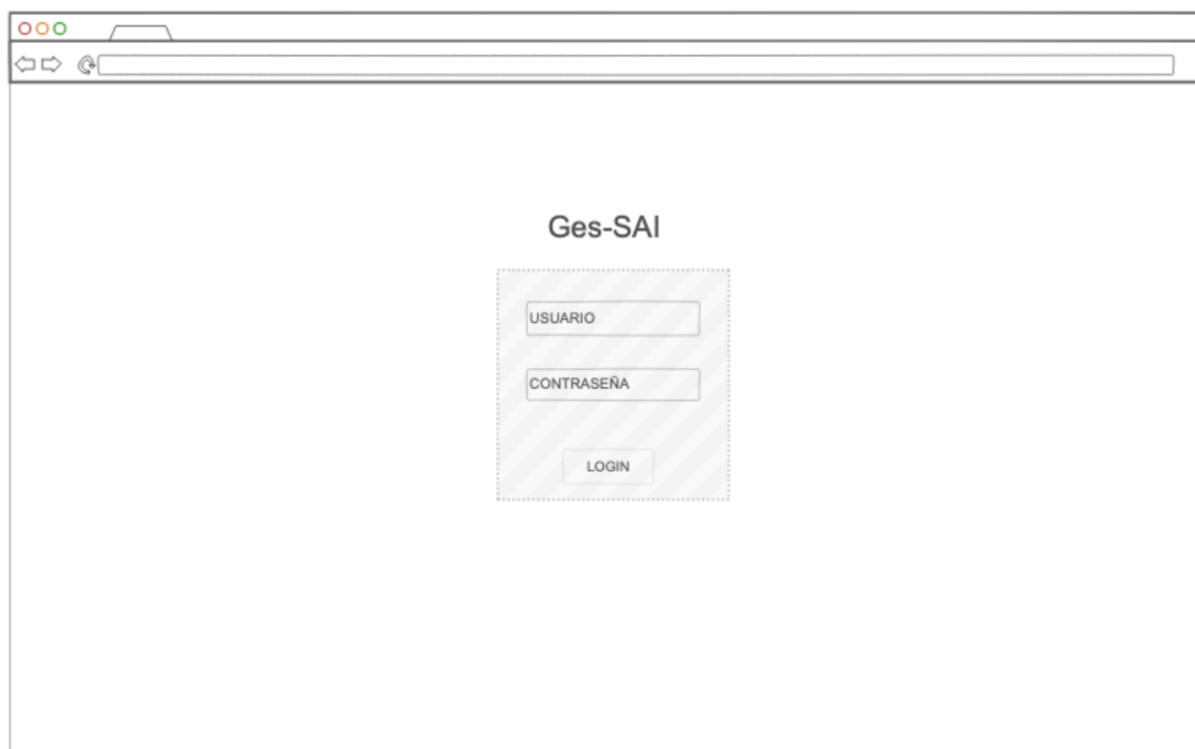


Figura 57 - Wireframe formulario de login

6.2.2. Interfaz Módulo SAI

Todas las características funcionales que puede realizar un SAI se incluyen en este módulo, las cuales son, crear, editar, listar y eliminar. Estas se resumen a continuación en las siguientes historias de usuario y Wireframes.

Historia de usuario	Crear o Editar SAI
ID	HU30
Como	Usuario técnico
Quiero	Tener una única interfaz grafica
Para	Crear o editar un SAI.

Tabla 32 - HU30 Crear o Editar SAI

Dependiendo de la opción que el usuario elija, podrá dar de alta o editar un SAI y esto lo podrá hacer desde la misma interfaz como se ve a continuación.

Figura 58 - Wireframe Sai

Historia de usuario	Listar SAIS
ID	HU31
Como	Usuario técnico
Quiero	Visualizar una lista con todos los SAIS que tengo dados de alta
Para	Elegir individualmente si quiero editar o eliminar alguno.

Tabla 33 - HU31 Listar SAIS

A continuación, se muestra el Wireframe donde el usuario puede visualizar en un listado todos los SAIS que ha dado de alta junto con el detalle de su información, aparte de esto también tiene las opciones de editar o borrar el SAI que desee.

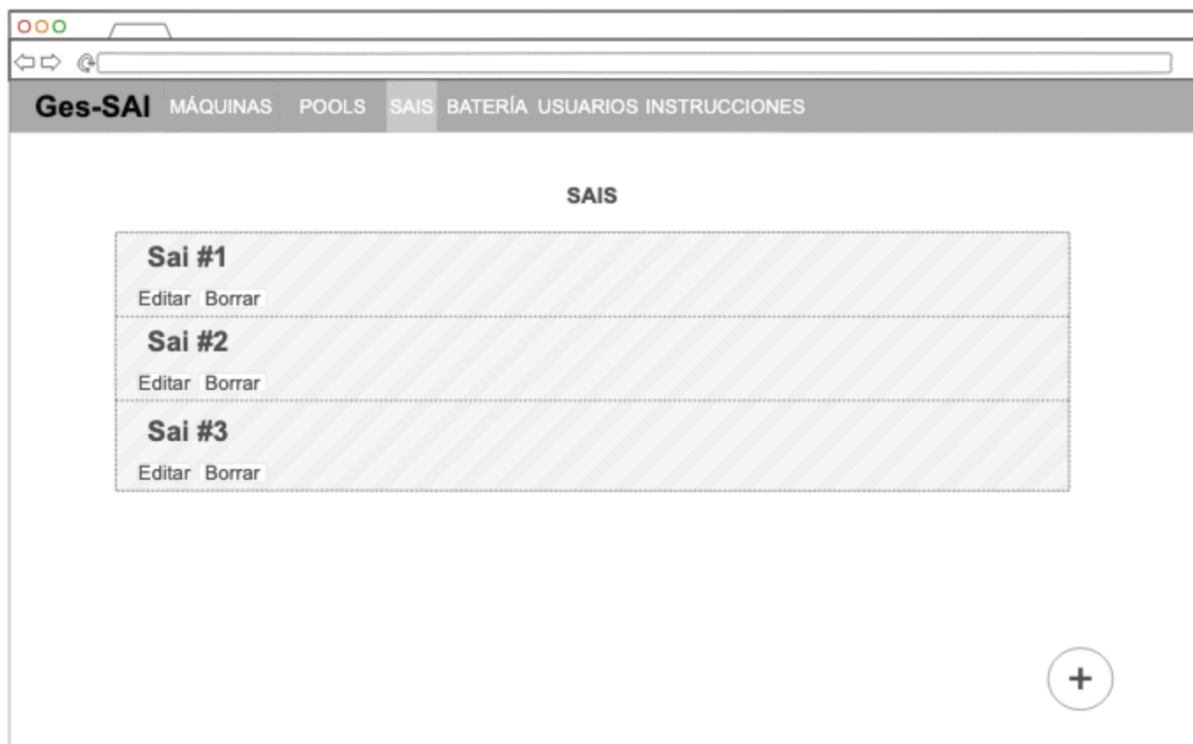


Figura 59 - Wireframe Sais

6.2.3. Interfaz Módulo Pool

Las características funcionales que puede realizar un Pool se incluyen en este módulo, las cuales son, crear, sincronizar, editar, listar y eliminar. A continuación, estas se detallan en las siguientes historias de usuario y Wireframe.

Historia de usuario	Crear o Editar Pool
ID	HU32
Como	Usuario técnico o genérico
Quiero	Tener una única interfaz grafica
Para	Crear o editar un Pool.

Dependiendo de la opción que el usuario elija, podrá dar de alta o editar un Pool y esto lo podrá hacer desde la misma interfaz gráfica como se presenta en la siguiente imagen.

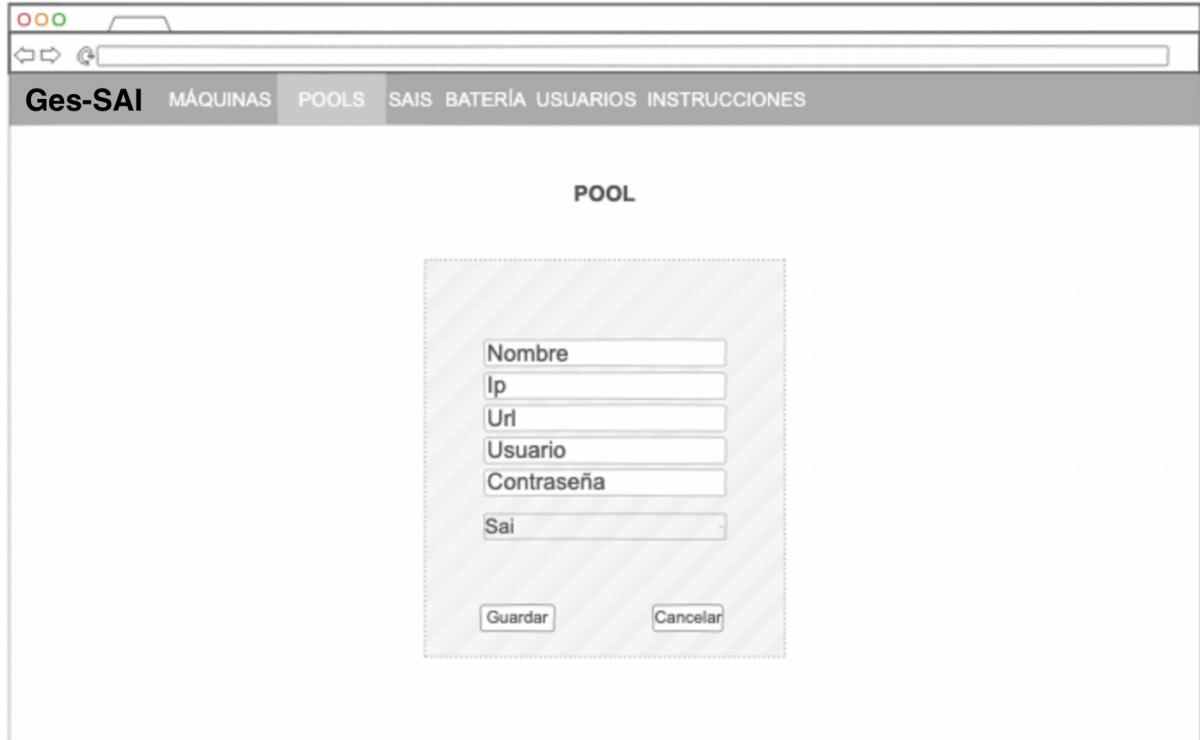


Figura 60 - Wireframe Pool

Historia de usuario	Listar Pools
ID	HU33
Como	Usuario técnico o genérico
Quiero	Visualizar una lista con todos los Pools que tengo dados de alta
Para	Elegir individualmente si quiero editar, sincronizar o eliminar alguno.

Tabla 34 - HU33 Listar Pools

A continuación, se muestra el Wireframe donde el usuario puede visualizar en un listado todos los Pools que ha dado de alta junto con el detalle de su información, aparte de esto también tiene las opciones de editar, sincronizar y borrar el Pool que desee.

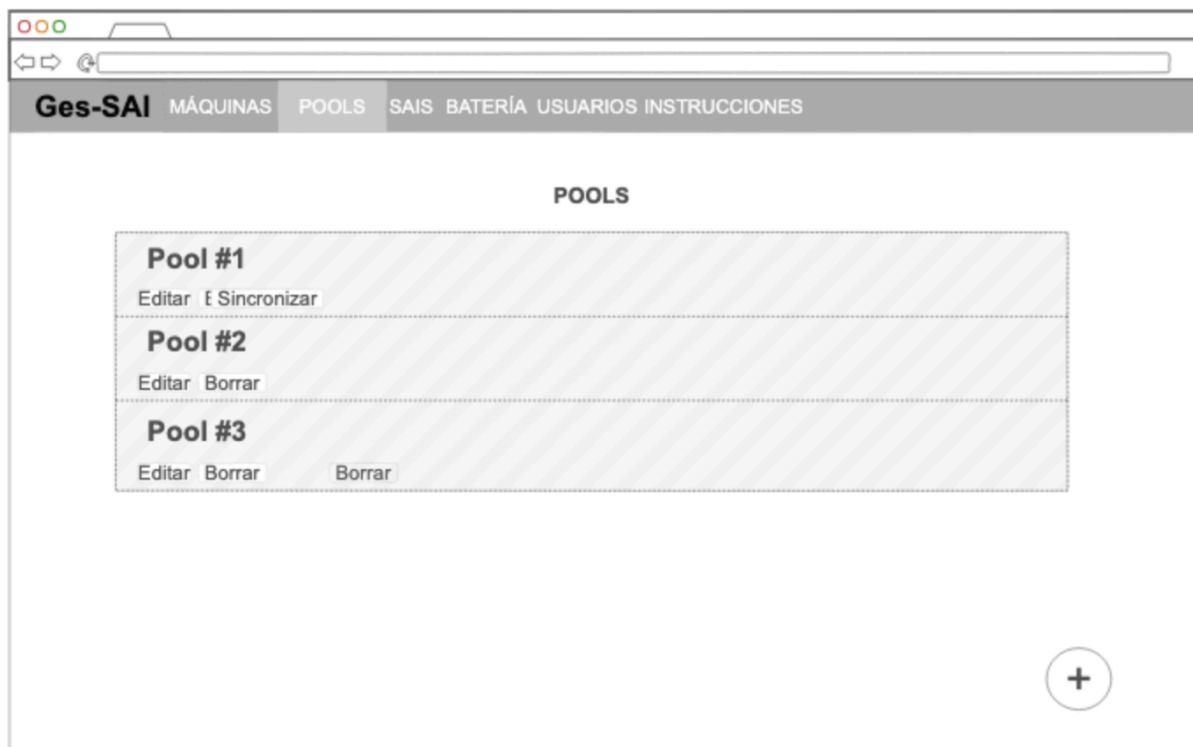


Figura 61 - Wireframe Pools

6.2.4. Interfaz Módulo Máquinas

El módulo de máquinas es un poco diferente a los dos anteriores, ya que desde este no solo se podrá listar las máquinas dadas de alta, sino que también incluirá los árboles de dependencia dados de alta, todas las características funcionales como crear, editar, listar y eliminar, se incluyen en este módulo. A continuación, estas se detallan en las siguientes historias de usuario y Wireframes.

Historia de usuario	Crear o Editar Máquinas físicas
ID	HU34
Como	Usuario técnico o. genérico
Quiero	Tener una única interfaz grafica
Para	Crear o editar una Máquina física.

Tabla 35 - HU34 Crear o Editar Máquinas físicas

Dependiendo de la opción que el usuario elija, podrá dar de alta o editar una Máquina y esto lo podrá hacer desde la misma interfaz gráfica como se presenta en la siguiente imagen.



Figura 62 – Wireframe Máquina

Historia de usuario	Listar Máquinas físicas
ID	HU35
Como	Usuario técnico o genérico
Quiero	Visualizar una lista con todas las Máquinas y dependencias que tengo dadas de alta
Para	Elegir individualmente si quiero editar o eliminar alguna.

Tabla 36 - HU35 Listar Máquinas físicas

A continuación, se muestra el Wireframe donde el usuario puede visualizar en un listado todas las máquinas que ha dado de alta junto con el detalle de su información, aparte de esto también tiene las opciones de editar o borrar la máquina que desee, sin embargo, en la parte baja del formulario también cuenta con la opción de visualizar los árboles de dependencia dados de alta.



Figura 63 - Wireframe Máquinas

6.2.5. Interfaz Módulo Usuarios

El usuario técnico es el único con los permisos de, crear, editar, listar y eliminar otros usuarios. A continuación, estas características funcionales se detallan en las siguientes historias de usuario y Wireframes.

Historia de usuario	Crear o Editar Usuario
ID	HU36
Como Usuario técnico	
Quiero Tener una única interfaz grafica	
Para Crear o editar un Usuario.	

Tabla 37 - HU36 Crear o Editar Usuario

Dependiendo de la opción que el usuario técnico elija, podrá dar de alta o editar un usuario en el sistema y esto lo podrá hacer desde la misma interfaz gráfica como se presenta en la siguiente imagen.



Figura 64 - Wireframe Usuario

Historia de usuario	Listar Usuarios
ID	HU37
Como	Usuario técnico
Quiero	Visualizar una lista con todos los Usuarios que están dados de alta en el sistema
Para	Elegir individualmente si quiero editar o eliminar alguno.

Tabla 38 - HU37 Listar Usuarios

A continuación, se muestra el Wireframe donde el usuario técnico puede visualizar en un listado todos los usuarios dados de alta en el sistema junto con el detalle de su información, aparte de esto también cuenta con las opciones de editar o borrar el usuario que desee.

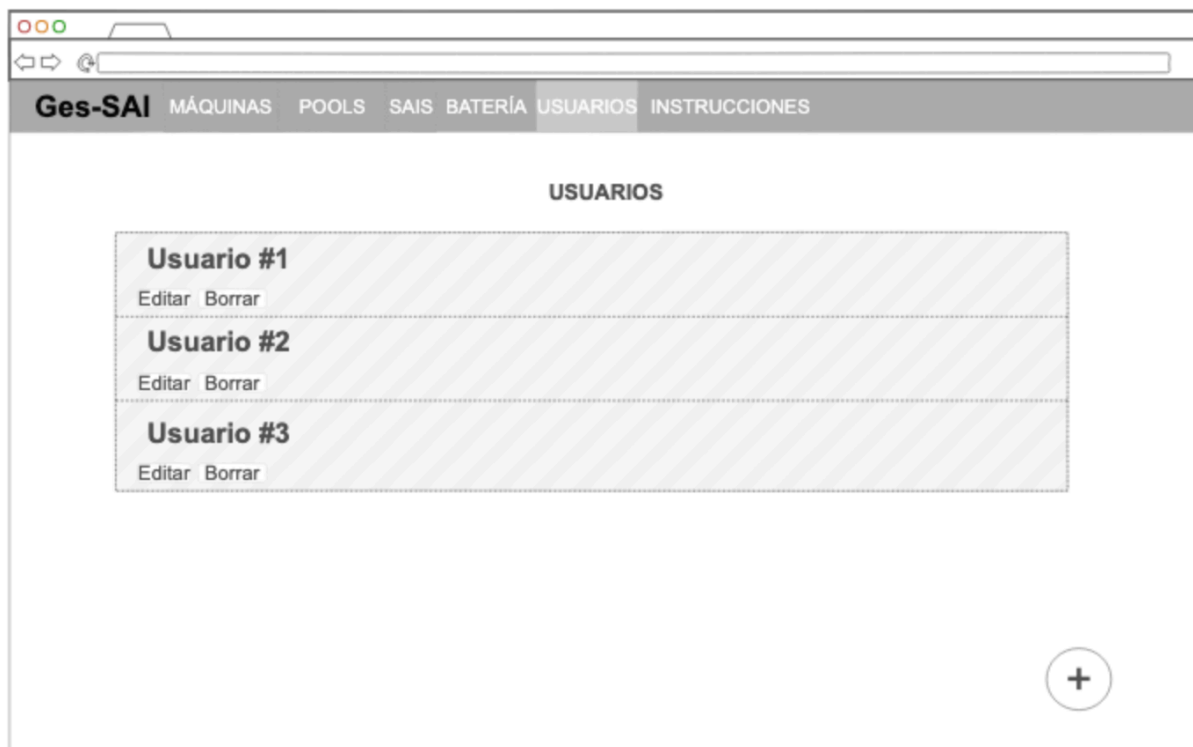


Figura 65 - Wireframe Usuarios

6.2.6. Interfaz Módulo Dependencias

Este módulo es totalmente diferente a todos los demás, ya que desde una única interfaz se podrá realizar todas las características funcionales que este requiere, como crear o eliminar una dependencia y al mismo tiempo visualizar el árbol de dependencias de una máquina. A continuación, se detallan tres historias de usuario acompañadas de un único Wireframe el cual las engloba a todas.

Historia de usuario	Crear dependencia
ID	HU38
Como	Usuario técnico o genérico
Quiero	Dar de alta a una dependencia entre dos máquinas, ya sean físicas o virtuales
Para	Incluir las en el sistema automático de encendido y apagado.

Tabla 39 - HU38 Crear dependencia

Historia de usuario	Eliminar dependencia
ID	HU39
Como	Usuario técnico o genérico
Quiero	Dar de baja una dependencia entre dos máquinas, ya sean físicas o virtuales
Para	Excluir las del sistema automático de encendido y apagado.

Tabla 40 - HU39 Eliminar dependencia

Historia de usuario	Listar árbol de dependencia
ID	HU40
Como	Usuario técnico o genérico
Quiero	Visualizar el árbol de dependencias de una máquina padre que seleccione
Para	Llevar un control de todas sus máquinas hijo.

Tabla 41 - HU40 Listar árbol de dependencia

Como se mencionó anteriormente el siguiente Wireframe incluye todas las tres historias de usuario, en la parte superior de la pantalla se encuentra el apartado de alta dependencia, en la parte inferior izquierda de la pantalla se encuentra el apartado de eliminar dependencia, y por último en la parte inferior derecha de la pantalla, se encuentra el árbol de dependencia de la máquina seleccionada.



Figura 66 - Wireframe Dependencias

6.2.7. Interfaz Módulo Baterías

A este módulo solo puede acceder el usuario técnico, pudiendo visualizar todos los SAIS que se encuentran dados de alta en el sistema y desde allí llevar un control del nivel de las baterías de cada uno.

Historia de usuario	Visualizar nivel de baterías
ID	HU41

Como

Usuario técnico

Quiero

Visualizar el nivel de baterías de todos los SAIS dados de alta en el sistema

Para

Llevar un control del tiempo restante de cada uno.

Tabla 42 - HU41 Visualizar nivel de baterías

La mejor forma para presentar el nivel restante de baterías de los SAIS es en un diagrama de barras, a continuación, se presenta el Wireframe requerido.

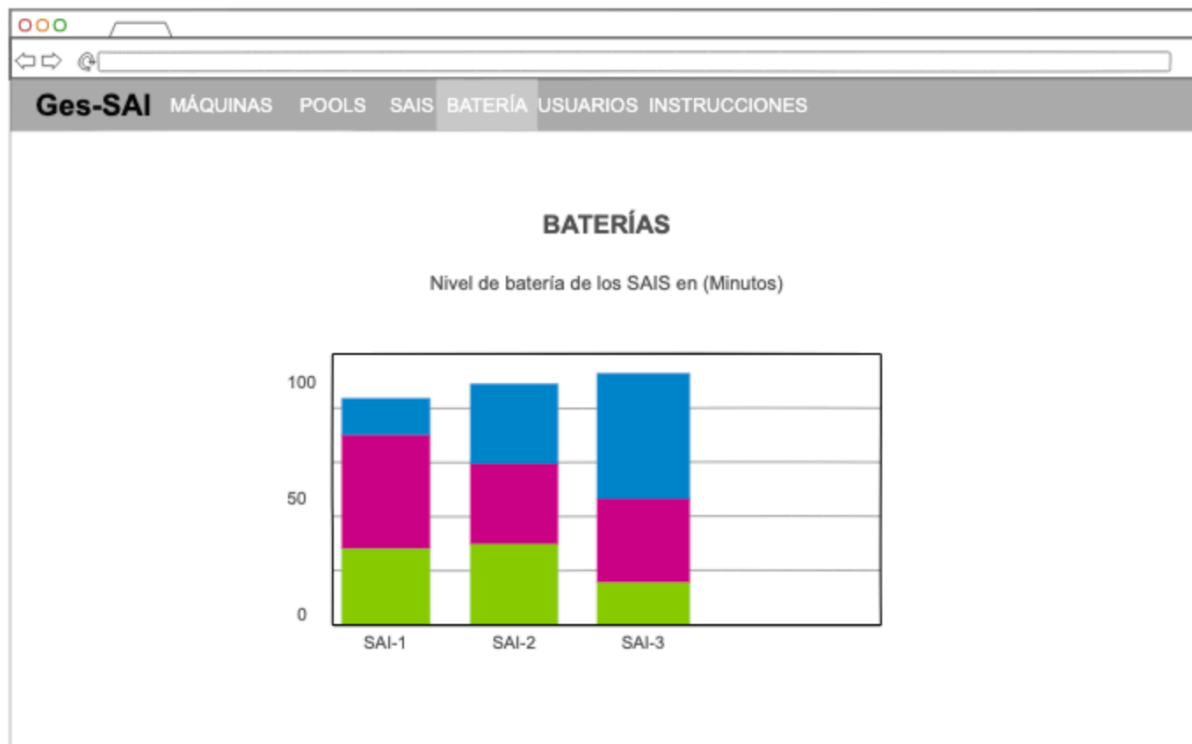


Figura 67 - Wireframe Grafica del nivel restante de baterías

6.2.8. Interfaz Módulo Instrucciones

Este módulo al igual que el anterior es solo visual, con la diferencia que desde el formulario el usuario podrá descargar un archivo el cual contiene la clave pública del servidor, todo esto con el fin de que el servidor de Ges-SAI tenga acceso a la máquina que aloje ese archivo descargado.

Historia de usuario	Descargar archivo de clave pública
ID	HU42
Como	Usuario técnico o genérico
Quiero	Visualizar las instrucciones de como descargar y ubicar el archivo de clave publica en la máquina que deseo incluir en el sistema de apagado o encendido
Para	No cometer errores a la hora de descargar el archivo.

Tabla 43 - HU42 Descargar archivo de clave pública

El siguiente Wireframe es la forma de representar un formulario con las instrucciones que el usuario debe seguir para descargar la clave pública, esto con el fin de que Ges-SAI tenga acceso a la máquina física que contenga el archivo.

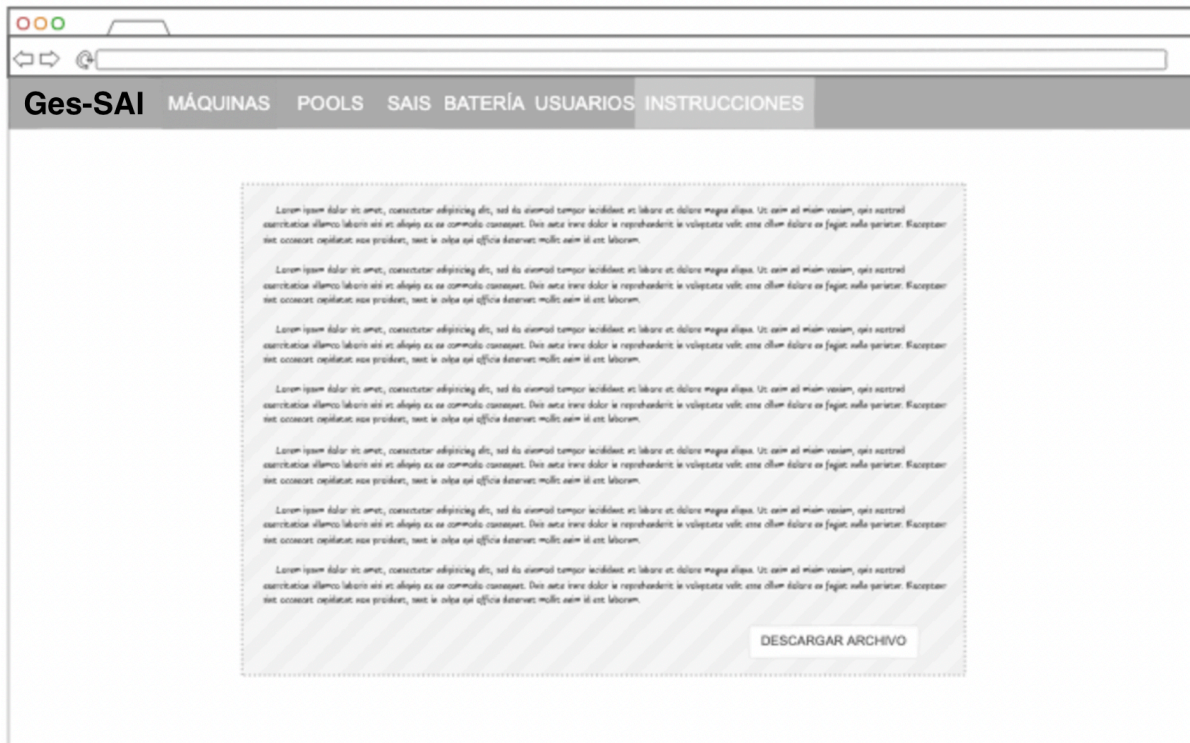


Figura 68 - Wireframe Instrucciones

CAPÍTULO 7:

Pruebas

En este capítulo encontrara las diferentes pruebas efectuadas al sistema a lo largo de todo el desarrollo, todas estas han sido pruebas manuales y realizadas al final de cada Sprint, esto con el fin de validar que todos los objetivos se cumplieran satisfactoriamente y sin errores.

Luego de que todo el sistema estuviera completo he integrado, se ha ejecutado para cada caso de uso al menos una prueba manual, con el fin de verificar que todo funciona correctamente.

7.1. Pruebas del Backend API

Para cada uno de los API endpoint empleados, se han realizado llamadas esperando el retorno de datos correctos. Durante estas pruebas se ha comprobado que el funcionamiento del Backend API es correcto, con objeto de evitar errores en el Frontend.

A continuación, se presentan 2 imágenes como ejemplo de las diferentes pruebas realizadas a la API a través de la herramienta Postman, la Figura 69 muestra cómo obtener el token de login a la API, y la Figura 70 muestra la obtención de todas las máquinas dadas de alta en el sistema.

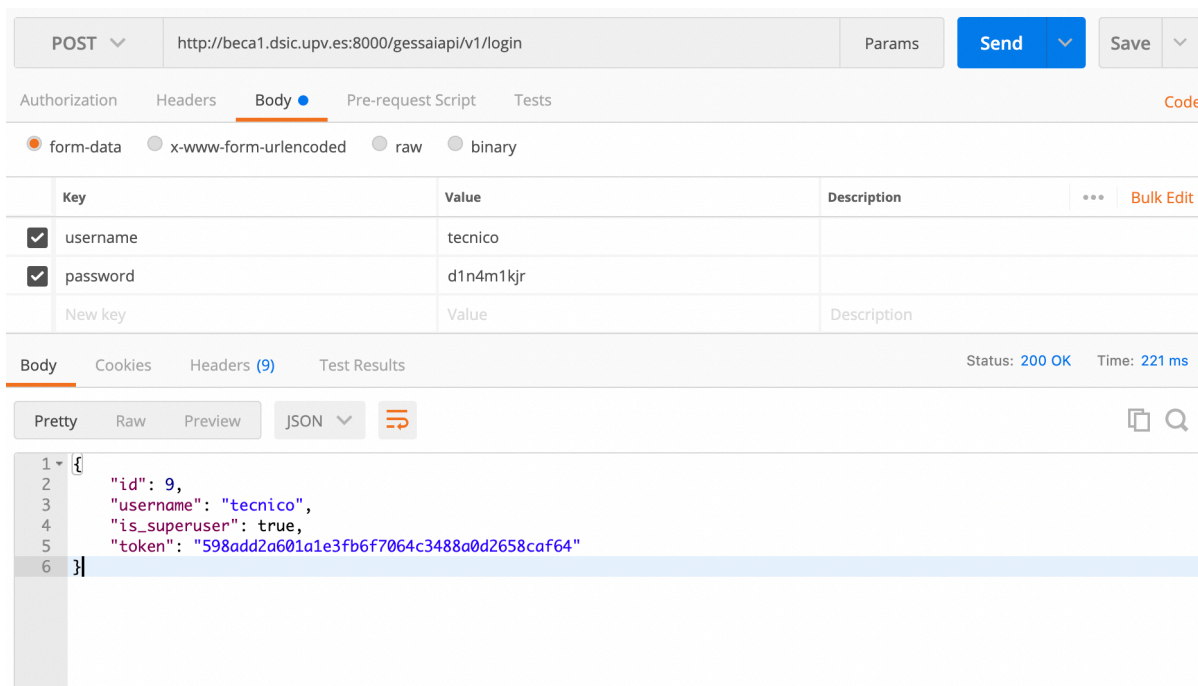


Figura 69 - Prueba de Backend, Obtener el token de login

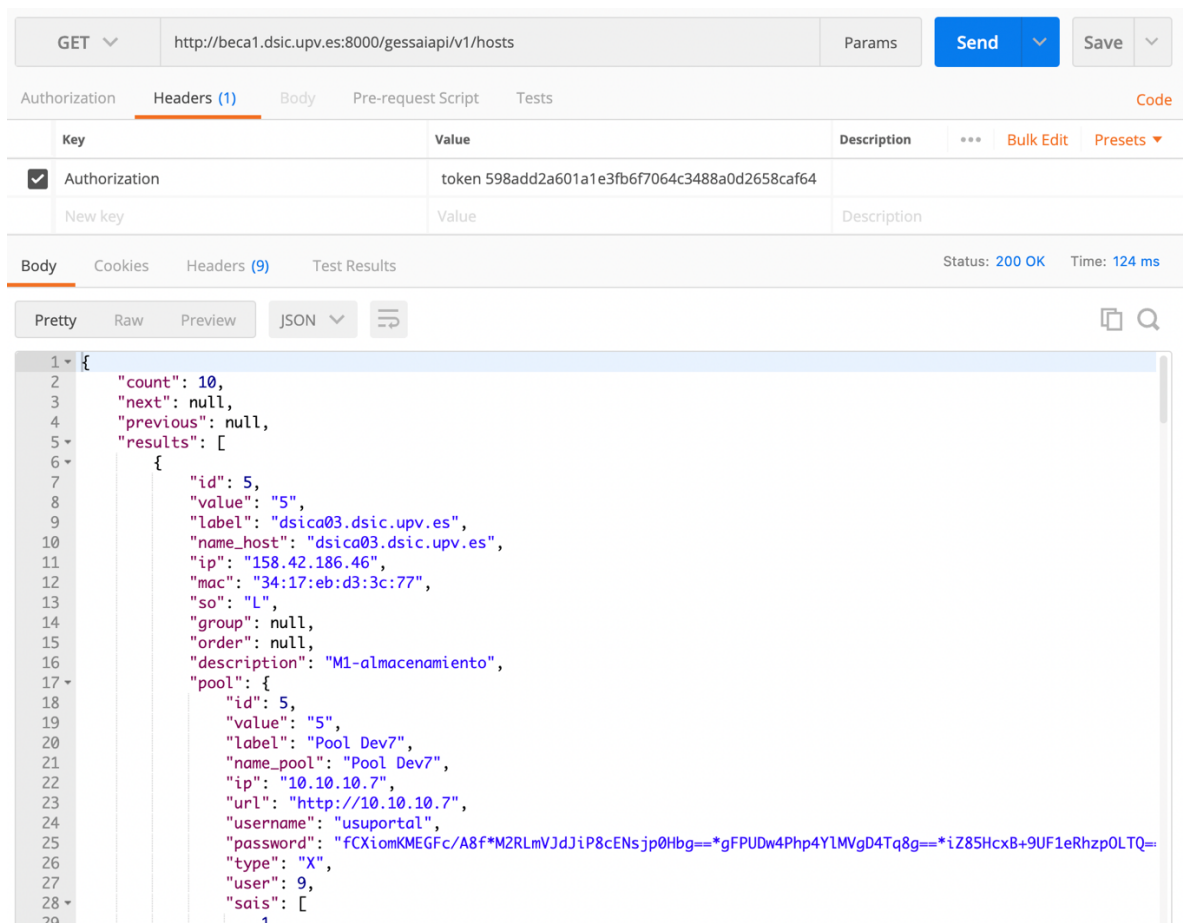


Figura 70 - Prueba de Backend, Obtener todas las Máquinas

7.2. Pruebas del Frontend

En este caso se utilizó la herramienta React Testing Library, esta es una librería que le permite al desarrollador realizar diferentes testing a componentes de React, los componentes en React son funciones que reciben propiedades y estas se renderizan en el Frontend, por lo que es bastante útil testear el Frontend y encontrar fallos en los diferentes componentes.

7.3. Pruebas de API XenAPI

Como ya se mencionó, para probar la conexión a los diferentes Pools y máquinas virtuales, se utilizó la librería de Python llamada xenapi. Las pruebas se realizaron utilizando la consola de Python, a continuación, se presentan 2 imágenes como ejemplo de las diferentes pruebas realizadas a los procedimientos remotos de XenAPI, la Figura 71 muestra cómo obtener una sesión de consulta, y la Figura 72 muestra cómo obtener todas las máquinas virtuales de un Pool.

```

bin — python3 — 74x13
(env_tfm_api) (base) alejojr@veraw140-160 bin % python3
Python 3.8.5 (default, Sep  4 2020, 02:22:02)
[Clang 10.0.0 ] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more information.
[>>> from XenAPI import XenAPI
[>>> session = XenAPI.Session('http://10.10.10.8')
[>>> print(session)
<Session for 10.10.10.8/RPC2>
[>>> session.xenapi.login_with_password('XXXXXXXXXX', 'XXXXXXXXXX')
[>>> print(session._session)
OpaqueRef:20df982f-b298-4dfd-a0ee-47252c2d6376
>>> █

```

Figura 71 - Prueba de XenAPI, Obtener la sesión de Api

```

alejojr — python3 — 80x24
[>>> from XenAPI import XenAPI
[>>> session = XenAPI.Session('http://10.10.10.8')
[>>> session.xenapi.login_with_password('XXXXXXXXXX', 'XXXXXXXXXX')
[>>> records = session.xenapi.VM.get_all_records_where('field "is_a_template" = "
false" and "field "is_a_snapshot" = "false" and "field "is_control_domain" = "
false"')
[>>> print(records)
{'OpaqueRef:f730f8b5-31b1-4e44-a79c-d9b034cc79df': {'uuid': 'f88c2053-c5aa-bc29-
bf14-9f644127afe3', 'allowed_operations': ['changing_NVRAM', 'changing_dynamic_r
ange', 'changing_shadow_memory', 'changing_static_range', 'make_into_template',
'migrate_send', 'destroy', 'export', 'start_on', 'start', 'clone', 'copy', 'snap
shot'], 'current_operations': {}, 'name_label': 'JAV-002-p5', 'name_description'
: '', 'power_state': 'Halted', 'user_version': '1', 'is_a_template': False, 'is_
default_template': False, 'suspend_VDI': 'OpaqueRef:NULL', 'resident_on': 'Opaqu
eRef:NULL', 'scheduled_to_be_resident_on': 'OpaqueRef:NULL', 'affinity': 'Opaqu
eRef:NULL', 'memory_overhead': '37748736', 'memory_target': '0', 'memory_static_m
ax': '4194304000', 'memory_dynamic_max': '4194304000', 'memory_dynamic_min': '41
94304000', 'memory_static_min': '134217728', 'VCPUs_params': {'weight': '256'},
'VCPUs_max': '2', 'VCPUs_at_startup': '2', 'actions_after_shutdown': 'destroy',
'actions_after_reboot': 'restart', 'actions_after_crash': 'restart', 'consoles':
[], 'VIFs': ['OpaqueRef:363d3b1e-7273-4d20-a2c5-8c2b37034ae2'], 'VBDs': ['Opaqu
eRef:a4bc91da-b781-42ab-8bbd-dd9ae4ffaf78', 'OpaqueRef:1af122ac-1b7f-42af-bc50-2
413c507cbde'], 'USBs': [], 'crash_dumps': [], 'VTPMs': [], 'PV_bootloader': '',
'PV_kernel': '', 'PV_ramdisk': '', 'PV_args': '', 'PV_bootloader_args': ''}

```

Figura 72 - Prueba de XenAPI, Obtener todas las máquinas virtuales

CAPÍTULO 8:

Conclusión

Este capítulo detalla las conclusiones obtenidas tras la realización del proyecto, junto con los posibles trabajos a futuro que se puedan llevar a cabo.

Tener el privilegio de realizar este proyecto me ha permitido reforzar los conocimientos en React y el desarrollo de APIs, y aún más contando que el desarrollo fue dividido en dos partes, Backend y Frontend permitiendo enfocarme en las necesidades que requerían cada una, por el lado de la API Backend la obtención y canalización de toda la información recopilada de la base de datos, y en el lado del Frontend el diseño de las interfaces de usuario y estructura de la web.

8.1. Conclusiones

El objetivo principal del TFM era desarrollar un software capaz de visualizar el nivel de batería de los dispositivos SAI y a su vez gestionar y automatizar un apagado y encendido correcto de las máquinas físicas y virtuales del DSIC, con el fin de evitar daños en estas. Este objetivo se pudo cumplir a partir de la creación de Ges-SAI, que además incorpora la forma de realizar un apagado y encendido manejado por dependencias, evitando así que los ordenadores se quedaran bloqueados.

Tener el privilegio de realizar este proyecto me ha permitido reforzar los conocimientos en React y el desarrollo de APIs, y aún más contando que el desarrollo fue dividido en dos partes, Backend y Frontend permitiendo enfocarme en las necesidades que requerían cada una, por el lado de la API Backend la obtención y canalización de toda la información recopilada de la base de datos, y en el lado del Frontend el diseño de las interfaces de usuario y estructura de la web.

Una de las mayores dificultades del proyecto fue poder interrogar de una manera exitosa al SAI, ya que se contaba con muy poca información al respecto en internet. Sin embargo, estos dispositivos al contar con una tarjeta de red que permite la implementación de diferentes protocolos, nos dimos cuenta de que nos podríamos abrir paso si utilizábamos SNMP, que, a pesar de ser antiguo, es uno de los protocolos más usados, robustos y bien definido. Aunque Ges-SAI únicamente se centró en obtener el nivel de batería, este protocolo permite obtener aún más información del dispositivo, lo que da paso a tener una administración más amplia en algún futuro.

Se logro minimizar tiempos utilizando programación en hilos, ya que esto permite ejecutar simultáneamente diferentes tareas en un mismo espacio de proceso, y así nos evitamos el hecho de tener que esperar respuesta de una máquina sin dependencia para continuar con el flujo. En lugar de eso, logramos que (N) número de máquinas físicas o virtuales, se puedan apagar o encender al mismo tiempo.

8.2. Trabajos futuros

Como cualquier proyecto, Ges-SAI deja abiertos diferentes caminos o líneas por las que es posible ampliar aún más el trabajo. A continuación, se destacan los siguientes trabajos futuros que pueden desarrollarse como resultado de este proyecto.

- Ampliar la administración de un dispositivo SAI, interrogando no solo su nivel de batería si no también el estado de la batería, voltaje de la batería, corriente de la batería, temperatura de la batería, entre otros.
- Al ser un sistema con una API como Backend, desarrollar diferentes Frontend compatibles con plataformas móviles y tables.
- Notificar por mensaje de texto al móvil de cada usuario y propietario de una máquina virtual o física que esta ha sido apagada o encendida.

Bibliografía

Serafini, F. (2011). ¿Cómo crear Pools en Citrix XenServer? Blog Virtualización & Cloud Computing en español. Obtenido de: <https://www.josemariagonzalez.es/citrix/como-crear-pools-citrix-xenserver.html#:~:text=Un%20Pool%20se%20define%20como,%2C%20etc.%20de%20forma%20centralizada>.

Castillo, J. A. (2019). SAI: Qué es, para qué sirve y que tipos hay en el mercado. Obtenido de: <https://www.profesionalreview.com/2019/02/23/que-es-sai/>

Laoyan, S. (2022). Scrumban: lo mejor de dos metodologías ágiles. Obtenido de: <https://asana.com/es/resources/scrumban>

ManageEngine. (2022). Conceptos básicos del protocolo SNMP. Obtenido de: <https://www.manageengine.com/es/network-monitoring/what-is-snmp.html#snmp-version>

Hartman, C. (2019). Tripplite Smart UPS: A Review. Obtenido de: <https://l-a.org/2019/06/14/tripplite-smart-ups-a-review/>

Red Hat. (2020). ¿Qué es un hipervisor? Obtenido de: <https://www.redhat.com/es/topics/virtualization/what-is-a-hypervisor>

Easy UPS 3S Manuals. (s.f.). Easy UPS 3S 10-40 kVA – Funcionamiento. Obtenido de: <https://www.productinfo.schneider-electric.com/easyups3s/viewer?docidentity=UserInterface-181044E9&lang=es&extension=xml&manualidentity=OperationEasyUPSSpecificProductNumber-0CD4049E>

Deyimar, A. (2022). ¿Qué es React y cómo funciona realmente? Obtenido de: https://www.hostinger.es/tutoriales/que-es-react?ppc_campaign=google_performance_max&gclid=CjwKCAjwi8iXBhBeEiwAKbUofSaJzEgKQU3uGOT7aw_bcZDzonoqyu5Y-ZyQDluYe6EHXrTm_zexVRoCANEQAvD_BwE

Reyes, A. (2018). Procesos, hilos y multihilos. Obtenido de: <http://aisii.azc.uam.mx/areyes/archivos/licenciatura/sd/U2/ConceptoHilos.pdf>

Miguel. (2016). La planificación por capas. Obtenido de: <https://proagilist.es/blog/posts/planificacion-por-capas-en-scrum-agile/#:~:text=En%20un%20proyecto%20Scrum%2C%20al,de%20planificaci%C3%B3n%20que%20se%20superponen>.

PyCharm. (2013). (versión 2022.2). [Software]. JetBrains. Obtenido de: <https://www.jetbrains.com/es-es/pycharm/>

MySQL Workbench. (2015). (versión 8.0.30). [Software]. MySQL. Obtenido de: <https://dev.mysql.com/downloads/workbench/>

GitLab. (2011). (versión 8.0.30). [Software]. GitLab Inc. Obtenido de: <https://about.gitlab.com/>

PySNMP. (2019). (versión 4.4.2). [Library Python]. Ilya Etingof. Obtenido de: <https://pysnmp.readthedocs.io/en/latest/index.html>

React. (2013). (versión 18.2.0). [Library JavaScript]. Facebook. Obtenido de: <https://es.reactjs.org/>

Django REST Framework. (2005). (versión 3.13.1). [Framework Django]. Django Software Foundation. Obtenido de: <https://www.django-rest-framework.org/>

Material UI. (2014). (versión 5.10.0). [Library React]. Material UI SAS. Obtenido de: <https://mui.com/material-ui/getting-started/overview/>

XenAPI. (2014) (versión 1.160.2). [API]. Xapi-Project. Obtenido de: <https://xapi-project.github.io/xen-api/>

MDN Web Docs. (2022). Fundamentos de JavaScript - Aprende sobre desarrollo web. Obtenido de: https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/JavaScript_basics

Anexos

Anexo A: Casos de uso

Índice de Tablas

Tabla 1 - CU-1: Iniciar sesión.....	102
Tabla 2 - CU-2: Cerrar sesión.....	103
Tabla 3 - CU-3: Alta máquina	103
Tabla 4 - CU-4: Baja máquina.....	104
Tabla 5 - CU-5: Editar máquina.....	104
Tabla 6 - CU-6: Alta pool.....	105
Tabla 7 - CU-7: Editar pool.....	105
Tabla 8 - CU-8: Baja pool.....	106
Tabla 9 - CU-9: Alta dependencia	107
Tabla 10 - CU-10: Eliminar dependencia.....	107
Tabla 11 - CU-11: Descargar archivo de clave publica.....	108
Tabla 12 - CU-12: Ver baterías SAI.....	108
Tabla 13 - CU-13: Alta Sai.....	109
Tabla 14 - CU-14: Editar Sai.....	109
Tabla 15 - CU-15: Eliminar Sai.....	110
Tabla 16 - CU-16: Alta usuario	110
Tabla 17 - CU-17: Editar usuario.....	111
Tabla 18 - CU-18: Cambiar contraseña	111
Tabla 19 - CU-19: Eliminar usuario	112
Tabla 20 - CU-20: Alta máquina virtual.....	112

Se presentan los casos de uso que intervienen en el aplicativo Ges-SAI, para ello se utilizara la siguiente plantilla de descripción, detallando el significado de cada uno de sus campos.

<i>Id</i>	Código único para cada caso de uso	
<i>Nombre</i>	Nombre del caso de uso	
<i>Actores</i>	Rol de usuario que interactúa con el caso de uso	
<i>Resumen</i>	Descripción del caso de uso	
<i>Precondición</i>	Condición para poder realizar la actividad	
<i>Flujo normal</i>	Actor	Sistema
	Flujo normal del actor	Flujo normal del sistema
<i>Flujo alternativo</i>	Actor	Sistema
	Flujo alternativo del actor	Flujo alternativo del sistema
<i>Postcondición</i>	Estado del sistema luego de ejecutar completamente el caso de uso	

Tabla 44 - Plantilla de descripción de casos de uso

<i>Id</i>	CU-1	
<i>Nombre</i>	Iniciar sesión	
<i>Actores</i>	Genérico y Técnico	
<i>Resumen</i>	El usuario inicia sesión con sus credenciales	
<i>Precondición</i>	El usuario se encuentra dado de alta en el sistema	
<i>Flujo normal</i>	Actor	Sistema
	2. El usuario rellena los campos, usuario y contraseña y pulsa el botón iniciar sesión.	1. El sistema muestra el formulario de inicio de sesión. 3. El sistema comprueba si las credenciales son válidas. 4. El sistema permite avanzar a la siguiente vista.
<i>Flujo alternativo</i>	Actor	Sistema
		3.1. El sistema detecta que los datos de usuario no son válidos. 3.2. El sistema muestra mensaje de error indicando que el usuario y la contraseña son inválidos.
<i>Postcondición</i>	El usuario ha iniciado sesión en el sistema	

Tabla 45 - CU-1: Iniciar sesión

Id	CU-2									
Nombre	Cerrar sesión									
Actores	Genérico y Técnico									
Resumen	El usuario registrado cierra su sesión									
Precondición	El usuario debe estar identificado mediante un inicio de sesión									
Flujo normal	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>2. El usuario registrado da clic en el botón “Cerrar sesión”.</td> <td>1. El sistema ofrece en todas las interfaces un botón de cierre de sesión. 3. El sistema limpia la información de la sesión del usuario. 4. El sistema vuelve a la vista de inicio de sesión.</td> </tr> <tr> <td>Actor</td> <td>Sistema</td> </tr> <tr> <td></td> <td></td> </tr> </tbody> </table>		Actor	Sistema	2. El usuario registrado da clic en el botón “Cerrar sesión”.	1. El sistema ofrece en todas las interfaces un botón de cierre de sesión. 3. El sistema limpia la información de la sesión del usuario. 4. El sistema vuelve a la vista de inicio de sesión.	Actor	Sistema		
Actor	Sistema									
2. El usuario registrado da clic en el botón “Cerrar sesión”.	1. El sistema ofrece en todas las interfaces un botón de cierre de sesión. 3. El sistema limpia la información de la sesión del usuario. 4. El sistema vuelve a la vista de inicio de sesión.									
Actor	Sistema									
Flujo alternativo										
Postcondición	El usuario cerró la sesión en el sistema.									

Tabla 46 - CU-2: Cerrar sesión

Id	CU-3							
Nombre	Alta máquina							
Actores	Genérico y Técnico							
Resumen	El usuario da de alta a una máquina en el sistema							
Precondición	<ul style="list-style-type: none"> - El dispositivo SAI debe estar dado de alta en base de datos - Si es una máquina física, esta debe estar dada de alta en la base de datos externa. 							
Flujo normal	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>1. El usuario ingresa al módulo “alta maquinas”. 2. Si es una máquina física debe ingresar el nombre de la máquina. 3. El usuario debe hacer clic en el botón “buscar”. 5. El usuario termina de llenar el formulario y hacer clic en el botón “guardar”.</td> <td>4. El sistema muestra los datos de la máquina física encontrada. 6. El sistema vuelve a la vista de máquinas físicas.</td> </tr> <tr> <td>Actor</td> <td>Sistema</td> </tr> </tbody> </table>		Actor	Sistema	1. El usuario ingresa al módulo “alta maquinas”. 2. Si es una máquina física debe ingresar el nombre de la máquina. 3. El usuario debe hacer clic en el botón “buscar”. 5. El usuario termina de llenar el formulario y hacer clic en el botón “guardar”.	4. El sistema muestra los datos de la máquina física encontrada. 6. El sistema vuelve a la vista de máquinas físicas.	Actor	Sistema
Actor	Sistema							
1. El usuario ingresa al módulo “alta maquinas”. 2. Si es una máquina física debe ingresar el nombre de la máquina. 3. El usuario debe hacer clic en el botón “buscar”. 5. El usuario termina de llenar el formulario y hacer clic en el botón “guardar”.	4. El sistema muestra los datos de la máquina física encontrada. 6. El sistema vuelve a la vista de máquinas físicas.							
Actor	Sistema							
Flujo alternativo	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>2.a.1. Si es una máquina de almacenamiento debe dar clic en la opción “almacenamiento”. 2.a.2. El usuario debe rellenar todos los campos requeridos y dar clic en el botón “guardar”</td> <td>2.a.3. El sistema vuelve a la vista de máquinas físicas. 4.a.1. El sistema no encuentra la máquina. 4.a.2. El sistema muestra mensaje informativo “Sin resultados”</td> </tr> </tbody> </table>		Actor	Sistema	2.a.1. Si es una máquina de almacenamiento debe dar clic en la opción “almacenamiento”. 2.a.2. El usuario debe rellenar todos los campos requeridos y dar clic en el botón “guardar”	2.a.3. El sistema vuelve a la vista de máquinas físicas. 4.a.1. El sistema no encuentra la máquina. 4.a.2. El sistema muestra mensaje informativo “Sin resultados”		
Actor	Sistema							
2.a.1. Si es una máquina de almacenamiento debe dar clic en la opción “almacenamiento”. 2.a.2. El usuario debe rellenar todos los campos requeridos y dar clic en el botón “guardar”	2.a.3. El sistema vuelve a la vista de máquinas físicas. 4.a.1. El sistema no encuentra la máquina. 4.a.2. El sistema muestra mensaje informativo “Sin resultados”							
Postcondición	La máquina se encuentra dada de alta en el sistema							

Tabla 47 - CU-3: Alta máquina

Id	CU-4								
Nombre	Baja máquina								
Actores	Genérico y Técnico								
Resumen	El usuario da de baja a una máquina en el sistema								
Precondición									
Flujo normal	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>1. El usuario busca la máquina que desea dar de baja en el módulo “maquinas”.</td> <td>3. El sistema muestra un mensaje de confirmación “Seguro desea eliminar la máquina?”</td> </tr> <tr> <td>2. El usuario debe hacer clic en el botón “eliminar máquina”.</td> <td>5. El sistema elimina la máquina y todas sus dependencias de la base de datos.</td> </tr> <tr> <td>4. El usuario hace clic en el botón “SI”</td> <td></td> </tr> </tbody> </table>	Actor	Sistema	1. El usuario busca la máquina que desea dar de baja en el módulo “maquinas”.	3. El sistema muestra un mensaje de confirmación “Seguro desea eliminar la máquina?”	2. El usuario debe hacer clic en el botón “eliminar máquina”.	5. El sistema elimina la máquina y todas sus dependencias de la base de datos.	4. El usuario hace clic en el botón “SI”	
Actor	Sistema								
1. El usuario busca la máquina que desea dar de baja en el módulo “maquinas”.	3. El sistema muestra un mensaje de confirmación “Seguro desea eliminar la máquina?”								
2. El usuario debe hacer clic en el botón “eliminar máquina”.	5. El sistema elimina la máquina y todas sus dependencias de la base de datos.								
4. El usuario hace clic en el botón “SI”									
Flujo alternativo	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>4.a.1. El usuario hace clic en el botón “NO”</td> <td>4.a.2. El sistema no elimina la máquina</td> </tr> </tbody> </table>	Actor	Sistema	4.a.1. El usuario hace clic en el botón “NO”	4.a.2. El sistema no elimina la máquina				
Actor	Sistema								
4.a.1. El usuario hace clic en el botón “NO”	4.a.2. El sistema no elimina la máquina								
Postcondición	La máquina ya no se encuentra en el sistema								

Tabla 48 - CU-4: Baja máquina

Id	CU-5								
Nombre	Editar máquina								
Actores	Genérico y Técnico								
Resumen	El usuario edita la información de una máquina								
Precondición									
Flujo normal	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>1. El usuario busca la máquina que desea editar en el módulo “Maquinas”.</td> <td>3. El sistema muestra la información de la máquina.</td> </tr> <tr> <td>2. El usuario debe hacer clic en el botón “editar máquina”.</td> <td>5. El sistema actualiza la información de la máquina en la base de datos.</td> </tr> <tr> <td>4. El usuario realiza los cambios que desee y hace clic en el botón “Guardar”</td> <td></td> </tr> </tbody> </table>	Actor	Sistema	1. El usuario busca la máquina que desea editar en el módulo “Maquinas”.	3. El sistema muestra la información de la máquina.	2. El usuario debe hacer clic en el botón “editar máquina”.	5. El sistema actualiza la información de la máquina en la base de datos.	4. El usuario realiza los cambios que desee y hace clic en el botón “Guardar”	
Actor	Sistema								
1. El usuario busca la máquina que desea editar en el módulo “Maquinas”.	3. El sistema muestra la información de la máquina.								
2. El usuario debe hacer clic en el botón “editar máquina”.	5. El sistema actualiza la información de la máquina en la base de datos.								
4. El usuario realiza los cambios que desee y hace clic en el botón “Guardar”									
Flujo alternativo	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> </tr> </tbody> </table>	Actor	Sistema						
Actor	Sistema								
Postcondición	La información de la máquina ha cambiado.								

Tabla 49 - CU-5: Editar máquina

Id	CU-6											
Nombre	Alta Pool											
Actores	Genérico y Técnico											
Resumen	El usuario da de alta a un Pool en el sistema											
Precondición	<ul style="list-style-type: none"> - El dispositivo SAI debe estar dado de alta en base de datos - El Pool debe existir en el administrador de pools XenCenter - El Pool debe estar encendido. 											
Flujo normal	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>1. El usuario ingresa al módulo “alta pool”.</td> <td>2. El sistema muestra el formulario con los datos requeridos.</td> </tr> <tr> <td>3. El usuario ingresa los datos del Pool y hace clic en el botón “Guardar”.</td> <td>4. El sistema comprueba que se pueda conectar al Pool.</td> </tr> <tr> <td></td> <td>5. El sistema da de alta al Pool en la base de datos.</td> </tr> <tr> <td></td> <td>6. El sistema vuelve a la vista de máquinas físicas.</td> </tr> </tbody> </table>	Actor	Sistema	1. El usuario ingresa al módulo “alta pool”.	2. El sistema muestra el formulario con los datos requeridos.	3. El usuario ingresa los datos del Pool y hace clic en el botón “Guardar”.	4. El sistema comprueba que se pueda conectar al Pool.		5. El sistema da de alta al Pool en la base de datos.		6. El sistema vuelve a la vista de máquinas físicas.	
Actor	Sistema											
1. El usuario ingresa al módulo “alta pool”.	2. El sistema muestra el formulario con los datos requeridos.											
3. El usuario ingresa los datos del Pool y hace clic en el botón “Guardar”.	4. El sistema comprueba que se pueda conectar al Pool.											
	5. El sistema da de alta al Pool en la base de datos.											
	6. El sistema vuelve a la vista de máquinas físicas.											
Flujo alternativo	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td></td> <td>4.a.1. El sistema no tiene conexión con el Pool.</td> </tr> <tr> <td></td> <td>4.a.2. El sistema muestra mensaje de error “No se ha podido establecer conexión con el Pool, compruebe los datos”.</td> </tr> <tr> <td></td> <td>4.a.3. El sistema no da de alta al Pool.</td> </tr> </tbody> </table>	Actor	Sistema		4.a.1. El sistema no tiene conexión con el Pool.		4.a.2. El sistema muestra mensaje de error “No se ha podido establecer conexión con el Pool, compruebe los datos”.		4.a.3. El sistema no da de alta al Pool.			
Actor	Sistema											
	4.a.1. El sistema no tiene conexión con el Pool.											
	4.a.2. El sistema muestra mensaje de error “No se ha podido establecer conexión con el Pool, compruebe los datos”.											
	4.a.3. El sistema no da de alta al Pool.											
Postcondición	El Pool se encuentra dado de alta en el sistema											

Tabla 50 - CU-6: Alta pool

Id	CU-7									
Nombre	Editar Pool									
Actores	Genérico y Técnico									
Resumen	El usuario edita solo la información permitida de un Pool									
Precondición	<ul style="list-style-type: none"> - El Pool no dejara editar sus credenciales de conexión 									
Flujo normal	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>1. El usuario busca el Pool que desea editar en el módulo “Pools”.</td> <td>3. El sistema muestra la información del Pool.</td> </tr> <tr> <td>2. El usuario debe hacer clic en el botón “editar pool”.</td> <td>5. El sistema actualiza la información del Pool en la base de datos.</td> </tr> <tr> <td>4. El usuario realiza los cambios permitidos y hace clic en el botón “Guardar”</td> <td></td> </tr> </tbody> </table>	Actor	Sistema	1. El usuario busca el Pool que desea editar en el módulo “Pools”.	3. El sistema muestra la información del Pool.	2. El usuario debe hacer clic en el botón “editar pool”.	5. El sistema actualiza la información del Pool en la base de datos.	4. El usuario realiza los cambios permitidos y hace clic en el botón “Guardar”		
Actor	Sistema									
1. El usuario busca el Pool que desea editar en el módulo “Pools”.	3. El sistema muestra la información del Pool.									
2. El usuario debe hacer clic en el botón “editar pool”.	5. El sistema actualiza la información del Pool en la base de datos.									
4. El usuario realiza los cambios permitidos y hace clic en el botón “Guardar”										
Flujo alternativo	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> </tr> </tbody> </table>	Actor	Sistema							
Actor	Sistema									
Postcondición	La información del Pool ha cambiado.									

Tabla 51 - CU-7: Editar pool

<i>Id</i>	CU-8									
<i>Nombre</i>	Eliminar Pool									
<i>Actores</i>	Genérico y Técnico									
<i>Resumen</i>	El usuario da de baja a un Pool en el sistema									
<i>Precondición</i>										
<i>Flujo normal</i>	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>1. El usuario busca el Pool que desea dar de baja en el módulo "POOLS".</td> <td>3. El sistema muestra un mensaje de confirmación "Seguro desea eliminar el Pool?"</td> </tr> <tr> <td>2. El usuario debe hacer clic en el botón "eliminar pool".</td> <td>5. El sistema elimina el Pool.</td> </tr> <tr> <td>4. El usuario hace clic en el botón "SI"</td> <td>6. El sistema elimina todas las máquinas virtuales que pertenecían al Pool. 7. El sistema elimina todas las dependencias que tenían las máquinas virtuales.</td> </tr> </tbody> </table>	Actor	Sistema	1. El usuario busca el Pool que desea dar de baja en el módulo "POOLS".	3. El sistema muestra un mensaje de confirmación "Seguro desea eliminar el Pool?"	2. El usuario debe hacer clic en el botón "eliminar pool".	5. El sistema elimina el Pool.	4. El usuario hace clic en el botón "SI"	6. El sistema elimina todas las máquinas virtuales que pertenecían al Pool. 7. El sistema elimina todas las dependencias que tenían las máquinas virtuales.	
Actor	Sistema									
1. El usuario busca el Pool que desea dar de baja en el módulo "POOLS".	3. El sistema muestra un mensaje de confirmación "Seguro desea eliminar el Pool?"									
2. El usuario debe hacer clic en el botón "eliminar pool".	5. El sistema elimina el Pool.									
4. El usuario hace clic en el botón "SI"	6. El sistema elimina todas las máquinas virtuales que pertenecían al Pool. 7. El sistema elimina todas las dependencias que tenían las máquinas virtuales.									
<i>Flujo alternativo</i>	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>4.a.1. El usuario hace clic en el botón "NO"</td> <td>4.a.2. El sistema no elimina el Pool</td> </tr> </tbody> </table>	Actor	Sistema	4.a.1. El usuario hace clic en el botón "NO"	4.a.2. El sistema no elimina el Pool					
Actor	Sistema									
4.a.1. El usuario hace clic en el botón "NO"	4.a.2. El sistema no elimina el Pool									
<i>Postcondición</i>	El Pool ya no se encuentra en el sistema									

Tabla 52 - CU-8: Baja pool

<i>Id</i>	CU-9					
<i>Nombre</i>	Alta Dependencia					
<i>Actores</i>	Genérico y Técnico					
<i>Resumen</i>	El usuario da de alta a una dependencia entre dos máquinas					
<i>Precondición</i>	<ul style="list-style-type: none"> - Si es una máquina virtual, el Pool debe estar dado de alta en la base de datos. - Si es una máquina física, esta debe estar dada de alta en la base de datos. 					
<i>Flujo normal</i>	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> </tr> </tbody> </table>	Actor	Sistema			
Actor	Sistema					

Flujo alternativo	<ol style="list-style-type: none"> 1. El usuario ingresa al módulo “alta dependencias”. 3. El usuario selecciona que tipo de máquina desea como Padre. (Virtual-Física) 4. Elegida la opción, selecciona que máquina será el Padre en la dependencia. 5. El usuario selecciona que tipo de máquina desea como Hijo. (Virtual-Física) 6. Elegida la opción, selecciona que máquina será el Hijo en la dependencia. 8. El usuario hace clic en el botón “Alta”. 	<ol style="list-style-type: none"> 2. El sistema muestra una vista para seleccionar a la izquierda la máquina Padre y a la derecha la máquina Hijo. 7. El Sistema muestra un árbol de dependencias, indicando las máquinas que dependen de ese Padre. 4. El sistema guarda la dependencia en base de datos.
	Actor	Sistema
	Postcondición La dependencia se encuentra dada de alta en el sistema	

Tabla 53 - CU-9: Alta dependencia

Id	CU-10								
Nombre	Eliminar Dependencia								
Actores	Genérico y Técnico								
Resumen	El usuario da de baja a una dependencia entre dos máquinas								
Precondición									
Flujo normal									
Flujo alternativo	<table border="1" style="width: 100%;"> <thead> <tr> <th style="text-align: center;">Actor</th> <th style="text-align: center;">Sistema</th> </tr> </thead> <tbody> <tr> <td> <ol style="list-style-type: none"> 1. El usuario ingresa al módulo “alta dependencias”. 3. El usuario selecciona que tipo de máquina desea buscar como Padre de la dependencia. (Virtual-Física) 4. Elegida la opción, selecciona que máquina es el Padre de la dependencia que desea eliminar. 6. El usuario elige la máquina que desea quitarle la dependencia, y hace clic en el botón “Borrar”. </td> <td> <ol style="list-style-type: none"> 2. El sistema muestra una vista para seleccionar a la izquierda la máquina Padre y a la derecha la máquina Hijo. 5. El Sistema muestra todas las máquinas que dependen de ese Padre. 7. El sistema elimina la dependencia en base de datos. 8. El sistema comprueba si la dependencia que elimino pertenece a una máquina virtual. Si lo es, elimina la máquina virtual de la base de datos. </td> </tr> <tr> <td style="text-align: center;">Actor</td> <td style="text-align: center;">Sistema</td> </tr> <tr> <td colspan="2">Postcondición La dependencia ya no existe en el sistema.</td> </tr> </tbody> </table> <p style="text-align: center;"><i>Tabla 54 - CU-10: Eliminar dependencia</i></p>	Actor	Sistema	<ol style="list-style-type: none"> 1. El usuario ingresa al módulo “alta dependencias”. 3. El usuario selecciona que tipo de máquina desea buscar como Padre de la dependencia. (Virtual-Física) 4. Elegida la opción, selecciona que máquina es el Padre de la dependencia que desea eliminar. 6. El usuario elige la máquina que desea quitarle la dependencia, y hace clic en el botón “Borrar”. 	<ol style="list-style-type: none"> 2. El sistema muestra una vista para seleccionar a la izquierda la máquina Padre y a la derecha la máquina Hijo. 5. El Sistema muestra todas las máquinas que dependen de ese Padre. 7. El sistema elimina la dependencia en base de datos. 8. El sistema comprueba si la dependencia que elimino pertenece a una máquina virtual. Si lo es, elimina la máquina virtual de la base de datos. 	Actor	Sistema	Postcondición La dependencia ya no existe en el sistema.	
	Actor	Sistema							
	<ol style="list-style-type: none"> 1. El usuario ingresa al módulo “alta dependencias”. 3. El usuario selecciona que tipo de máquina desea buscar como Padre de la dependencia. (Virtual-Física) 4. Elegida la opción, selecciona que máquina es el Padre de la dependencia que desea eliminar. 6. El usuario elige la máquina que desea quitarle la dependencia, y hace clic en el botón “Borrar”. 	<ol style="list-style-type: none"> 2. El sistema muestra una vista para seleccionar a la izquierda la máquina Padre y a la derecha la máquina Hijo. 5. El Sistema muestra todas las máquinas que dependen de ese Padre. 7. El sistema elimina la dependencia en base de datos. 8. El sistema comprueba si la dependencia que elimino pertenece a una máquina virtual. Si lo es, elimina la máquina virtual de la base de datos. 							
Actor	Sistema								
Postcondición La dependencia ya no existe en el sistema.									

Id	CU-11												
Nombre	Descargar archivo de clave publica												
Actores	Genérico y Técnico												
Resumen	El usuario puede ver las instrucciones para descargar y ubicar el archivo de clave publica												
Precondición													
Flujo normal	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>1. El usuario ingresa al módulo “Instrucciones”.</td> <td>2. El sistema muestra un pequeño manual para la integración de una máquina al sistema de apagado.</td> </tr> <tr> <td>2. El usuario lee las instrucciones sobre su sistema operativo.</td> <td>4. El sistema descarga el archivo de clave publica en la máquina del usuario.</td> </tr> <tr> <td>3. El usuario hace clic en el botón “Descargar archivo”</td> <td></td> </tr> <tr> <th>Actor</th> <th>Sistema</th> </tr> <tr> <td></td> <td></td> </tr> </tbody> </table>	Actor	Sistema	1. El usuario ingresa al módulo “Instrucciones”.	2. El sistema muestra un pequeño manual para la integración de una máquina al sistema de apagado.	2. El usuario lee las instrucciones sobre su sistema operativo.	4. El sistema descarga el archivo de clave publica en la máquina del usuario.	3. El usuario hace clic en el botón “Descargar archivo”		Actor	Sistema		
Actor	Sistema												
1. El usuario ingresa al módulo “Instrucciones”.	2. El sistema muestra un pequeño manual para la integración de una máquina al sistema de apagado.												
2. El usuario lee las instrucciones sobre su sistema operativo.	4. El sistema descarga el archivo de clave publica en la máquina del usuario.												
3. El usuario hace clic en el botón “Descargar archivo”													
Actor	Sistema												
Flujo alternativo													
Postcondición	El usuario tiene la clave pública del servidor en su ordenador.												

Tabla 55 - CU-11: Descargar archivo de clave publica

Id	CU-12										
Nombre	Ver baterías SAI										
Actores	Técnico										
Resumen	El usuario técnico puede ver el nivel y tiempo restante de las baterías de todos los SAIS dados de alta en el sistema.										
Precondición											
Flujo normal	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>1. El usuario ingresa al módulo “Batería”.</td> <td>2. El sistema interroga a los SAIS preguntando por el nivel de batería.</td> </tr> <tr> <td></td> <td>3. El sistema muestra un diagrama de tipo columna para representar la batería de todos los SAIS.</td> </tr> <tr> <th>Actor</th> <th>Sistema</th> </tr> <tr> <td></td> <td></td> </tr> </tbody> </table>	Actor	Sistema	1. El usuario ingresa al módulo “Batería”.	2. El sistema interroga a los SAIS preguntando por el nivel de batería.		3. El sistema muestra un diagrama de tipo columna para representar la batería de todos los SAIS.	Actor	Sistema		
Actor	Sistema										
1. El usuario ingresa al módulo “Batería”.	2. El sistema interroga a los SAIS preguntando por el nivel de batería.										
	3. El sistema muestra un diagrama de tipo columna para representar la batería de todos los SAIS.										
Actor	Sistema										
Flujo alternativo											
Postcondición	El usuario esta informado del nivel de batería de todos los SAIS										

Tabla 56 - CU-12: Ver baterías SAI

Id	CU-13					
Nombre	Alta Sai					
Actores	Técnico					
Resumen	El usuario técnico da de alta a un SAI en el sistema					
Precondición						
Flujo normal	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>1. El usuario ingresa al módulo “SAIS”. 3. El usuario ingresa los datos del Sai y hace clic en el botón “Guardar”.</td> <td>2. El sistema muestra el formulario con los datos requeridos. 4. El sistema comprueba que se pueda conectar al Sai. 5. El sistema da de alta al Sai en la base de datos. 6. El sistema vuelve a la vista de SAIS.</td> </tr> </tbody> </table>		Actor	Sistema	1. El usuario ingresa al módulo “SAIS”. 3. El usuario ingresa los datos del Sai y hace clic en el botón “Guardar”.	2. El sistema muestra el formulario con los datos requeridos. 4. El sistema comprueba que se pueda conectar al Sai. 5. El sistema da de alta al Sai en la base de datos. 6. El sistema vuelve a la vista de SAIS.
Actor	Sistema					
1. El usuario ingresa al módulo “SAIS”. 3. El usuario ingresa los datos del Sai y hace clic en el botón “Guardar”.	2. El sistema muestra el formulario con los datos requeridos. 4. El sistema comprueba que se pueda conectar al Sai. 5. El sistema da de alta al Sai en la base de datos. 6. El sistema vuelve a la vista de SAIS.					
Flujo alternativo	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td></td> <td>4.a.1. El sistema no tiene conexión con el Sai. 4.a.2. El sistema muestra mensaje de error “No se ha podido establecer conexión con el Sai, compruebe los datos”. 4.a.3. El sistema no da de alta al Sai.</td> </tr> </tbody> </table>		Actor	Sistema		4.a.1. El sistema no tiene conexión con el Sai. 4.a.2. El sistema muestra mensaje de error “No se ha podido establecer conexión con el Sai, compruebe los datos”. 4.a.3. El sistema no da de alta al Sai.
Actor	Sistema					
	4.a.1. El sistema no tiene conexión con el Sai. 4.a.2. El sistema muestra mensaje de error “No se ha podido establecer conexión con el Sai, compruebe los datos”. 4.a.3. El sistema no da de alta al Sai.					
Postcondición	El Sai se encuentra dado de alta en el sistema					

Tabla 57 - CU-13: Alta Sai

Id	CU-14					
Nombre	Editar Sai					
Actores	Técnico					
Resumen	El usuario edita solo la información permitida de un Sai					
Precondición	- El Sai no dejara editar sus credenciales de conexión					
Flujo normal	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>1. El usuario busca el Sai que desea editar en el módulo “SAIS”. 2. El usuario debe hacer clic en el botón “editar Sai”. 4. El usuario realiza los cambios permitidos y hace clic en el botón “Guardar”</td> <td>3. El sistema muestra la información del Sai. 5. El sistema actualiza la información del Sai en la base de datos.</td> </tr> </tbody> </table>		Actor	Sistema	1. El usuario busca el Sai que desea editar en el módulo “SAIS”. 2. El usuario debe hacer clic en el botón “editar Sai”. 4. El usuario realiza los cambios permitidos y hace clic en el botón “Guardar”	3. El sistema muestra la información del Sai. 5. El sistema actualiza la información del Sai en la base de datos.
Actor	Sistema					
1. El usuario busca el Sai que desea editar en el módulo “SAIS”. 2. El usuario debe hacer clic en el botón “editar Sai”. 4. El usuario realiza los cambios permitidos y hace clic en el botón “Guardar”	3. El sistema muestra la información del Sai. 5. El sistema actualiza la información del Sai en la base de datos.					
Flujo alternativo	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> </tr> </tbody> </table>		Actor	Sistema		
Actor	Sistema					
Postcondición	La información del Sai ha cambiado.					

Tabla 58 - CU-14: Editar Sai

<i>Id</i>	CU-15								
<i>Nombre</i>	Eliminar Sai								
<i>Actores</i>	Técnico								
<i>Resumen</i>	El usuario da de baja a un Sai en el sistema								
<i>Precondición</i>									
<i>Flujo normal</i>	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>1. El usuario busca el Sai que desea dar de baja en el módulo "SAIS". 2. El usuario debe hacer clic en el botón "eliminar Sai". 4. El usuario hace clic en el botón "SI"</td> <td>3. El sistema muestra un mensaje de confirmación "Seguro desea eliminar el Sai?" 5. El sistema elimina el Sai.</td> </tr> <tr> <th>Actor</th> <th>Sistema</th> </tr> <tr> <td>4.a.1. El usuario hace clic en el botón "NO"</td> <td>4.a.2. El sistema no elimina el Sai</td> </tr> </tbody> </table>	Actor	Sistema	1. El usuario busca el Sai que desea dar de baja en el módulo "SAIS". 2. El usuario debe hacer clic en el botón "eliminar Sai". 4. El usuario hace clic en el botón "SI"	3. El sistema muestra un mensaje de confirmación "Seguro desea eliminar el Sai?" 5. El sistema elimina el Sai.	Actor	Sistema	4.a.1. El usuario hace clic en el botón "NO"	4.a.2. El sistema no elimina el Sai
Actor	Sistema								
1. El usuario busca el Sai que desea dar de baja en el módulo "SAIS". 2. El usuario debe hacer clic en el botón "eliminar Sai". 4. El usuario hace clic en el botón "SI"	3. El sistema muestra un mensaje de confirmación "Seguro desea eliminar el Sai?" 5. El sistema elimina el Sai.								
Actor	Sistema								
4.a.1. El usuario hace clic en el botón "NO"	4.a.2. El sistema no elimina el Sai								
<i>Flujo alternativo</i>									
<i>Postcondición</i>	El Sai ya no se encuentra en el sistema								

Tabla 59 - CU-15: Eliminar Sai

<i>Id</i>	CU-16								
<i>Nombre</i>	Alta Usuario								
<i>Actores</i>	Técnico								
<i>Resumen</i>	El usuario técnico da de alta a un Usuario en el sistema								
<i>Precondición</i>									
<i>Flujo normal</i>	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>1. El usuario técnico ingresa al módulo "USUARIOS". 3. El usuario hace clic en el botón "Alta usuario". 5. El usuario ingresa los datos del nuevo Usuario genérico y hace clic en el botón "Guardar".</td> <td>2. El sistema muestra el listado de todos los usuarios existentes. 4. El sistema muestra el formulario con los datos requeridos. 6. El sistema da de alta al nuevo Usuario en la base de datos.</td> </tr> <tr> <th>Actor</th> <th>Sistema</th> </tr> <tr> <td>5.a.1. El usuario marca la casilla "Técnico" para crear un usuario de ese rol. 5.a.2. El usuario hace clic en el botón "Guardar"</td> <td></td> </tr> </tbody> </table>	Actor	Sistema	1. El usuario técnico ingresa al módulo "USUARIOS". 3. El usuario hace clic en el botón "Alta usuario". 5. El usuario ingresa los datos del nuevo Usuario genérico y hace clic en el botón "Guardar".	2. El sistema muestra el listado de todos los usuarios existentes. 4. El sistema muestra el formulario con los datos requeridos. 6. El sistema da de alta al nuevo Usuario en la base de datos.	Actor	Sistema	5.a.1. El usuario marca la casilla "Técnico" para crear un usuario de ese rol. 5.a.2. El usuario hace clic en el botón "Guardar"	
Actor	Sistema								
1. El usuario técnico ingresa al módulo "USUARIOS". 3. El usuario hace clic en el botón "Alta usuario". 5. El usuario ingresa los datos del nuevo Usuario genérico y hace clic en el botón "Guardar".	2. El sistema muestra el listado de todos los usuarios existentes. 4. El sistema muestra el formulario con los datos requeridos. 6. El sistema da de alta al nuevo Usuario en la base de datos.								
Actor	Sistema								
5.a.1. El usuario marca la casilla "Técnico" para crear un usuario de ese rol. 5.a.2. El usuario hace clic en el botón "Guardar"									
<i>Flujo alternativo</i>									
<i>Postcondición</i>	El nuevo Usuario se encuentra dado de alta en el sistema.								

Tabla 60 - CU-16: Alta usuario

Id	CU-17								
Nombre	Editar Usuario								
Actores	Técnico								
Resumen	El usuario edita solo la información permitida de un Usuario								
Precondición	- El sistema no dejara editar el campo "Login"								
Flujo normal	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>1. El usuario técnico busca el Usuario que desea editar en el módulo "USUARIOS". 2. El usuario técnico debe hacer clic en el botón "editar Usuario". 4. El usuario realiza los cambios permitidos y hace clic en el botón "Guardar"</td> <td>3. El sistema muestra la información del Usuario. 5. El sistema actualiza la información del Usuario en la base de datos.</td> </tr> <tr> <td>Actor</td> <td>Sistema</td> </tr> <tr> <td></td> <td></td> </tr> </tbody> </table>	Actor	Sistema	1. El usuario técnico busca el Usuario que desea editar en el módulo "USUARIOS". 2. El usuario técnico debe hacer clic en el botón "editar Usuario". 4. El usuario realiza los cambios permitidos y hace clic en el botón "Guardar"	3. El sistema muestra la información del Usuario. 5. El sistema actualiza la información del Usuario en la base de datos.	Actor	Sistema		
Actor	Sistema								
1. El usuario técnico busca el Usuario que desea editar en el módulo "USUARIOS". 2. El usuario técnico debe hacer clic en el botón "editar Usuario". 4. El usuario realiza los cambios permitidos y hace clic en el botón "Guardar"	3. El sistema muestra la información del Usuario. 5. El sistema actualiza la información del Usuario en la base de datos.								
Actor	Sistema								
Flujo alternativo									
Postcondición	La información del Usuario ha cambiado.								

Tabla 61 - CU-17: Editar usuario

Id	CU-18								
Nombre	Cambiar contraseña								
Actores	Técnico								
Resumen	El usuario técnico cambia la contraseña de un Usuario								
Precondición	- El sistema no dejara editar el campo "Login"								
Flujo normal	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>1. El usuario técnico debe hacer clic en el botón "editar Usuario". 3. El usuario técnico cambia la contraseña antigua del usuario por una contraseña nueva. 4. El usuario técnico hace clic en el botón "Guardar"</td> <td>2. El sistema muestra la información del Usuario. 5. El sistema actualiza la contraseña del Usuario en la base de datos.</td> </tr> <tr> <td>Actor</td> <td>Sistema</td> </tr> <tr> <td></td> <td></td> </tr> </tbody> </table>	Actor	Sistema	1. El usuario técnico debe hacer clic en el botón "editar Usuario". 3. El usuario técnico cambia la contraseña antigua del usuario por una contraseña nueva. 4. El usuario técnico hace clic en el botón "Guardar"	2. El sistema muestra la información del Usuario. 5. El sistema actualiza la contraseña del Usuario en la base de datos.	Actor	Sistema		
Actor	Sistema								
1. El usuario técnico debe hacer clic en el botón "editar Usuario". 3. El usuario técnico cambia la contraseña antigua del usuario por una contraseña nueva. 4. El usuario técnico hace clic en el botón "Guardar"	2. El sistema muestra la información del Usuario. 5. El sistema actualiza la contraseña del Usuario en la base de datos.								
Actor	Sistema								
Flujo alternativo									
Postcondición	La contraseña del Usuario se ha actualizado.								

Tabla 62 - CU-18: Cambiar contraseña

<i>Id</i>	CU-19									
<i>Nombre</i>	Eliminar Usuario									
<i>Actores</i>	Técnico									
<i>Resumen</i>	El usuario técnico da de baja a un Usuario del sistema									
<i>Precondición</i>										
<i>Flujo normal</i>	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>1. El usuario técnico busca el Usuario que desea dar de baja en el módulo “USUARIOS”.</td> <td>3. El sistema muestra un mensaje de confirmación “Seguro desea eliminar el Usuario?”</td> </tr> <tr> <td>2. El usuario técnico debe hacer clic en el botón “eliminar Usuario”.</td> <td>5. El sistema elimina el Usuario.</td> </tr> <tr> <td>4. El usuario técnico hace clic en el botón “SI”</td> <td></td> </tr> </tbody> </table>	Actor	Sistema	1. El usuario técnico busca el Usuario que desea dar de baja en el módulo “USUARIOS”.	3. El sistema muestra un mensaje de confirmación “Seguro desea eliminar el Usuario?”	2. El usuario técnico debe hacer clic en el botón “eliminar Usuario”.	5. El sistema elimina el Usuario.	4. El usuario técnico hace clic en el botón “SI”		
Actor	Sistema									
1. El usuario técnico busca el Usuario que desea dar de baja en el módulo “USUARIOS”.	3. El sistema muestra un mensaje de confirmación “Seguro desea eliminar el Usuario?”									
2. El usuario técnico debe hacer clic en el botón “eliminar Usuario”.	5. El sistema elimina el Usuario.									
4. El usuario técnico hace clic en el botón “SI”										
<i>Flujo alternativo</i>	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>4.a.1. El usuario técnico hace clic en el botón “NO”</td> <td>4.a.2. El sistema no elimina al Usuario.</td> </tr> </tbody> </table>	Actor	Sistema	4.a.1. El usuario técnico hace clic en el botón “NO”	4.a.2. El sistema no elimina al Usuario.					
Actor	Sistema									
4.a.1. El usuario técnico hace clic en el botón “NO”	4.a.2. El sistema no elimina al Usuario.									
<i>Postcondición</i>	El Usuario ya no se encuentra en el sistema									

Tabla 63 - CU-19: Eliminar usuario

<i>Id</i>	CU-20											
<i>Nombre</i>	Alta máquina virtual											
<i>Actores</i>	Genérico y Técnico											
<i>Resumen</i>	El usuario da de alta a una máquina virtual creando una dependencia entre dos máquinas.											
<i>Precondición</i>	- El Pool donde se aloja la máquina virtual debe estar dado de alta en la base de datos.											
<i>Flujo normal</i>	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>1. El usuario ingresa al módulo “alta dependencias”.</td> <td>2. El sistema muestra una vista para seleccionar a la izquierda la máquina Padre y a la derecha la máquina Hijo.</td> </tr> <tr> <td>3. El usuario selecciona el Pool donde se aloja la máquina virtual.</td> <td>6. El Sistema da de alta a la máquina virtual en base de datos.</td> </tr> <tr> <td>4. Seleccionado el Pool, elige la máquina virtual que será el Padre o Hijo en la dependencia.</td> <td>4. El sistema da de alta a la dependencia en base de datos.</td> </tr> <tr> <td>5. El usuario hace clic en el botón “Alta”.</td> <td></td> </tr> </tbody> </table>	Actor	Sistema	1. El usuario ingresa al módulo “alta dependencias”.	2. El sistema muestra una vista para seleccionar a la izquierda la máquina Padre y a la derecha la máquina Hijo.	3. El usuario selecciona el Pool donde se aloja la máquina virtual.	6. El Sistema da de alta a la máquina virtual en base de datos.	4. Seleccionado el Pool, elige la máquina virtual que será el Padre o Hijo en la dependencia.	4. El sistema da de alta a la dependencia en base de datos.	5. El usuario hace clic en el botón “Alta”.		
Actor	Sistema											
1. El usuario ingresa al módulo “alta dependencias”.	2. El sistema muestra una vista para seleccionar a la izquierda la máquina Padre y a la derecha la máquina Hijo.											
3. El usuario selecciona el Pool donde se aloja la máquina virtual.	6. El Sistema da de alta a la máquina virtual en base de datos.											
4. Seleccionado el Pool, elige la máquina virtual que será el Padre o Hijo en la dependencia.	4. El sistema da de alta a la dependencia en base de datos.											
5. El usuario hace clic en el botón “Alta”.												
<i>Flujo alternativo</i>	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> </tr> </tbody> </table>	Actor	Sistema									
Actor	Sistema											
<i>Postcondición</i>	La máquina virtual se encuentra dada de alta en el sistema											

Tabla 64 - CU-20: Alta máquina virtual

Anexo B: Manual de usuario Ges-SAI

HOJA DE CONTROL

Software	Ges-SAI		
Entregable	Manual de Usuario		
Autor	Alejandro Junior Caicedo Palacios		
Versión/Edición	01	Fecha Versión	01/08/2022

REGISTRO DE CAMBIOS

Versión	Causa del Cambio	Responsable del Cambio	Fecha del Cambio
01	Versión inicial	Alejandro Junior Caicedo Palacios	01/08/2022

Tabla de Contenido

1. DESCRIPCIÓN DEL SISTEMA	116
1.1. Objetivos del manual de usuario	116
1.2. Funcionalidades	116
2. MÓDULOS	117
2.1. Login	117
2.2. Sais	118
2.3. Pools	121
2.4. Máquinas	124
2.4.1. Alta máquina física / almacenamiento	124
2.4.2. Alta dependencias	128
2.5. Usuarios	133
2.6. Baterías	136
2.7. Instrucciones	137

1. DESCRIPCIÓN DEL SISTEMA

Ges-SAI es un sistema desarrollado para el DSIC, y su funcionalidad principal es poder prevenir un apagado forzado a las máquinas físicas y máquinas virtuales que se encuentren conectadas a los dispositivos SAI del departamento. Ges-SAI gestiona la automatización del apagado y encendido de las máquinas físicas y virtuales, mostrando en tiempo real el estado de los dispositivos SAI e indicando el nivel de sus baterías y el tiempo de duración de estas. Aparte de poder administrar los SAIs y Pools, la aplicación también le permite al usuario administrar las diferentes dependencias entre las máquinas que desee, con el fin de incluirlas en la automatización del encendido y apagado.

1.1. Objetivos del manual de usuario

Los objetivos que se quieren alcanzar en este manual son los siguientes:

- Servir como manual de referencia y aprendizaje de la aplicación
- Delimitar las funciones que debe realizar cada rol de usuario
- Capacitar al usuario para que conozca como preparar los datos de entrada
- Capacitar al usuario para que aprenda a obtener los resultados de salida
- Definir los módulos en que cada usuario participara

1.2. Funcionalidades

Ges-SAI cuenta con 2 roles de usuario, rol técnico y rol genérico y está conformado por 6 módulos generales que lo componen, a continuación, se describe cada uno de estos con su respectivo rol de usuario el cual tiene permisos sobre el módulo.

MODULO	DESCRIPCIÓN	ROL DE USUARIO
MÁQUINAS	En este módulo el usuario puede administrar todas sus máquinas dadas de alta en el sistema, y puede crear dependencias entre las diferentes máquinas.	<ul style="list-style-type: none"> • Técnico • Genérico
POOLS	En este módulo el usuario puede administrar todos sus Pools dados de alta en el sistema.	<ul style="list-style-type: none"> • Técnico • Genérico
SAIS	En este módulo el usuario puede administrar todos los SAIS dados de alta en el sistema.	<ul style="list-style-type: none"> • Técnico
BATERÍAS	En este módulo el usuario puede visualizar el nivel de baterías de cada uno de los dispositivos SAI dados de alta en el sistema	<ul style="list-style-type: none"> • Técnico
USUARIO	En este módulo el usuario puede administrar todos los usuarios dados de alta en el sistema	<ul style="list-style-type: none"> • Técnico

INSTRUCCIONES	En este módulo el usuario puede descargar el archivo de clave publica y seguir las instrucciones para su instalación	<ul style="list-style-type: none">• Técnico• Genérico
----------------------	----------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------

2. MÓDULOS

A continuación, se describen los módulos que componen al sistema de Ges-SAI, en cada uno de los siguientes apartados encontrara imágenes de las diferentes interfaz graficas que tiene el sistema para cumplir con las funcionalidades.

2.1. Login

Para poder acceder a Ges-SAI, ingrese usuario y contraseña, si no cuenta con un usuario dado de alta en el sistema, comuníquelo con los administradores, ellos le asignaran uno. Una vez ingrese las credenciales de autenticación, de clic en el botón “**Iniciar sesión**”

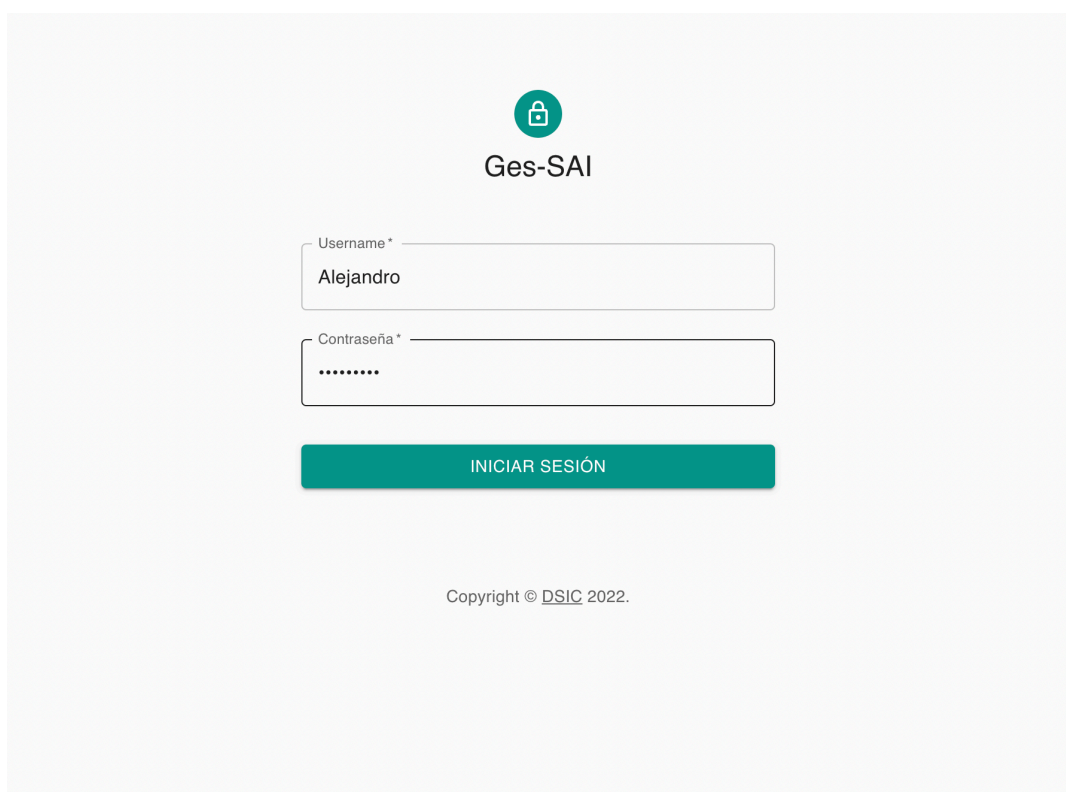


Imagen 1 - Login

Si las credenciales de autenticación son válidas, el sistema lo dejara acceder a la pantalla principal, si por el contrario son incorrectas, el sistema le mostrara el siguiente error.

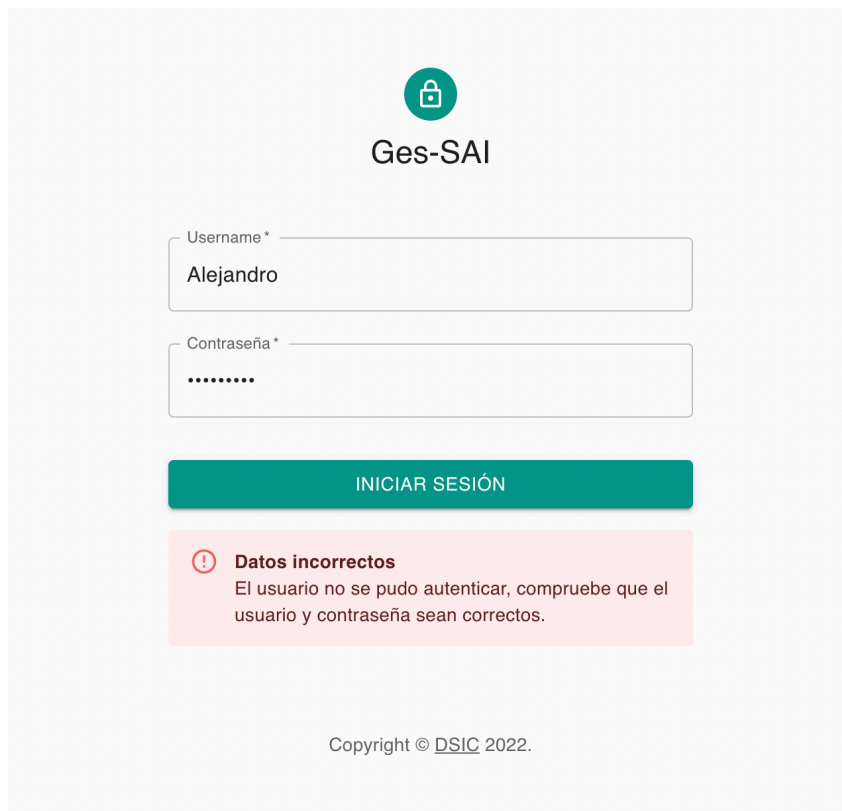


Imagen 2 - Login error

2.2. Sais

El usuario de rol técnico será el encargado de administrar el módulo **SAIS**. A continuación, se presenta la forma correcta de dar de alta a un dispositivo SAI.

De clic en el módulo SAIS y luego de clic en el botón de la parte inferior derecha, como se indica en la imagen.

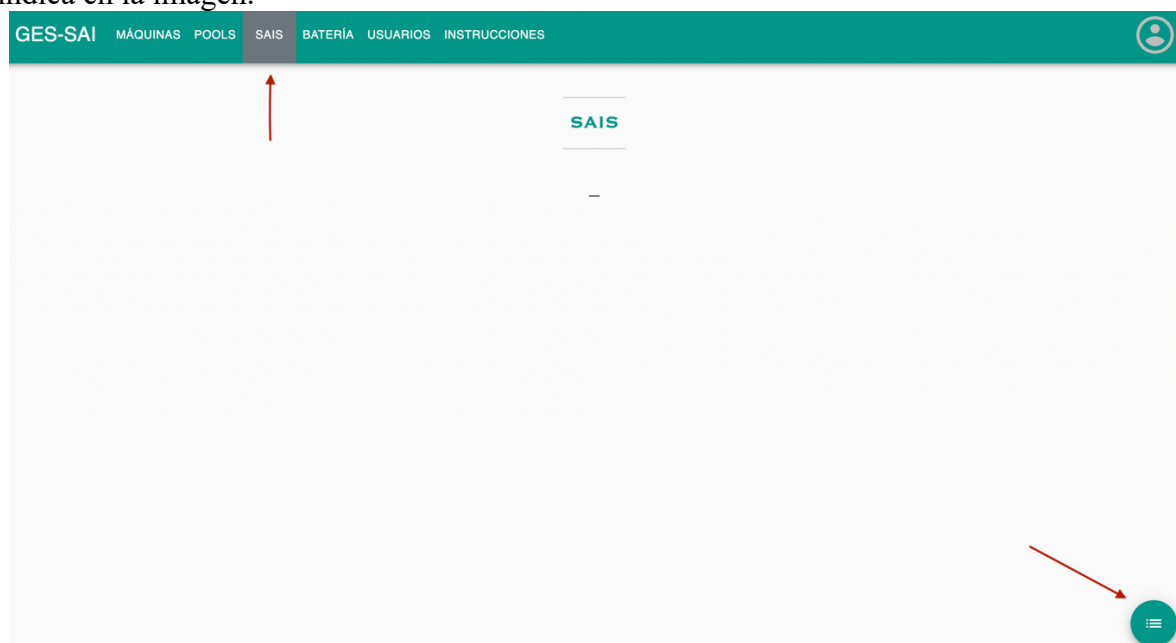


Imagen 3 - Modulo Sais

Se abrirá la siguiente pantalla, ingrese los datos que se le soliciten del SAI y luego de clic en el botón “Guardar” para darlo de alta.

The screenshot shows the 'SAI' configuration form. The form fields are as follows:

Nombre DNS	sai1f
Url	192.168.1.100
Usuario de conexión	root
Clave de Autorización	*****
Clave Privada	*****
Codigo MIB	1.3.6.1.2.1.33.1.2.4.0
Tiempo de apagado - (minutos)	10
Tiempo de encendido - (minutos)	50
Protocolo	SNMP

At the bottom of the form, there are two buttons: '✓ GUARDAR' and 'X CANCELAR'.

Imagen 4 - Alta SAI

A continuación, el sistema lo redireccionara al módulo de SAIS mostrándole la lista de todos los SAIS dados de alta en el sistema.

The screenshot shows the 'SAIS' list in the application. The list contains two entries:

sai1f	▼
sai2f	▼

At the bottom right of the screen, there is a green circular button with a white hamburger menu icon.

Imagen 5 - Lista de SAIS

Para ver el detalle de un dispositivo SAI, de clic sobre el nombre y se le desplegara su información.

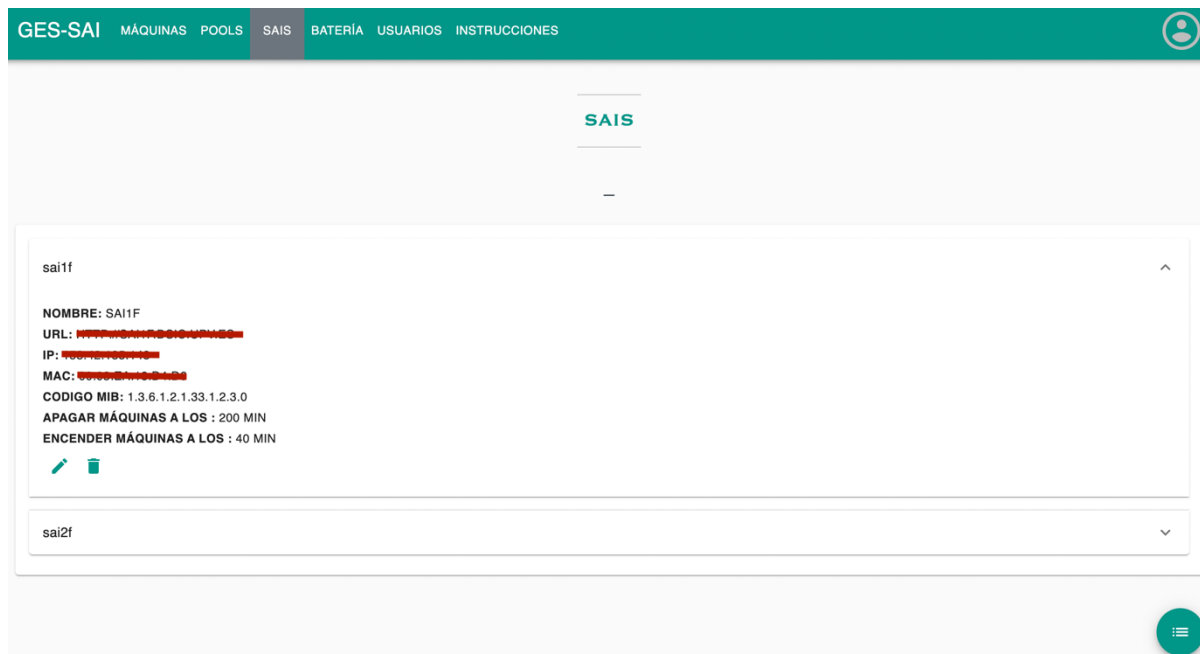




Imagen 6 - Detalle de SAI

Si desea **editar**lo de clic en el botón , si por el contrario desea **eliminar**lo de clic en el botón . A continuación, se presentan los 2 casos respectivamente.

Editar

Únicamente se podrá editar el tiempo de apagado y encendido de un dispositivo SAI.

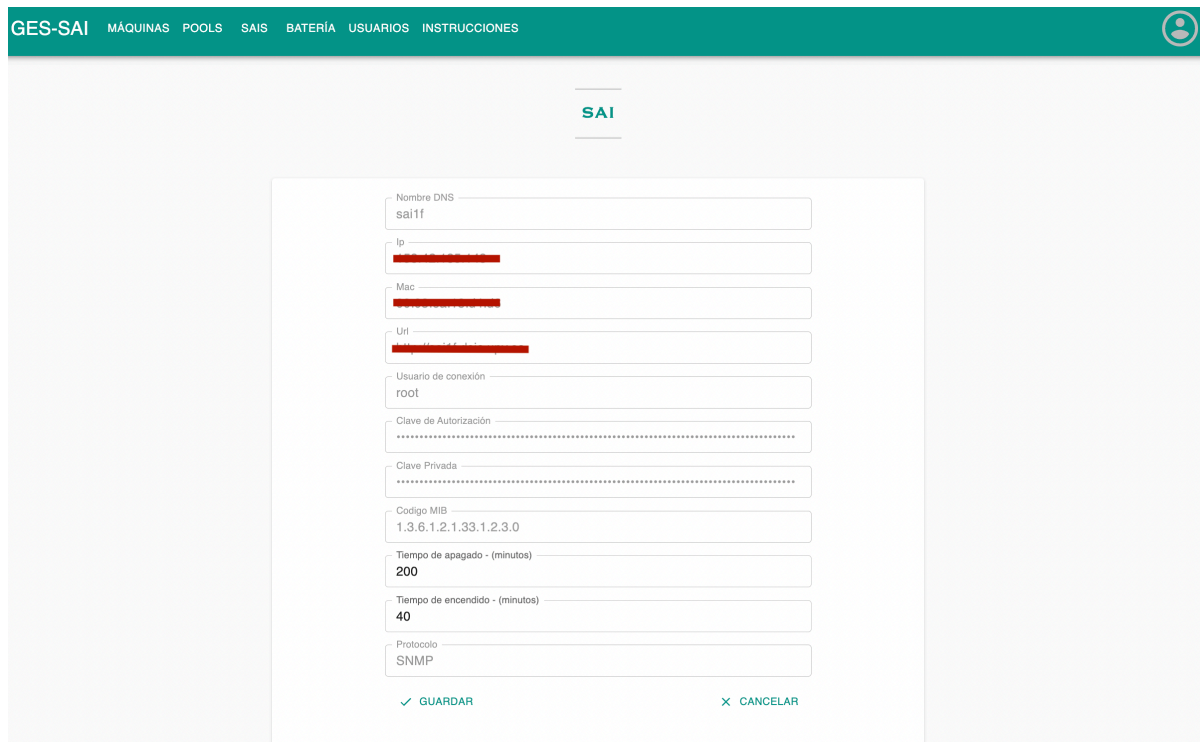


Imagen 7 - Editar SAI

Eliminar

Si elige la opción de **eliminar** el SAI, el sistema le mostrara el siguiente mensaje de confirmación.

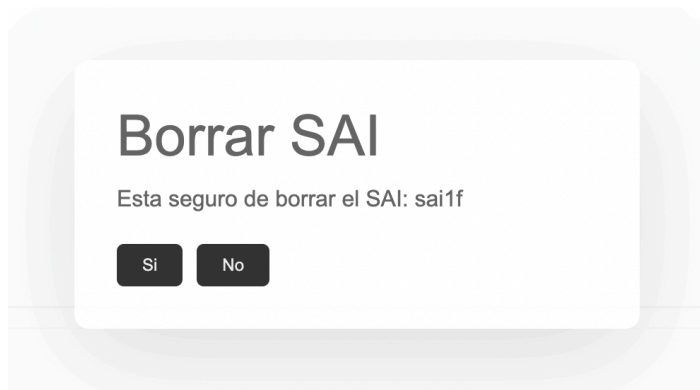


Imagen 8 - Mensaje eliminar SAI

2.3. Pools

Los Pools son los dispositivos que agrupan las diferentes máquinas virtuales del sistema. Para dar de alta a un Pool, de clic en el módulo POOLS y luego de clic en el botón de la parte inferior derecha, como se indica en la imagen.

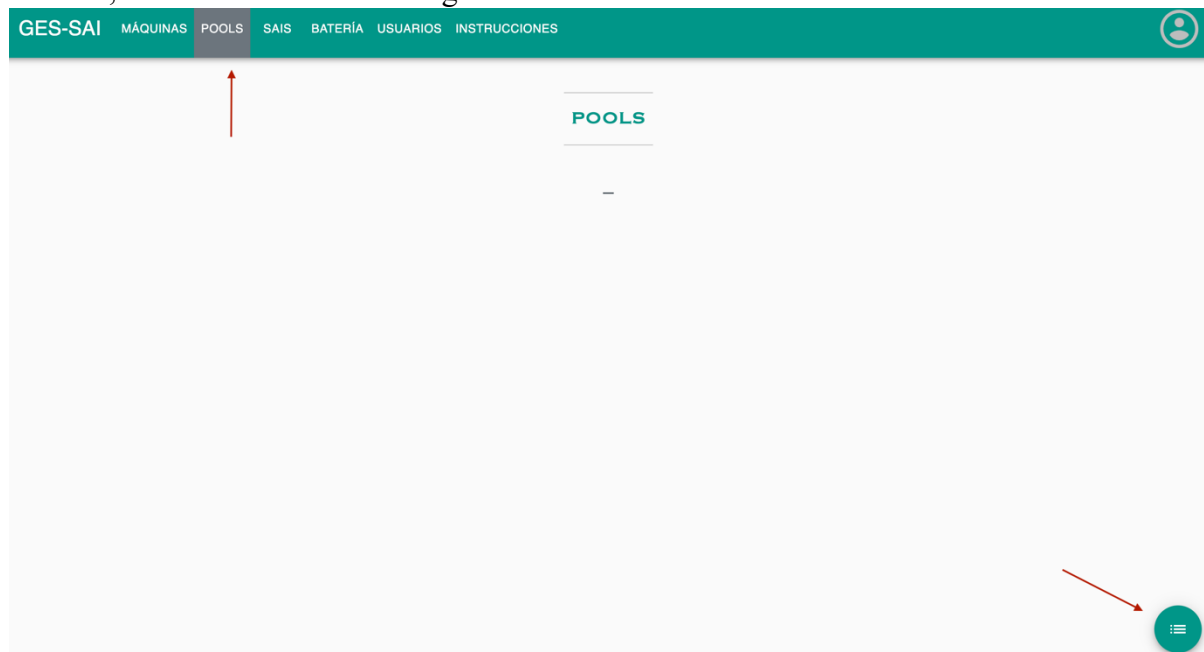


Imagen 9 - Módulo Pools

Se abrirá la siguiente pantalla, ingrese los datos que se le soliciten del POOL y luego de clic en el botón "**Guardar**" para darlo de alta.

The screenshot shows a web application interface with a green header containing the menu items: GES-SAI, MÁQUINAS, POOLS, SAIS, BATERÍA, USUARIOS, and INSTRUCCIONES. A user profile icon is visible in the top right corner. The main content area is titled 'POOL' and contains a form for creating a new pool. The form fields are: 'Nombre' (pool 7), 'Ip' (redacted), 'Url' (redacted), 'Username' (redacted), 'Contraseña' (masked with dots), 'Tipo de Hipervisor' (Xen), and 'SAI' (sai1f). At the bottom of the form are two buttons: 'GUARDAR' (with a checkmark icon) and 'CANCELAR' (with an 'X' icon).

Imagen 10 - Alta Pool

A continuación, el sistema lo redireccionara al módulo de POOLS mostrándole la lista de todos los POOLS dados de alta en el sistema.

The screenshot shows the 'POOLS' module in the application. The header is the same as in the previous image, but the 'POOLS' menu item is highlighted. The main content area is titled 'POOLS' and displays a list of pools. A single pool is visible: 'Pool Dev7'. A dropdown arrow is located to the right of the pool name. A hamburger menu icon is located in the bottom right corner of the page.

Imagen 11 - Listado Pool

Para ver el detalle de un dispositivo POOL, de clic sobre el nombre y se le desplegara su información.

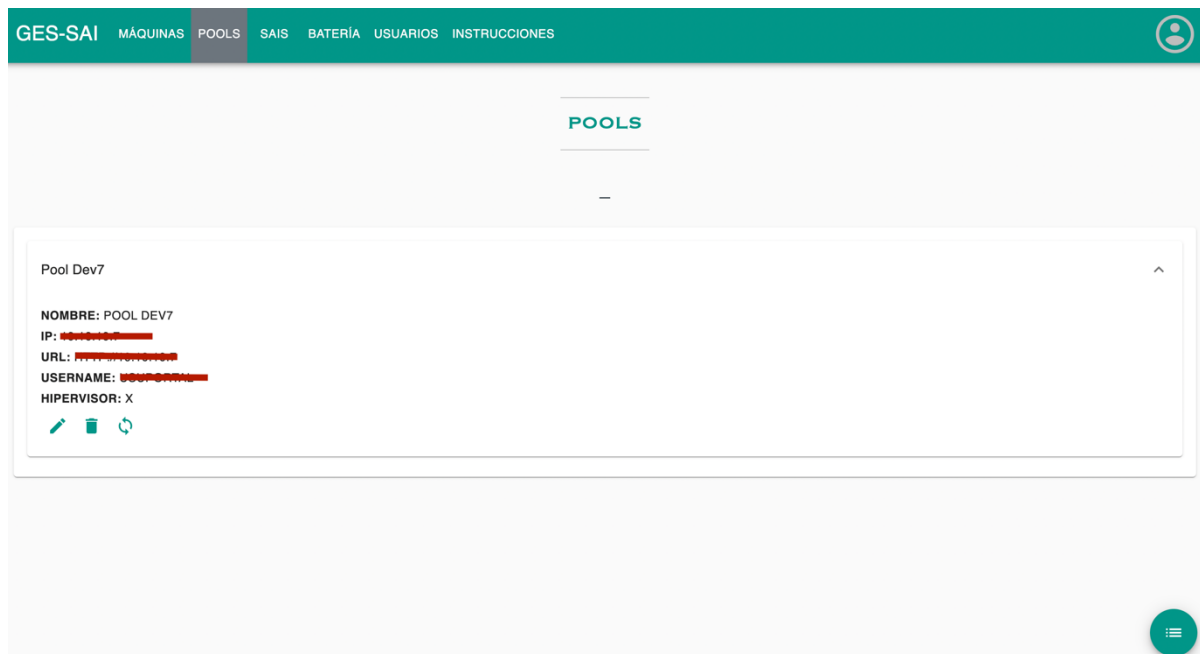





Imagen 12 - Detalle Pool

Como se puede observar en la imagen anterior, el POOL cuenta con 3 funciones las cuales son, editar, eliminar, sincronizar. Si desea **editar**lo de clic en el botón , si desea **eliminar**lo de clic en el botón  o si desea **sincronizar**lo de clic en el botón . A continuación, se presentan los 3 casos respectivamente.

Editar

Únicamente se podrá editar los dispositivos SAI que están conectados al POOL.

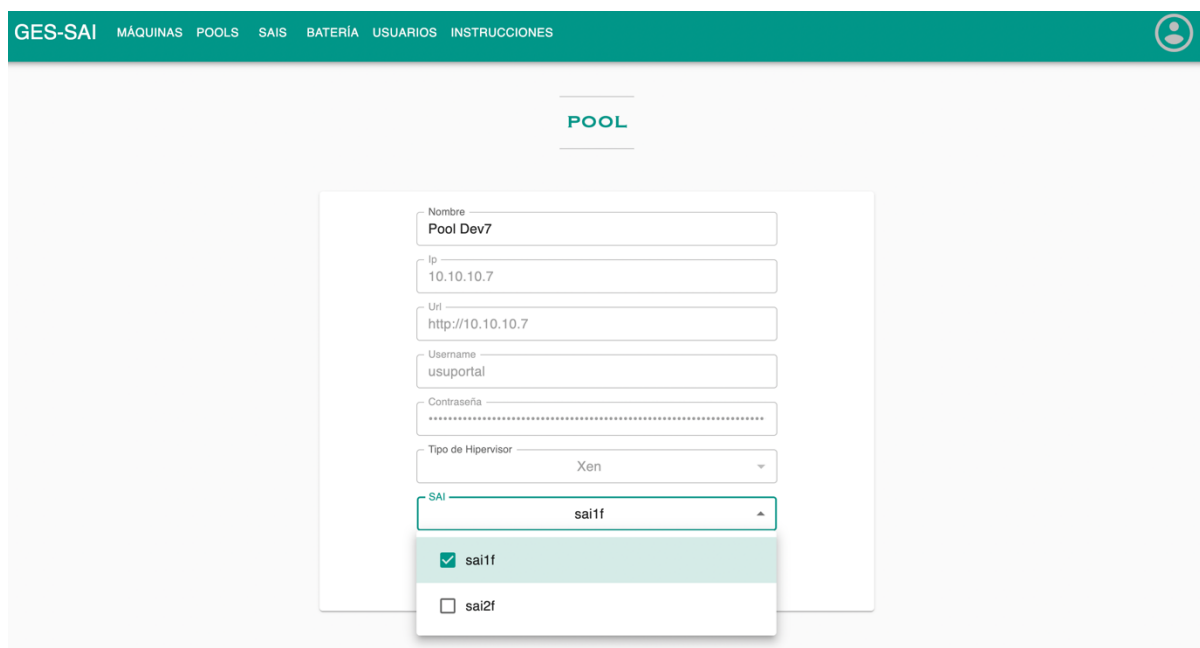


Imagen 13 - Editar Pool

Eliminar

Si elige la opción de **eliminar** el POOL, el sistema le mostrara el siguiente mensaje de confirmación.

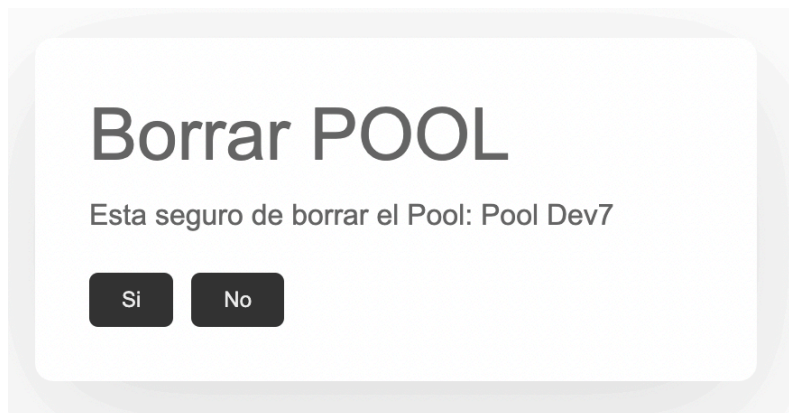


Imagen 14 - Mensaje eliminar Pool

2.4. Máquinas

Este módulo cuenta con 2 funcionalidades distintas, la primera dar de alta a las **máquinas físicas** y a la segunda dar de alta a las **dependencias**.

2.4.1. Alta máquina física / almacenamiento

Para dar de alta a una Máquina física o de almacenamiento, de clic en el módulo **MÁQUINAS** y luego de clic en el botón "**Alta máquina física**" ubicado en la parte inferior derecha, como se indica en la imagen.

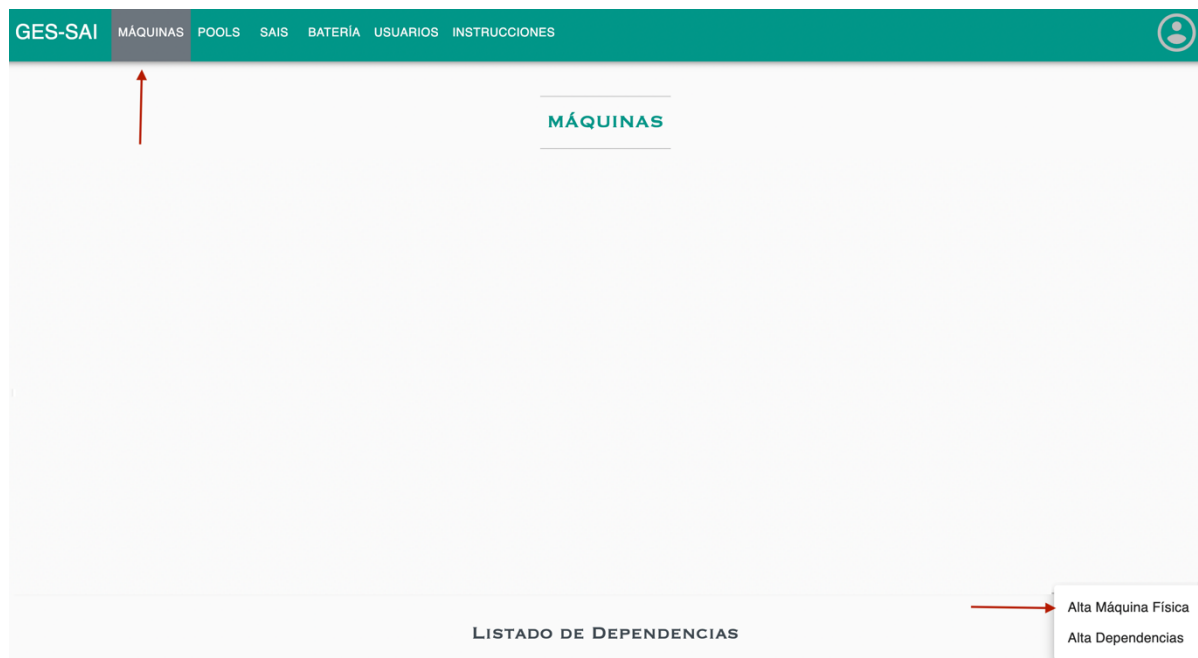


Imagen 15 - Módulo Máquinas

Se abrirá la siguiente ventana, donde podrá elegir si dar de alta una máquina física o una máquina de almacenamiento.

Nota: *Ambas son máquinas físicas, la diferencia es que las físicas el sistema busca sus datos automáticamente, y las de almacenamiento el usuario ingresa todos sus datos manualmente.*



Imagen 16 - Alta máquina

➤ Alta de Máquina física

Escriba el nombre de la máquina física que desea dar de alta (recuerde que esta máquina debe estar dada de alta en la base de datos de Piolín). Luego de clic en el botón “**Buscar**”, si el sistema encuentra la máquina, se llenarán sus datos automáticamente, como se muestra a continuación.

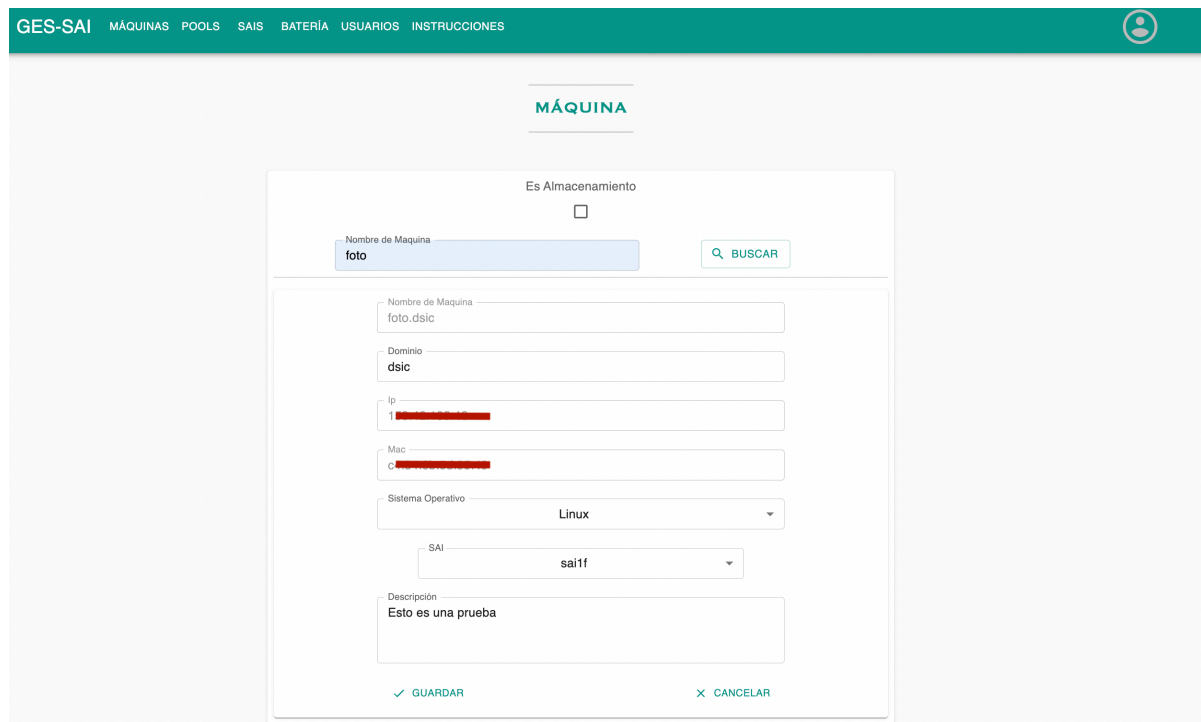
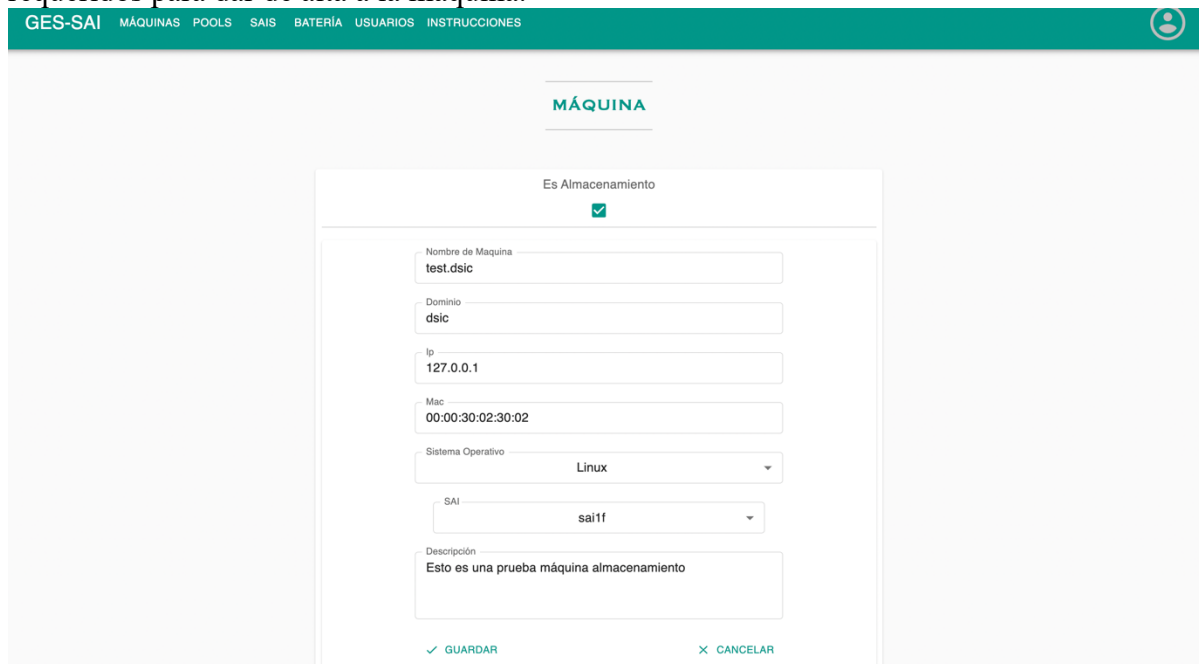


Imagen 17 - Alta máquina física

Solo tiene que ingresar el dominio, el sistema operativo, el SAI al que está conectada y si desea una descripción. Por último, de clic en el botón “**Guardar**” para darla de alta en el sistema.

➤ Alta de Máquina almacenamiento

Seleccione la casilla “**Es Almacenamiento**” se desplegará un formulario con todos los campos requeridos para dar de alta a la máquina.

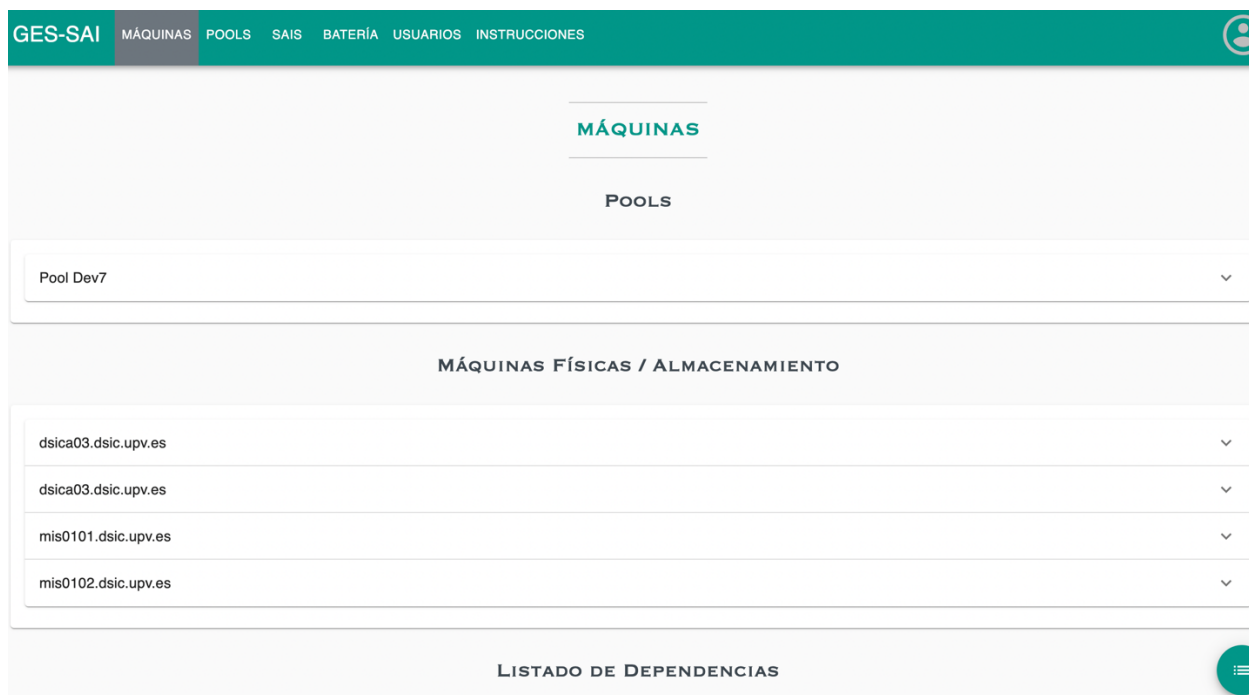


The screenshot shows a web application interface with a green header containing navigation links: GES-SAI, MÁQUINAS, POOLS, SAIS, BATERÍA, USUARIOS, INSTRUCCIONES, and a user profile icon. The main content area is titled 'MÁQUINA' and features a form for adding a storage machine. The form includes a checked checkbox labeled 'Es Almacenamiento'. Below this, there are several input fields: 'Nombre de Máquina' (test.dsic), 'Dominio' (dsic), 'Ip' (127.0.0.1), 'Mac' (00:00:30:02:30:02), 'Sistema Operativo' (Linux), and 'SAI' (sai1f). A 'Descripción' field contains the text 'Esto es una prueba máquina almacenamiento'. At the bottom of the form are two buttons: 'GUARDAR' (with a checkmark icon) and 'CANCELAR' (with an 'X' icon).

Imagen 18 - Alta máquina almacenamiento

Ingrese todos los campos requeridos, por último, de clic en el botón “**Guardar**” para darla de alta en el sistema.

Una vez de alta una máquina, en cualquiera de los 2 casos el sistema lo redirigirá nuevamente al módulo MÁQUINAS, listando todos los Pools, Máquinas y Dependencias dadas de alta.



The screenshot shows the 'MÁQUINAS' module in the web application. The header is the same as in the previous image. The main content area is titled 'MÁQUINAS' and has a sub-section 'POOLS' with a dropdown menu showing 'Pool Dev7'. Below this is a section titled 'MÁQUINAS FÍSICAS / ALMACENAMIENTO' which contains a list of physical machines. The list has four entries, each with a dropdown arrow on the right:

MÁQUINAS FÍSICAS / ALMACENAMIENTO	
dsica03.dsic.upv.es	▼
dsica03.dsic.upv.es	▼
mis0101.dsic.upv.es	▼
mis0102.dsic.upv.es	▼

At the bottom of the page, there is a section titled 'LISTADO DE DEPENDENCIAS' and a hamburger menu icon.

Imagen 19 - Lista máquinas

Para ver el detalle de una máquina, de clic sobre el nombre y se le desplegara su información.

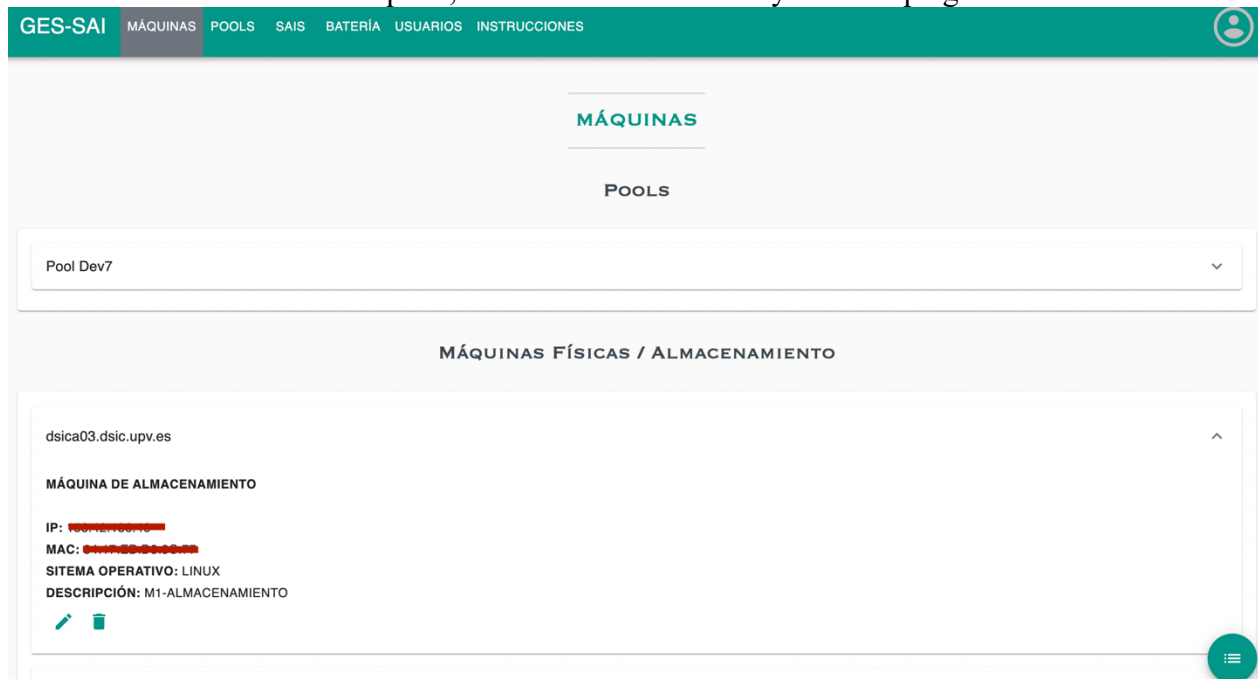


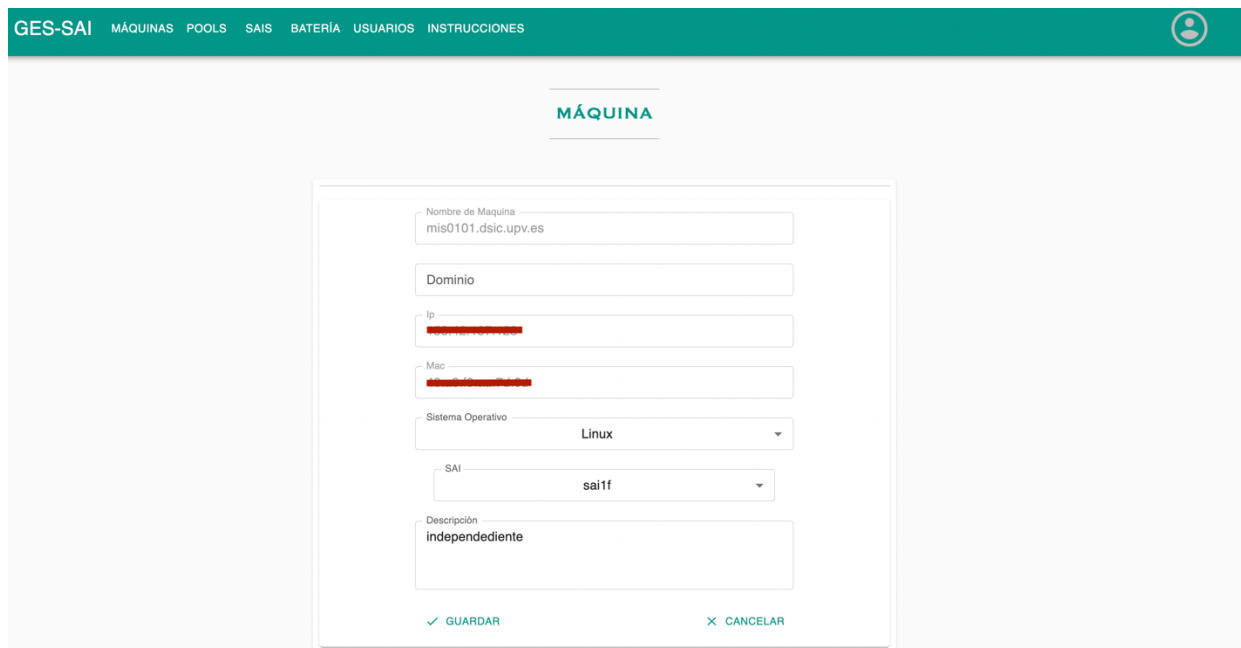


Imagen 20 - Detalle máquinas

Como se puede observar en la imagen anterior, la MÁQUINA cuenta con 2 funciones las cuales son, editar y eliminar. Si desea **editarla** de clic en el botón , si desea **eliminarla** de clic en el botón  .

Editar

Se mostrarán los campos habilitados para aditar.



GES-SAI MÁQUINAS POOLS SAIS BATERÍA USUARIOS INSTRUCCIONES

MÁQUINA

Nombre de Máquina
mis0101.dsic.upv.es

Dominio

Ip

Mac

Sistema Operativo
Linux

SAI
sai1f

Descripción
independiente

✓ GUARDAR × CANCELAR

Imagen 21 - Editar máquina

Eliminar

Si elige la opción de **eliminar** la MÁQUINA, el sistema le mostrara el siguiente mensaje de confirmación.



Imagen 22 - Mensaje eliminar máquina

2.4.2. Alta dependencias

Las dependencias se crean a partir de todas aquellas máquinas que comparten recursos entre sí, por lo que, si una máquina presta servicios a otras máquinas, estas son máquinas dependientes. Todas las máquinas (**Hijo**) son máquinas dependientes de la máquina (**Padre**).

Para dar de alta a una DEPENDENCIA, de clic en el módulo MÁQUINAS y luego de clic en el botón “**Alta Dependencias**” ubicado en la parte inferior derecha, como se indica en la imagen.

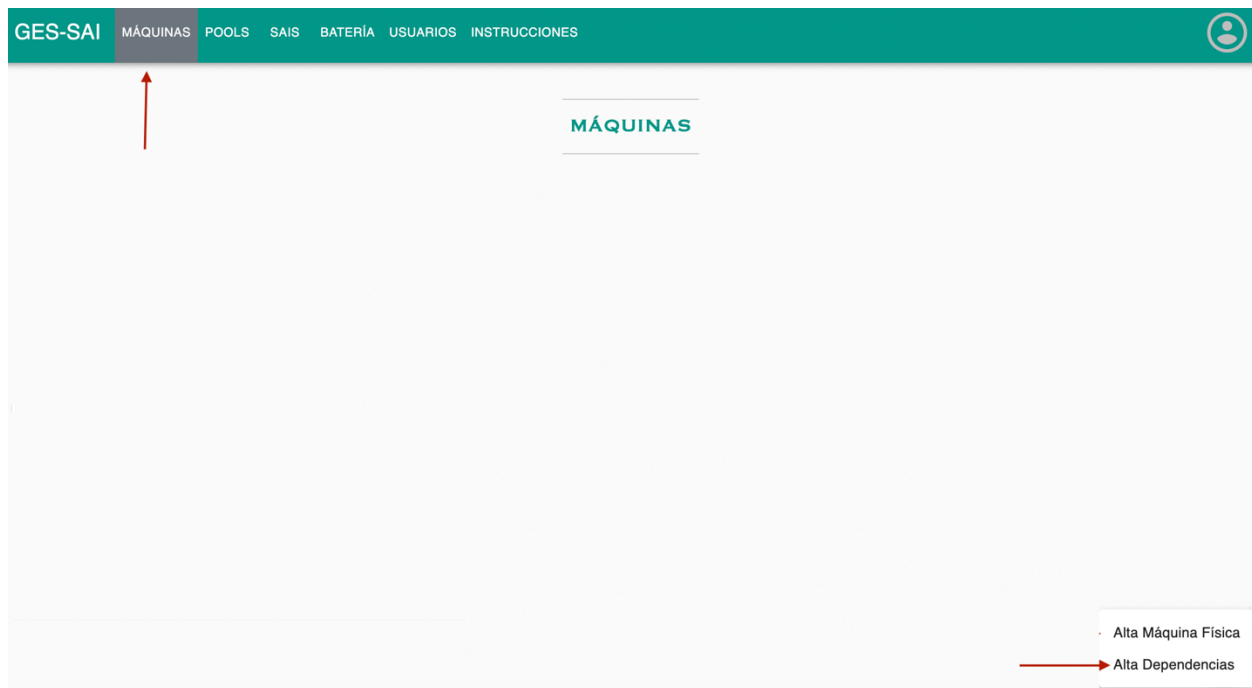


Imagen 23 - Módulo máquinas - Alta dependencias

Se abrirá la siguiente ventana, donde podrá dar de alta una Dependencia entre dos máquinas. Este módulo cuenta con 3 divisiones, en la parte superior de la pantalla se encuentra el apartado de alta dependencia, en la parte inferior izquierda de la pantalla se encuentra el apartado de eliminar dependencia, y por último en la parte inferior derecha de la pantalla, se encuentra el árbol de dependencia de la máquina seleccionada.

Nota: La máquina de la **izquierda** es la máquina (**Padre**) y la máquina de la **derecha** es la máquina (**Hijo**) - (máquina dependiente).

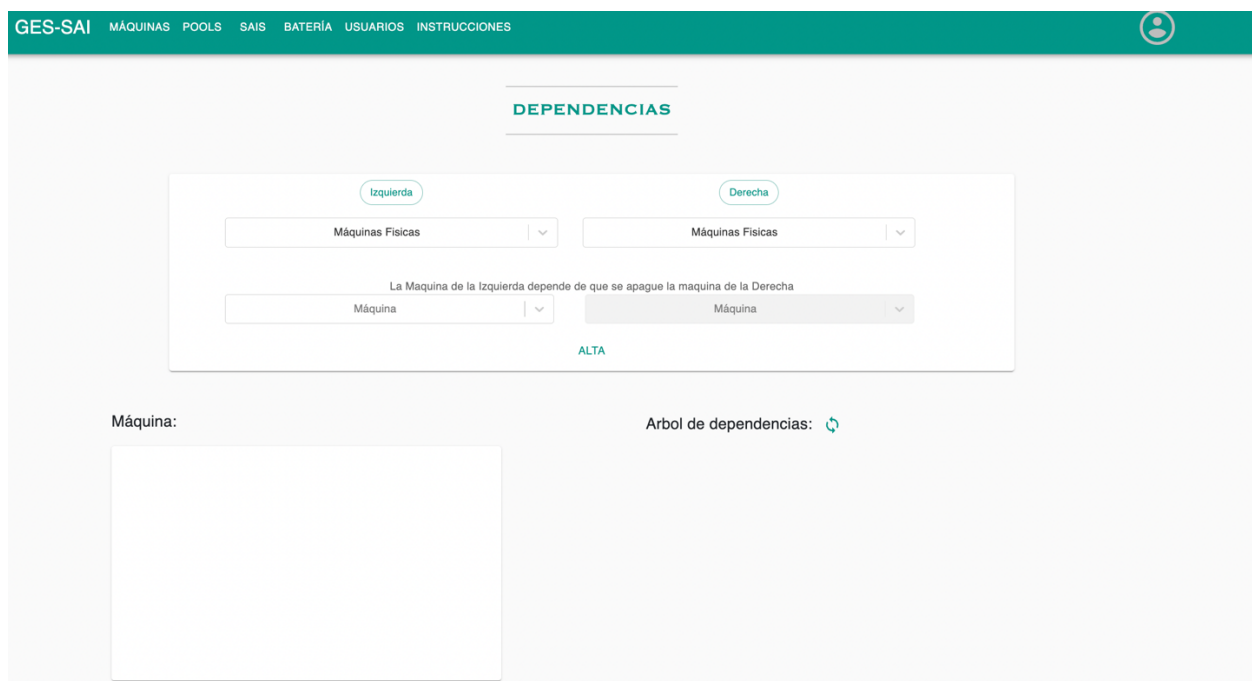


Imagen 24 - Módulo dependencias

➤ Alta Dependencia

En la parte **Izquierda** seleccione el primer desplegable y elija que tipo de máquina desea que sea la máquina (**Padre**), los tipos pueden ser: (Máquinas físicas, Máquinas de almacenamiento y los diferentes Pools).



Imagen 25 - Alta dependencia - Tipo máquina padre

Una vez hecho esto, seleccione el segundo desplegable de la parte **Izquierda** y elija la máquina que desea que sea la máquina (**Padre**).

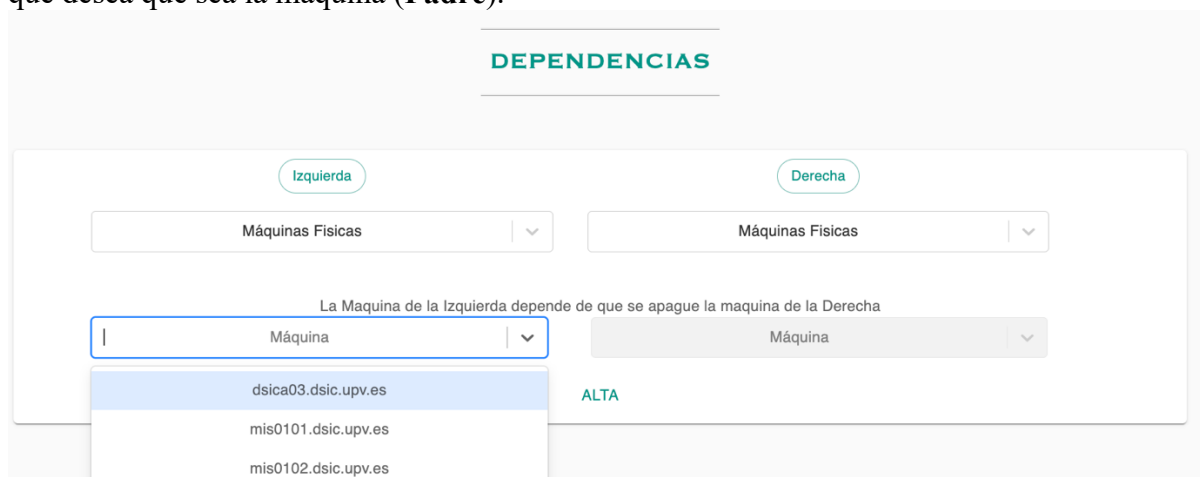


Imagen 26 - Alta dependencia – Máquina padre

En la parte **Derecha** seleccione el primer desplegable y elija que tipo de máquina desea que sea la máquina (**Hijo**), los tipos pueden ser: (Máquinas físicas, Máquinas de almacenamiento y los diferentes Pools).

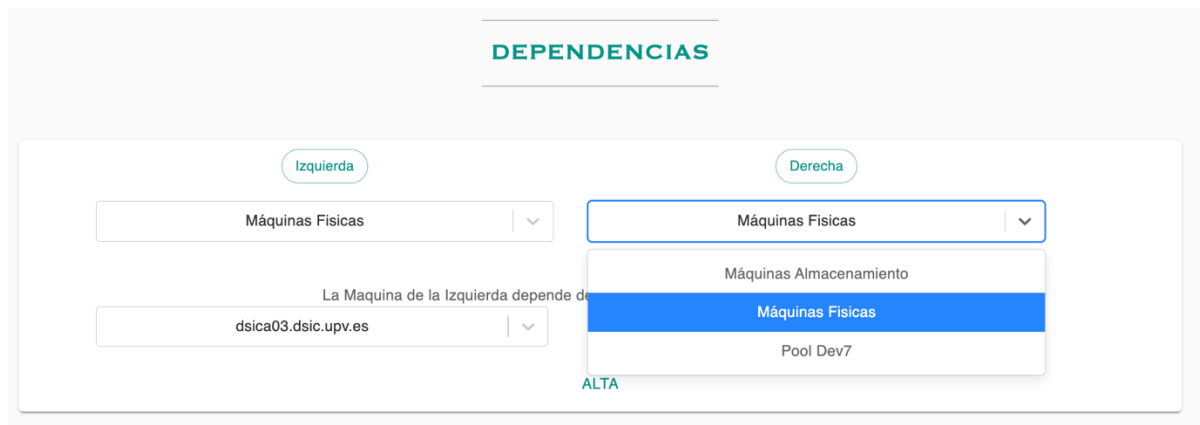


Imagen 27 - Alta dependencia - Tipo máquina hijo

Una vez hecho esto, seleccione el segundo desplegable de la parte **Derecha** y elija la máquina que desea que sea la máquina (**Hijo**).

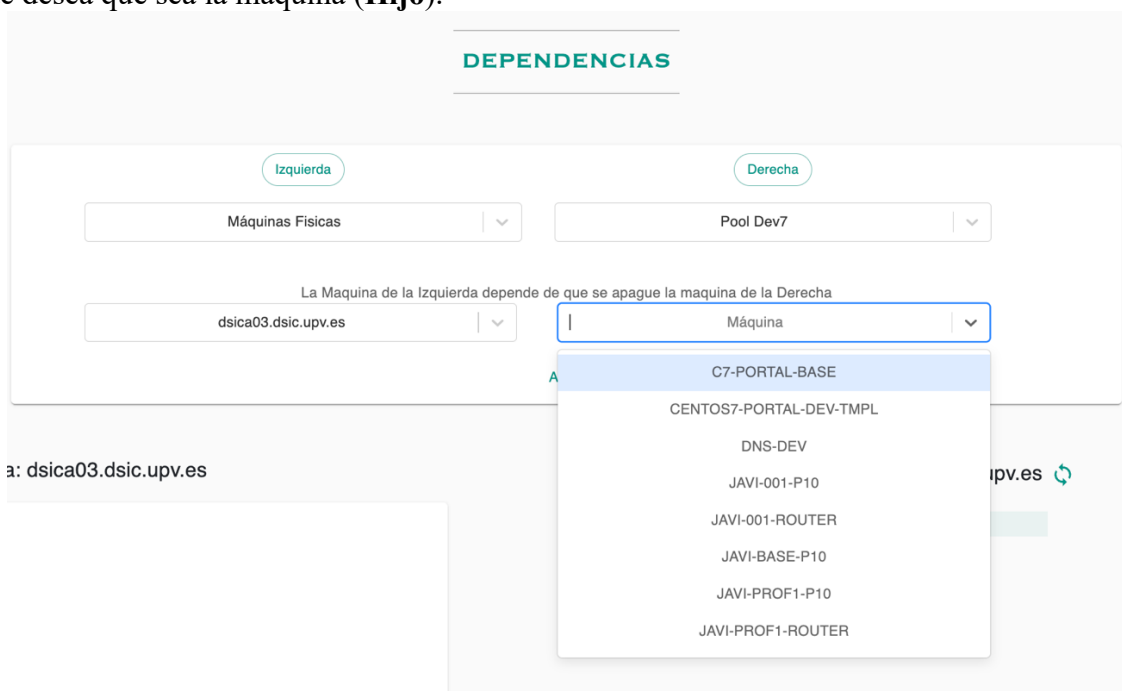


Imagen 28 – Alta dependencia - Máquina hijo

Por último, teniendo las dos máquinas seleccionadas, de clic en el botón “Alta” para guardar los cambios.



Imagen 29 - Paso final - Alta dependencia

El sistema mostrara automáticamente en la parte inferior izquierda la **máquina dada de alta** y en la parte inferior derecha su **árbol de dependencia**, como se ve a continuación.

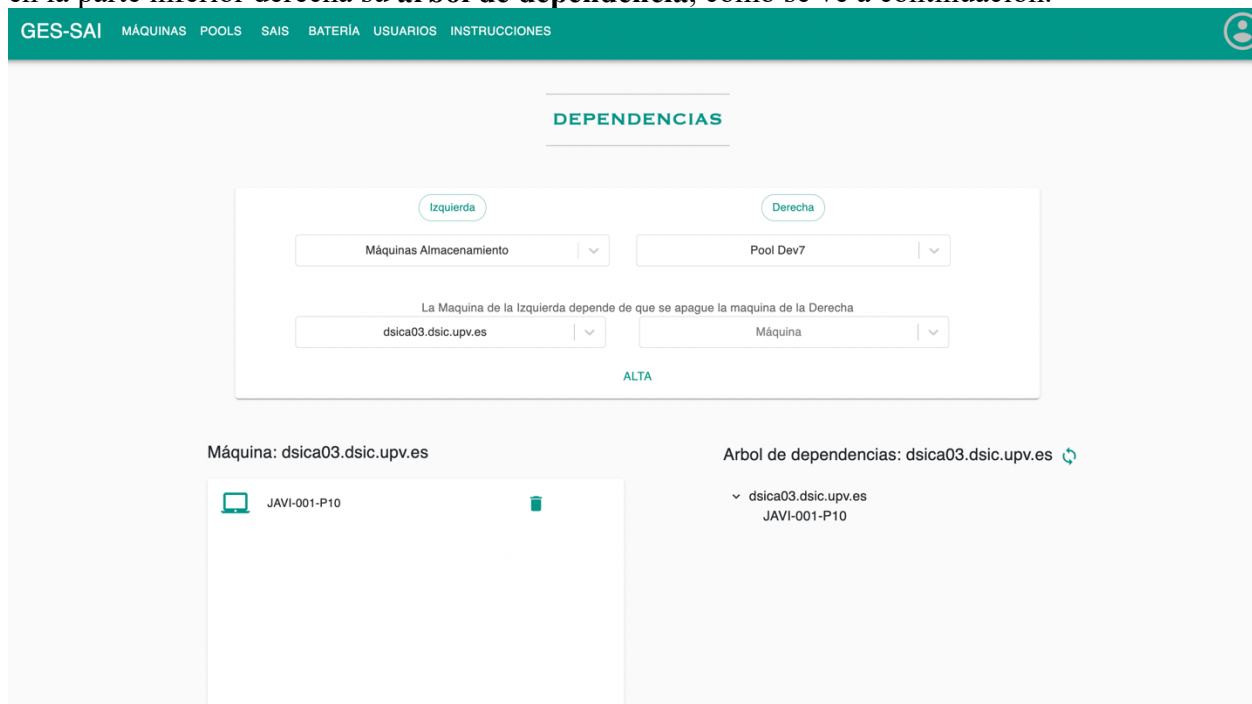


Imagen 30 - Interfaz de independencias – árbol de dependencias

➤ Baja Dependencia

Para dar de baja una dependencia, solo basta con buscar en los desplegables de la parte **Izquierda** el cual pertenece al apartado (**Padre**), la máquina que deseo ver sus dependencias.

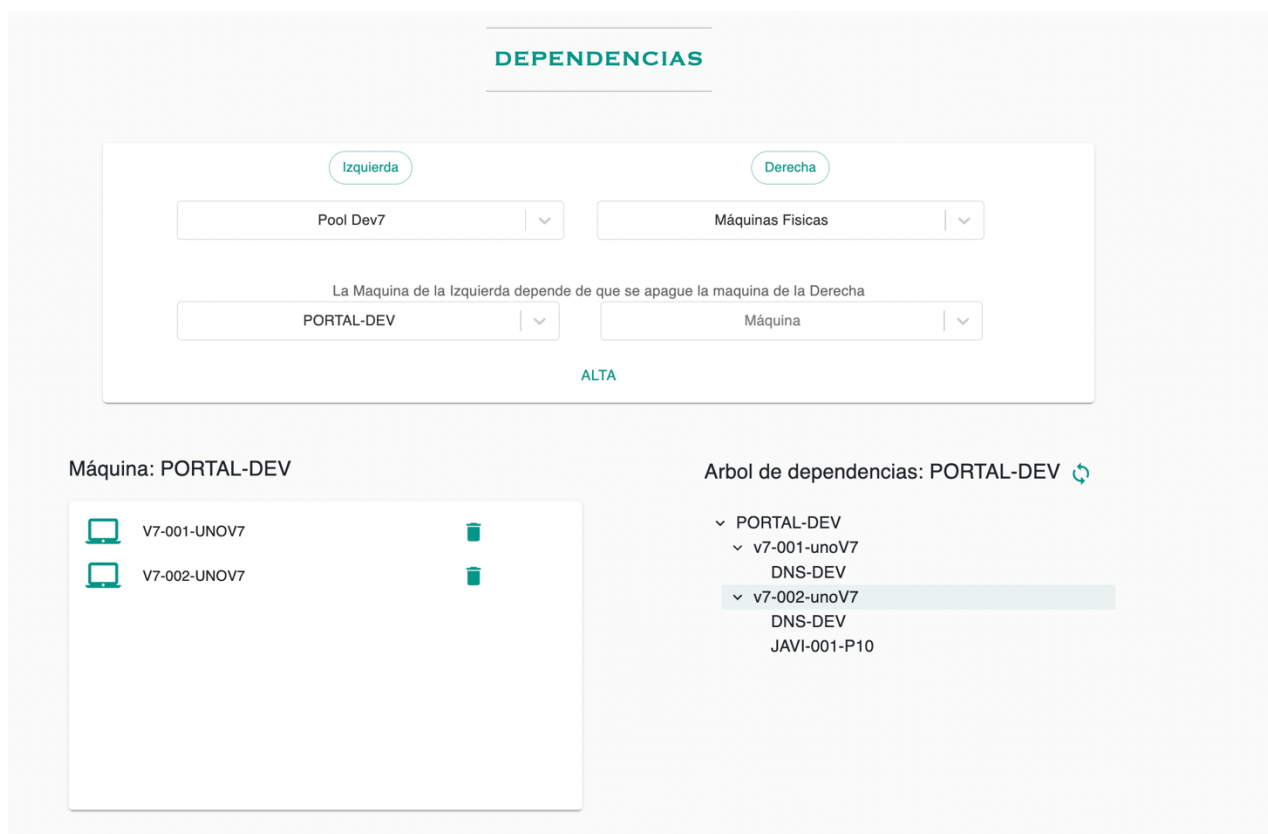



Imagen 31 - Baja dependencia

De clic en el botón  para dar de baja la dependencia. El sistema inmediatamente eliminara la dependencia asociada con la máquina padre.

Nota: Para este ejemplo la máquina (**Padre**) es *PORTAL-DEV* y sus máquinas (**Hijos**) son *V7-0001-UNOV7* y *V7-002-UNOV7*

2.5. Usuarios

El usuario de rol técnico será el encargado de administrar el módulo **USUARIOS**. A continuación, se presenta la forma correcta de dar de alta a un usuario.

De clic en el módulo **USUARIOS** y luego de clic en el botón de la parte inferior derecha, como se indica en la imagen.

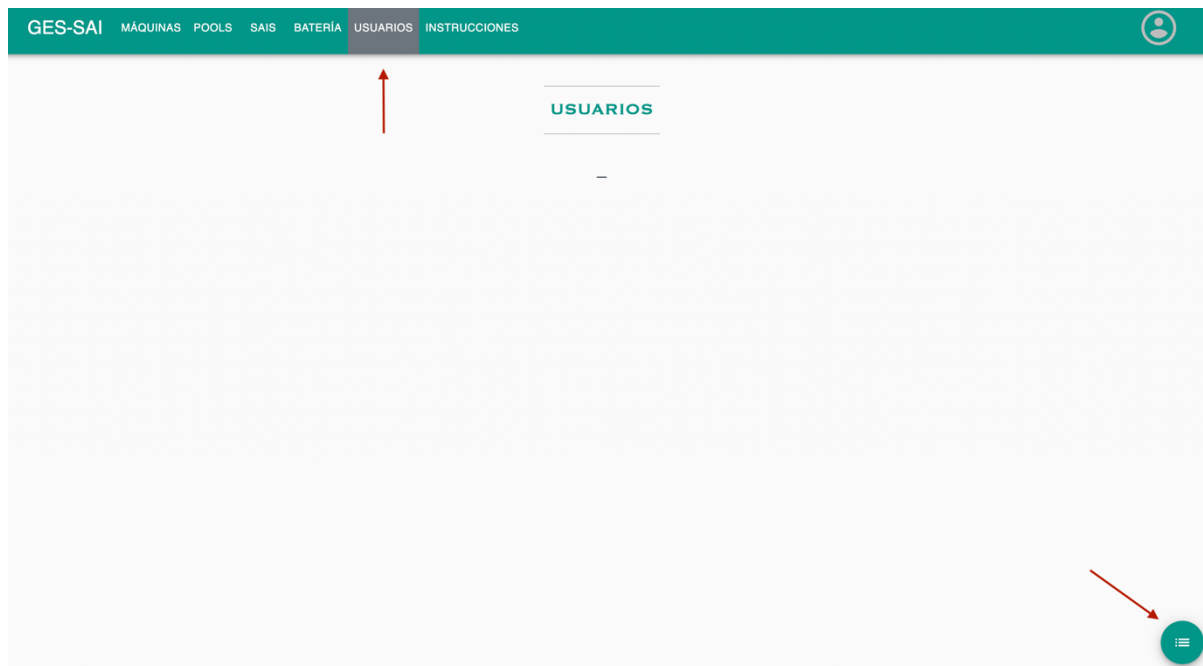


Imagen 32 - Módulo usuarios

Se abrirá la siguiente pantalla, ingrese los datos que se le soliciten del USUARIO y luego de clic en el botón “**Guardar**” para darlo de alta.

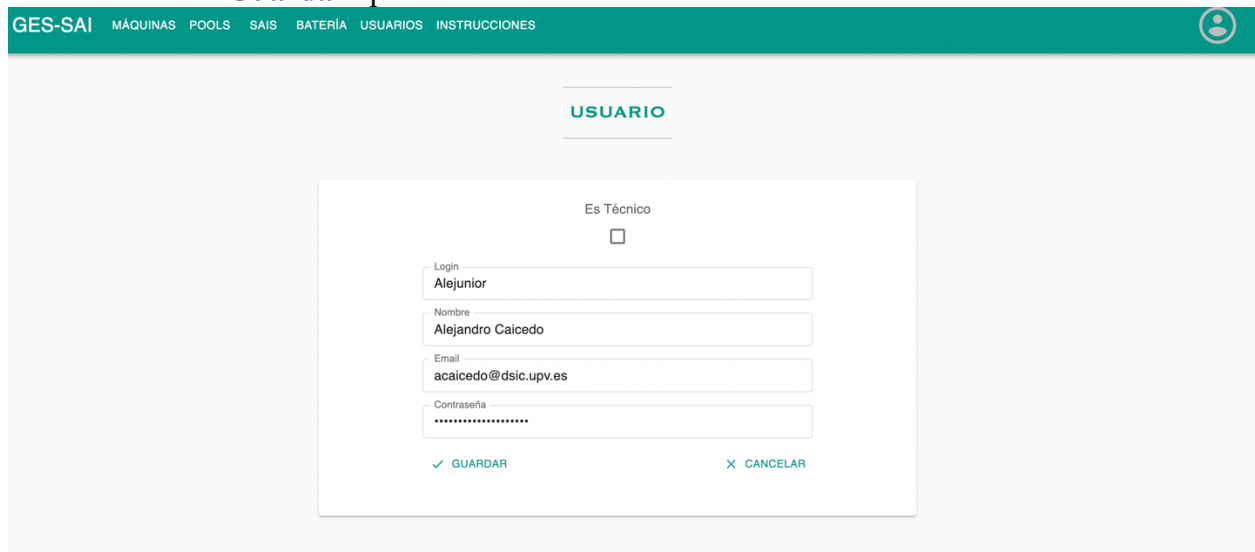


Imagen 33 - Alta usuario

Solo si desea que el usuario sea de rol Técnico y tenga acceso a todos los módulos, marque la casilla “**Es Técnico**”.

Es Técnico



Imagen 34 - Opción técnico

A continuación, el sistema lo redireccionara al módulo de USUARIOS mostrándole la lista de todos los Usuarios dados de alta en el sistema.

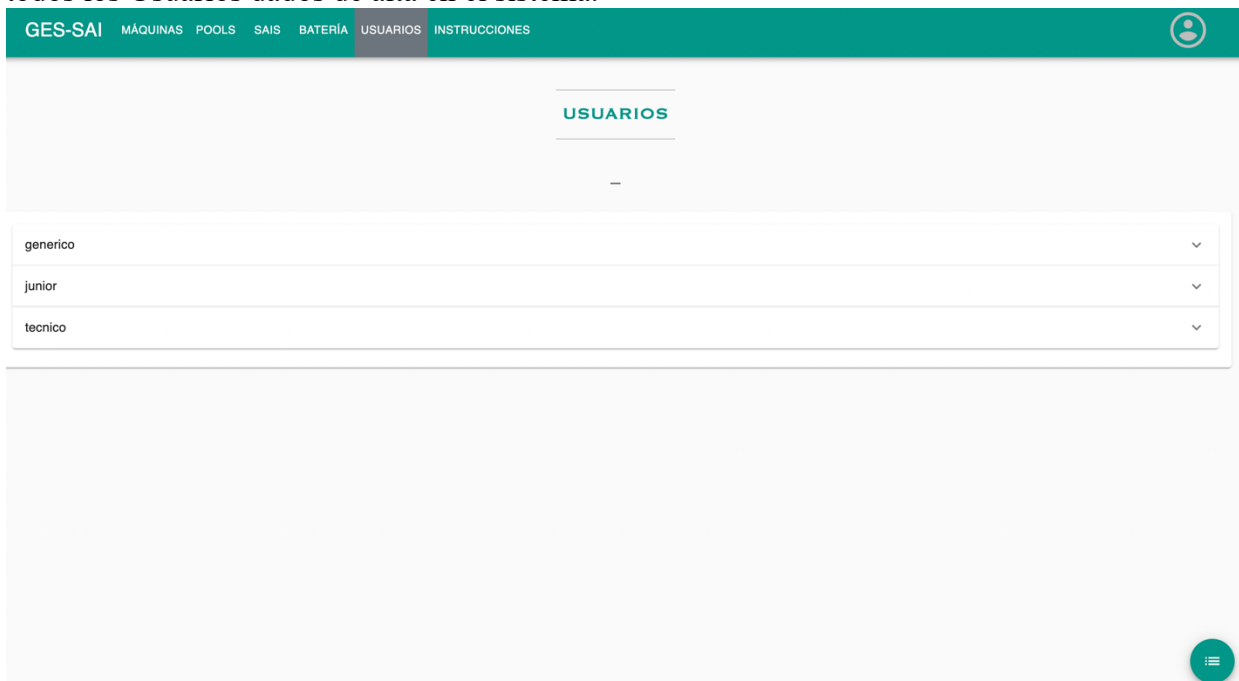


Imagen 35 - Listado usuarios

Para ver el detalle de un Usuario, de clic sobre el nombre y se le desplegara su información.

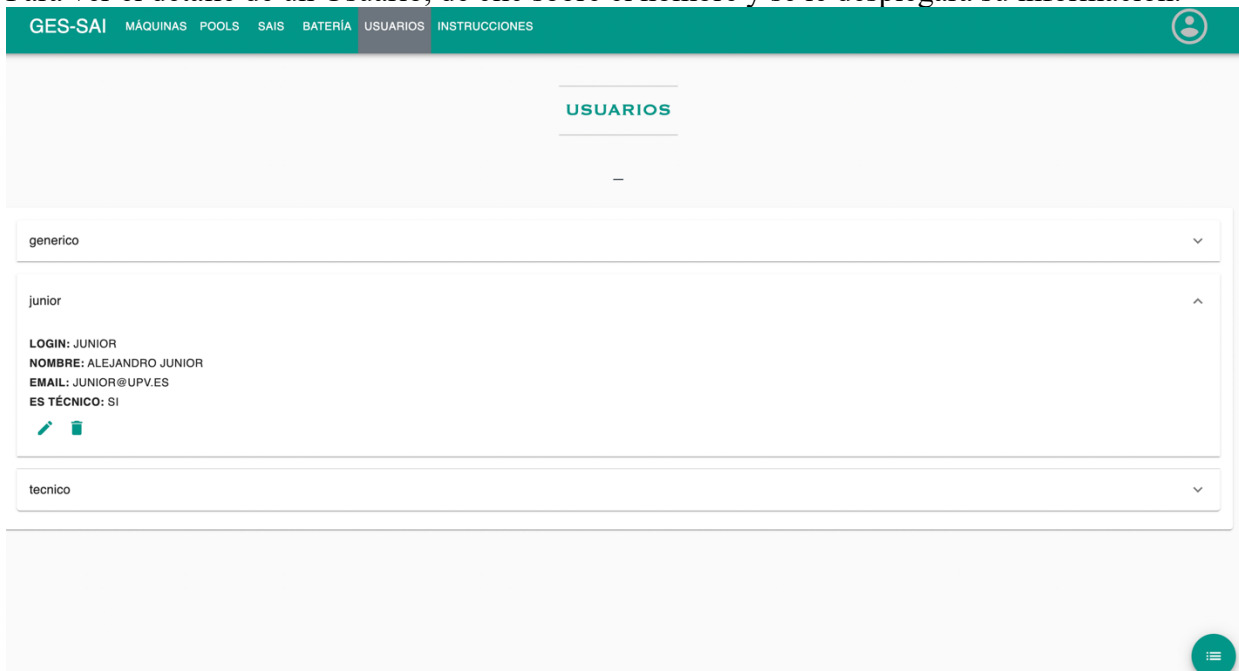


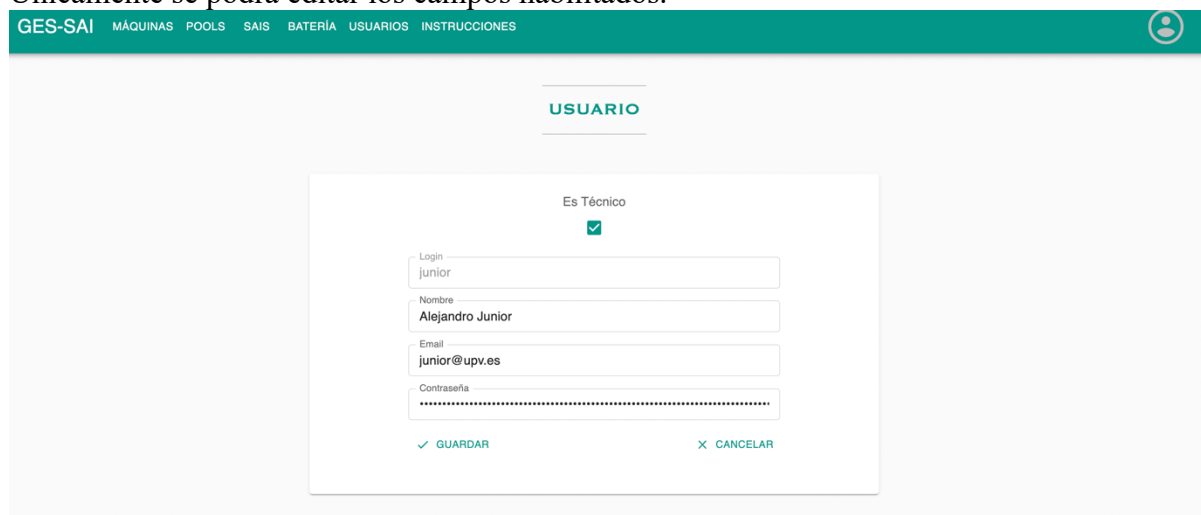


Imagen 36 - Detalle usuario

Si desea **editarlo** de clic en el botón , si por el contrario desea **eliminarlo** de clic en el botón . A continuación, se presentan los 2 casos respectivamente.

Editar

Únicamente se podrá editar los campos habilitados.



GES-SAI MÁQUINAS POOLS SAIS BATERÍA USUARIOS INSTRUCCIONES

USUARIO

Es Técnico

Login
junior

Nombre
Alejandro Junior

Email
junior@upv.es

Contraseña

GUARDAR CANCELAR

Imagen 37 - Editar usuario

Eliminar

Si elige la opción de **eliminar** el Usuario, el sistema le mostrara el siguiente mensaje de confirmación.

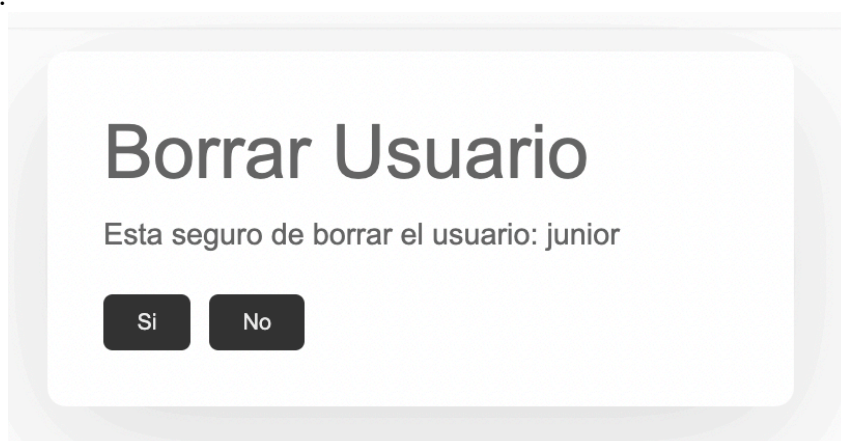


Imagen 38 - Mensaje eliminar usuario

2.6. Baterías

En este módulo el usuario de rol técnico podrá visualizar las baterías de todos los dispositivos SAI dados de alta en el sistema. De clic en el módulo **BATERÍA** y se mostrara la siguiente pantalla, con el nivel de baterías de cada uno de los SAI.

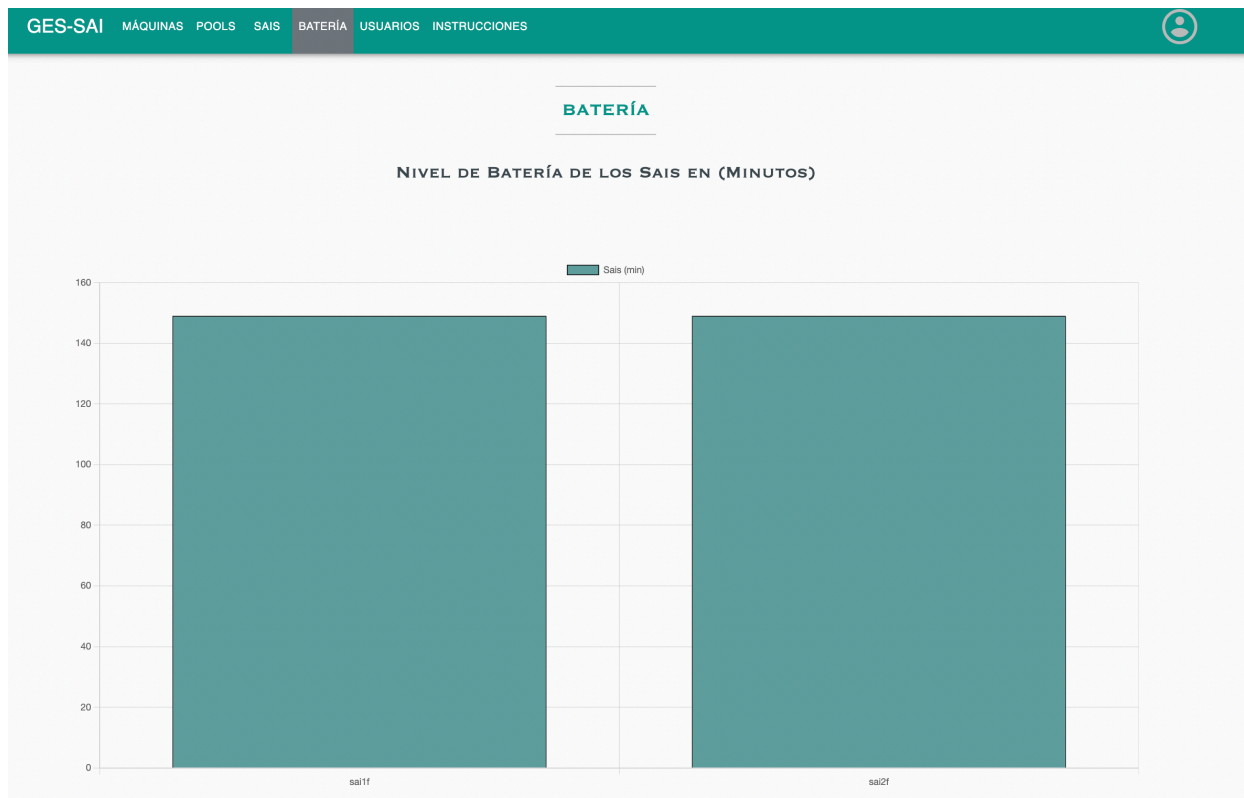


Imagen 39 - Módulo baterías

2.7. Instrucciones

En este módulo el usuario podrá descargar el archivo de clave publica y visualizar las instrucciones para su instalación.

Para ver las instrucciones y descargar el archivo de clave publica, de clic en el módulo **INSTRUCCIONES** y se mostrara la siguiente pantalla.

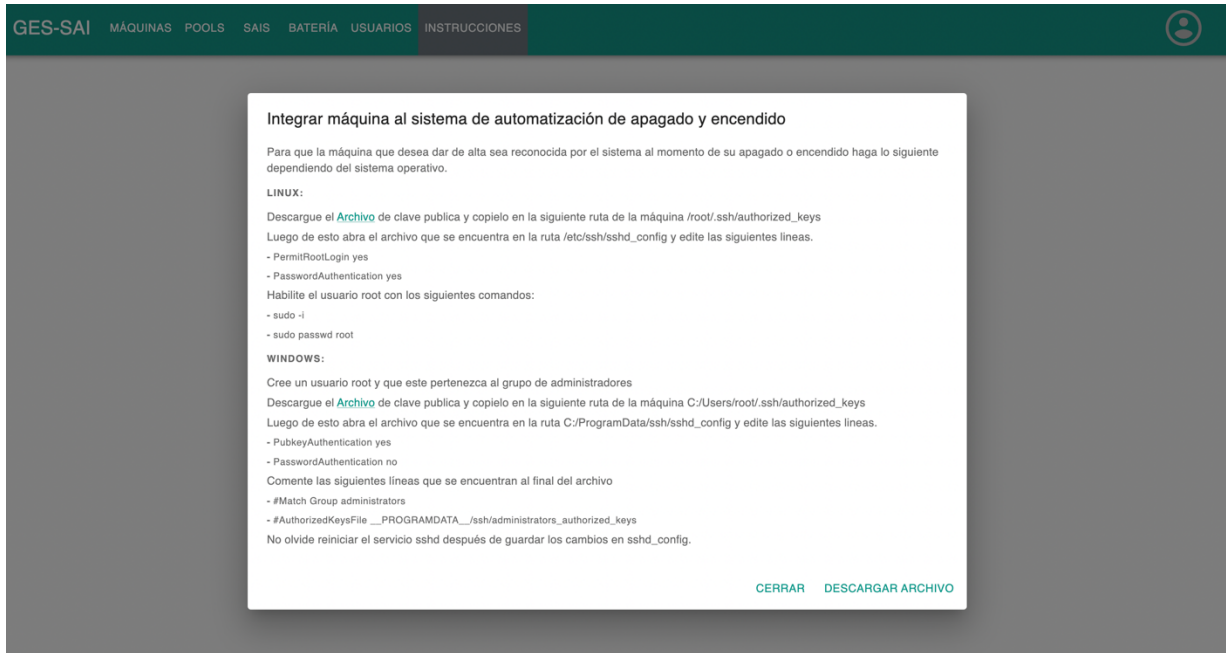


Imagen 40 - Módulo Instrucciones

De clic en el enlace “Archivo” o en el botón “Descargar archivo” para descargar la clave publica en su máquina y siga las instrucciones. Todo esto es con el fin de que la aplicación Ges-SAI tenga acceso a las máquinas dadas de alta y pueda apagarlas.