

***HERRAMIENTA PARA EL CONTROL DEL ACCESO A
RECURSOS WEB A TRAVÉS DEL NAVEGADOR EN
PUESTOS INFORMÁTICOS DURANTE LA
REALIZACIÓN DE EXÁMENES PARA LA EII***

**GRADO EN INGENIERÍA INFORMÁTICA DEL
SOFTWARE**

TRABAJO DE FIN DE GRADO

AUTOR

Andrés Casillas García

TUTORES

Luis Antonio Vinuesa Martínez

José Manuel Redondo López

Agradecimientos

En primer lugar, agradecer a Luis y a José la dedicación que han tenido con este trabajo fin de grado, habiendo estado reuniéndose conmigo incluso antes del comienzo oficial del trabajo además de haberme estado respondiendo emails en verano. También recalcar el esfuerzo que han realizado para intentar publicar este proyecto e incluso llevarlo a un concurso internacional, sumando a esto los ánimos que me han dado durante todo el proceso.

Por otro lado, quiero agradecer también a la Escuela de Ingeniería Informática de Oviedo las facilidades dadas para poner a prueba el proyecto e incluso implantarlo, aunque finalmente debido a la pandemia del COVID-19 no haya sido posible.

Para acabar, como no, agradecer a mis padres el esfuerzo que han hecho para permitirme llegar hasta aquí, además de los apoyos y la ayuda que me han proporcionado.

Índice de contenido

Capítulo 1 Introducción y análisis	9
Introducción	10
1.1. Motivación	10
1.2. Características ideales del sistema software	10
Análisis de alternativas	11
2.1. Descripción general	11
2.2. Alternativas del cliente	11
2.3. Alternativas del aplicativo servidor	13
2.4. Solución final	13
Análisis de mercado	14
3.1. Extensiones para Chrome existentes	14
3.2. Sistemas de seguridad vigentes	14
Prototipo a realizar	16
4.1. Introducción al prototipo	16
4.2. Planificación	16
4.3. Presupuesto	27
4.4. Requisitos funcionales	27
4.5. Requisitos no funcionales	28
4.6. Casos de uso	29
Capítulo 2 Diseño y arquitectura	33
Vista de bloques	34
1.1. Diagrama de caja negra	34
1.2. Diagrama de caja blanca 1	34
1.3. Diagrama de caja blanca 2	36
Modelo de dominio	40
Diagramas de secuencia	41
3.1. Rol student – Extensión de Chrome	41
3.2. Rol professor – Portal administración	43
3.3. Rol administrator – Portal de administración	45

Capítulo 3 Detalles de implementación	49
Extensión para Chrome	50
1.1. Descripción breve	50
1.2. Análisis de peticiones	50
1.3. Más detalles de implementación	53
Aplicativo NodeJS	55
2.1. Descripción breve	55
Capítulo 4 Procesos de calidad	57
Extensión para Chrome	58
1.1. Casos de test	58
1.2. Pruebas con usuarios en entorno reducido	63
Aplicativo NodeJS	64
2.1. Estructura facilitadora de pruebas	64
2.2. Casos de test	64
Pruebas con usuarios en entorno final	72
3.1. Descripción de la prueba	72
3.2. Análisis de resultados obtenidos	72
Capítulo 5 Medidas de seguridad.....	73
Medidas de seguridad globales.....	74
1.1. Políticas locales en los clientes	74
1.2. Configuración del servidor.....	75
Medidas de seguridad en la extensión	77
Medidas de seguridad en aplicativo NodeJS.....	79
3.1. Seguridad global	79
3.2. Seguridad en la API REST	79
3.3. Seguridad en el panel de administración	80
Capítulo 6 Proceso de despliegue	81
Situación ideal	82
1.1. Requisitos y configuración de los clientes	82
1.2. Requisitos y configuración del servidor	84
1.3. Proceso de actualización e instalación inicial.....	84
Restricciones del entorno final	86

2.1. Descripción de restricciones.....	86
2.2. Diferencias con despliegue ideal	86
Capítulo 7 Manual de usuario.....	87
Extensión para Chrome	88
1.1. Instalación.....	88
1.2. Guía de uso	88
API REST.....	91
2.1. Configuración.....	91
2.2. Guía de uso	91
Portal de administración	94
3.1. Configuración.....	94
3.2. Guía de uso – Aspectos comunes	94
3.3. Guía de uso – Rol administrador	95
3.4. Guía de uso – Rol profesor	102
Capítulo 8 Anexos	109
Mejoras a futuro	109
1.1. Mejoras funcionales	110
1.2. Mejoras de seguridad	110
1.3. Nuevos enfoques	111
Referencias.....	112

Capítulo 1 INTRODUCCIÓN Y ANÁLISIS



INTRODUCCIÓN

1.1. Motivación

Tras observar a través de las redes sociales diversas filtraciones de información, no intencionadas, realizadas por parte de trabajadores, incluso del sector público en entidades de especial cuidado como puede ser la agencia tributaria, me di realmente cuenta de que uno de los principales problemas que hay en el campo de la seguridad en los sistemas de procesamiento y tratamiento de la información es el factor humano.

Debido a este factor humano, y teniendo en cuenta que las filtraciones se debían a fotografías tomadas de un navegador en el que se indicaban puntos de acceso internos y privados, llegué a la conclusión de que debía buscarse la posibilidad de realizar una herramienta, que, aunque siguiera existiendo ese error humano permitiera minimizar los riesgos al mínimo.

1.2. Características ideales del sistema software

Después de un profundo análisis de las opciones que podría cubrir la herramienta, se llegó a la conclusión de que el primer punto que debiera cubrirse sería el de centralizar el control de todos los navegadores disponibles en los ordenadores de una organización, de tal forma que el administrador pudiera limitar las funciones del mismo en función de cada usuario. Este apartado por su importancia debe ser cubierto sí o sí por la solución escogida.

El siguiente paso se basa en determinar las funcionalidades ideales que debiera tener el sistema, siendo prescindibles algunas de ellas, en caso de que la solución escogida no pudiera resolverlas, no aportara grandes beneficios y/o el tiempo disponible no permitiera su realización. Son las siguientes:

- Analizar la URL
- Analizar el contenido de una web
- Ocultar la URL en la barra de direcciones
- Registrar las páginas visitadas junto a la fecha y la hora
- Controlar la gestión del historial, marcadores, configuración del navegador...
- Capturar el contenido que ve el usuario
- Obtener la localización del usuario
- Configurar un proxy que obligue a utilizar un determinado navegador para poder navegar
- Impedir el uso de dispositivos extraíbles
- Impedir el modo incógnito
- No registrar credenciales ni formularios
- Limpieza del navegador en cada inicio
- Adaptación de los filtros en función de una hora concreta
- Obtener y configurar determinadas características del ordenador destino

ANÁLISIS DE ALTERNATIVAS

2.1. Descripción general

Para solventar el problema descrito, está claro que lo principal es disponer de un aplicativo cliente-servidor, de forma que el comportamiento del cliente varíe en función de lo configurado de forma remota.

El sistema al tener que gestionar una gran cantidad de solicitudes, generalmente con poco esfuerzo computacional, necesita tener una capacidad multitarea muy elevada, de forma que los usuarios no se vean ralentizados por el uso de la nueva herramienta. También debe tener una gran versatilidad a la hora de ir evolucionándolo en el tiempo a las nuevas necesidades que pudieran ir surgiendo.

Y como requisito ineludible debe ser fácilmente implantable.

2.2. Alternativas del cliente

El aplicativo cliente al trabajar sobre un navegador da lugar a que puedan plantearse las siguientes dos alternativas:

- Realizar una extensión para un navegador ya existente
- Crear un nuevo navegador

Realizar una extensión, además de ser una tarea más liviana, permite beneficiarse de la madurez y los avances en materias de seguridad y funcionalidad que se implementan regularmente con las actualizaciones de los navegadores. Además, incorpora el hecho de que la gente ya deposita su confianza en ellos, y se crea o no, la gente es reacia a cambiar.

El problema que plantea esta solución es que debe buscarse un navegador para el cual realizarla, basándose en las limitaciones que nos impone cada uno, la tasa de uso, comprobar que nos permite hacer algo de lo que buscamos...

Dentro de este análisis se determinó por valorar el realizar una extensión para Google Chrome, ya que al estar basado en el proyecto Chromium permite que la extensión sea compatible con otros navegadores como Opera, Chromium y recientemente Edge de Microsoft entre otros. No obstante, aunque todos se basen sobre el mismo proyecto no todos incorporan todas las funcionalidades que ofrecen las extensiones de Google Chrome, sino que algunos como Opera no incorporan partes como la administración por directivas y el uso de la API management que permite controlar que los usuarios no desactiven/activen extensiones. Este hecho hace que nos decanemos por Google Chrome en exclusiva, ya que da mayor versatilidad y además es el más usado de entre los indicados, con el plus de que si los otros navegadores algún día incorporan esas otras características también será compatible para ellos y sin tener que realizar otra extensión de 0 adicional.

El realizar una extensión para Chrome nos permitiría realizar todas las funcionalidades descritas como ideales, a excepción de ocultar la URL en la barra de direcciones. Esto, no obstante, es de forma general, ya que hay opciones que solo se permiten si se utiliza Google Chrome en Chrome OS, como es el obtener y configurar determinadas características del ordenador destino y obtener la localización del usuario. Además, otra característica como es la de obtener el contenido de una web, está limitada, de forma que remotamente actualmente no se puede hacer, ya que requiere que en el archivo de configuración de la extensión se indiquen todas las páginas web existentes como analizables y eso haría que Google Chrome no nos permitiera alojar la extensión en su tienda, lo cual limita unas características en materia de seguridad que son muy interesantes, como es el hecho de que un usuario aunque acceda a la carpeta donde Chrome almacena las extensiones y modifique el código, al inicio el navegador compruebe la integridad de la misma para volver a descargarla de haber sido modificada, y esto solo se consigue si está alojada en su tienda de aplicaciones. No obstante, si se tiene la suscripción Google G Suite se pueden alojar extensiones privadas para una organización, lo que permitiría también añadir esta característica.

Por último, la segunda opción sería la de crear un navegador, lo cual no es una tarea fácil ni rápida, aunque se base en un motor de renderizado ya existente, en un objeto al estilo “WebView” de Android que permite generar un navegador personalizable mediante configuración de propiedades, a partir del mismo proyecto Chromium o ya por último el crear un navegador de 0.

Aunque las opciones del objeto al estilo “WebView” o basarse en el proyecto Chromium parecen bastante viables, hay que tener en cuenta una cantidad enorme de configuraciones que una persona que no tenga años de experiencia en la materia se dejaría la mitad por el camino. Esto se ha observado tras una pequeña prueba realizada sobre un “WebView” para tan solo subir y descargar ficheros, lo cual fue una tarea bastante compleja puesto que por defecto solo muestra HTML, prácticamente puro, sin tener en cuenta complementos como FlashPlayer, Java...

Otra característica decisiva, es el hecho de que habría que actualizar el navegador regularmente para solventar, mínimo, los problemas de seguridad que van surgiendo casi diariamente, y esto es una tarea irrealizable si no tienes un equipo de personas dedicado solamente a eso.

Como única ventaja respecto a la otra opción mencionada, es el hecho de que permitiría todas las configuraciones que quisiéramos y más, incluyendo el poder ocultar la barra de direcciones. No obstante, opciones como obtener y configurar determinadas características del ordenador destino y obtener la localización del usuario iban a estar limitadas por el sistema operativo.

Por todo lo anterior y teniendo en cuenta las personas involucradas en el proyecto, el tiempo de realización, la calidad de la solución y los pros y los contras de cada una, se opta por la primera opción planteada, realizar una extensión para Google Chrome.

2.3. Alternativas del aplicativo servidor

Con las necesidades planteadas en la descripción general, prácticamente se hace inevitable usar una de las tecnologías que más fuerza está cogiendo en los últimos tiempos en el campo del desarrollo web. Esta tecnología es NodeJS, la cual ofrece una cantidad más que suficiente de opciones como para realizar una solución al problema planteado. Además, combinado con el framework Express y el soporte con el gestor de dependencias NPM hace que su potencial aumente aún más, tanto en temas de seguridad como de funcionalidad.

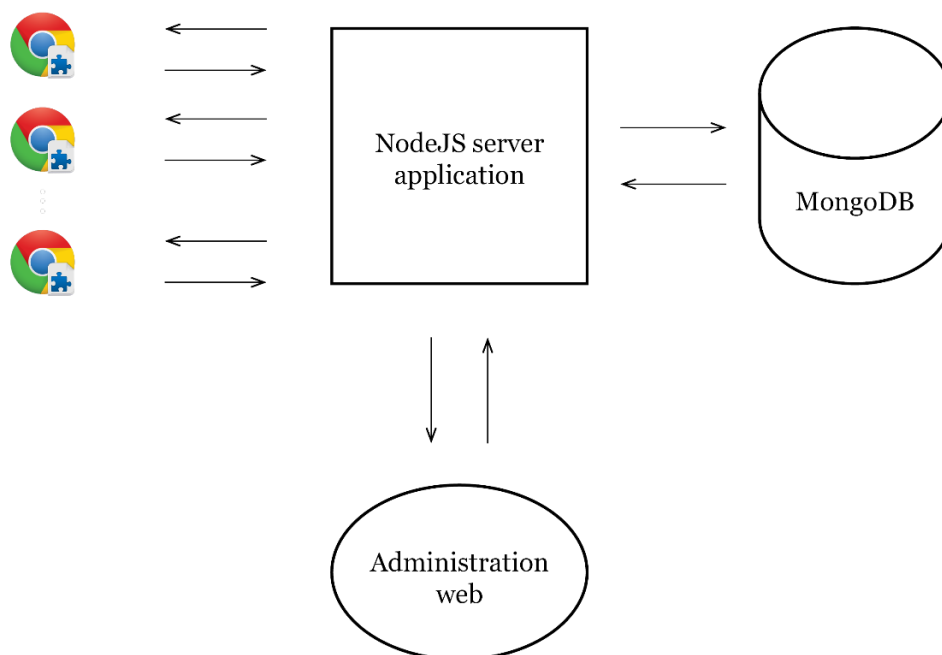
Tiene también como ventajas la agilidad que permite a la hora de realizar modificaciones y gestionar un proyecto de cero, ya que no requiere de realizar grandes configuraciones como sí pasaría con el framework Spring para Java.

Vinculado a NodeJS está la base de datos no relacional MongoDB, la cual va a ser usada también en el proyecto.

2.4. Solución final

Tras la elección de alternativas el sistema software completo quedaría basado en el lenguaje de programación JavaScript, estando formado por:

- La extensión para Google Chrome
- El aplicativo cliente-servidor basado en NodeJS que provee servicios REST y un panel de administración.



ANÁLISIS DE MERCADO

3.1. Extensiones para Chrome existentes

Actualmente en el mercado existen algunas extensiones para Google Chrome que tienen un fin similar al perseguido por el presente desarrollo, restringir el acceso a determinadas páginas web.

La diferencia principal que tienen es que no están centralizadas las restricciones, por lo que habría que ir navegador a navegador teniendo además las restricciones aplicadas a todos los usuarios por igual.

Otros déficits son el hecho de que no admiten analizar contenido, configurar el navegador, registrar acciones prohibidas... y en definitiva, están realizadas con otro propósito muy diferente al perseguido por el proyecto.

Las extensiones encontradas similares son las siguientes: TinyFilter PRO, WebFilter FREE, Tiny WebFilter, Parental Control, Block Site, Site Blocker y Dayboard.

Las 3 primeras, además de no estar actualizadas desde 2016, se venden como un filtro parental para evitar que los niños accedan a sitios pornográficos, de drogas... La siguiente, Parental Control, aunque actualizada, simplemente bloquea sitios web con pornografía. El resto se limitan a venderse como aplicaciones para evitar distracciones en tu tiempo de trabajo, bloqueando principalmente sitios web que contienen pornografía.

3.2. Sistemas de seguridad vigentes

En relación con filtros de contenido dentro de sistemas para la gestión de la información actualmente se emplean los proxys web.

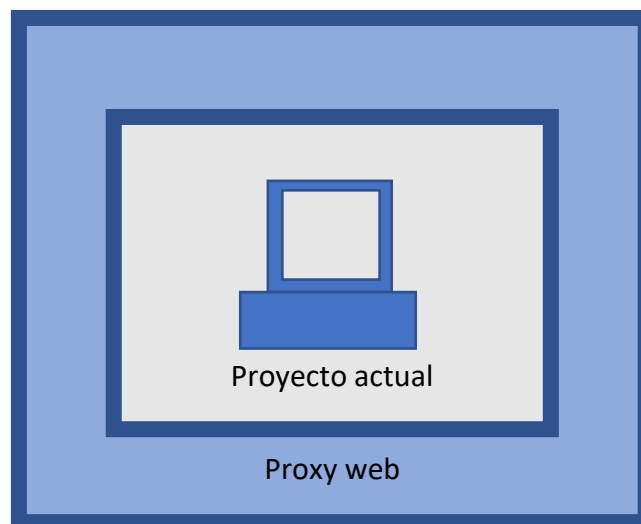
Esta opción tiene la desventaja de que funcionan de forma estática, lo que obliga a tener que cambiar las reglas que se quieran aplicar en cada momento de forma manual, a no ser que se implemente un script que permita cambiar las reglas de forma programada. No obstante, sigue estando el problema de que el filtro se aplica para todos los usuarios y no solo a uno en concreto, ya que no funcionan mediante un sistema de usuarios.

Otro problema, y quizás el más complejo al que se enfrenta esta protección hoy en día, es el uso de la encriptación en el contenido y los túneles cifrados que camuflan un protocolo de red dentro de otro que encripta la información. En primer lugar, el cifrado evita que se pueda leer el contenido, por lo que solamente se podrían filtrar las URL, y de segundas, en caso de camuflar un protocolo dentro de otro en el que la información viaja encriptada, como pasa con el protocolo SSH o alguno de los principales protocolos para conexiones VPN, haría que la petición real que se ha realizado, tan solo la sepa resolver el servidor que ya se encuentra fuera del alcance del proxy web. Un ejemplo de esto

sería el encontrarse la URL dentro del contenido cifrado, hecho que haría que ya ni la URL se pudiera analizar.

Por otro lado, cierto es que, aplicando una cantidad elevada de políticas de administración, como limitar la conexión al protocolo SSH o los utilizados en VPN a direcciones IP conocidas por la organización evitaría el uso anteriormente mencionado, pero ¿qué pasa cuando una organización se tiene que conectar a la VPN de otra la cual no aplica suficientes políticas o las que aplica para el contenido son diferentes a las tuyas? A través de esa conexión VPN insegura podría accederse a una información no deseada.

Es por todos los problemas anteriormente mencionados por lo que surge el presente proyecto, no obstante, aunque trata de solucionarlos no evita que alguien pueda encontrar una forma de saltarse esta nueva capa de seguridad. Por ello, el proyecto trata de ser una medida de refuerzo adicional a las ya existentes y no un sustituto en absoluto, porque por muchos muros que se pongan siempre va a ser posible romperlos todos. La única diferencia es que requerirá de más tiempo y esfuerzo para el atacante o posible infractor de una política interna de la organización.



PROTOTIPO A REALIZAR

4.1. Introducción al prototipo

Debido a que el sistema planteado es un proyecto de propósito general difícil de cubrir en su totalidad por un trabajo de fin de grado, se procede a desarrollar un prototipo funcional que permita comprobar que el sistema podría ser viable.

Para este prototipo se pensó en la posibilidad de realizar un control y limitar en tiempo real la navegación realizada por parte de los alumnos durante los exámenes en la facultad de Ingeniería Informática de la Universidad de Oviedo.

Principalmente se pretende evitar que los alumnos accedan a recursos no permitidos, notificándolo en caso de que incumplan la limitación.

Actualmente durante los exámenes se requiere a los alumnos de la desconexión de la toma de red para evitar que se acceda a internet. Este proceso además de hacer perder tiempo a la hora de iniciar un examen podría permitir en un descuido que un alumno acceda y descargue contenidos que le ayuden en la realización del examen, e incluso podrían permitirle enviar el examen y recibirlo resuelto una vez se han habilitado de nuevo las tomas de red para subir el examen resuelto al campus virtual.

4.2. Planificación

En relación con la naturaleza del proyecto, una investigación con posible realización de prototipo se opta por realizar una planificación basada en 9 grandes fases e incluyendo una estimación en días de la duración de cada fase suponiendo que la realiza una única persona, aunque de realizarse las tareas en paralelo no coincidiría con la duración total del proyecto. Además, dentro de cada fase se detallarán las principales tareas estimándolas con una puntuación según su complejidad temporal. Esta escala de puntos irá de 1 a 4 y la duración de cada tarea será la proporción de la duración total de la fase respecto a la puntuación de esa tarea y el total, es decir si la fase tiene duración 14 días y hay 4 tareas cuya complejidad temporal total es 12 puntos, la duración de la primera tarea estando estimada en 2 puntos es $\frac{2}{12} \times 14$ días, lo que resulta en poco más de 2 días.

Fase 1	Análisis	19 días
Tarea	Puntuación	
Analizar sistemas de seguridad y alternativas similares ya existentes	3	
Analizar requisitos técnicos y funcionales	4	
Analizar posibilidades de materializarlo (nuevo navegador, extensión...)	4	
Analizar que la opción elegida sea segura	4	
Analizar posibles roles de usuario necesarios	2	
Analizar costes, licencias y plataformas a utilizar	1	



Fase 2	Diseño	25 días
Tarea		Puntuación
Elección de tecnologías		4
Diseñar arquitectura		4
Definición final de roles de usuario		2
Diseño de entidades		2
Definición inicial de la documentación		1

Fase 3	Implementación de extensión para Chrome	70 días
Tarea		Puntuación
Aprendizaje de cómo funcionan las herramientas elegidas		4
Programación inicial para redirigir peticiones		3
Diseño de API de prueba para desarrollar el cliente		1
Mejora de la redirección de peticiones analizando alternativas		2
Interfaz gráfica de usuario mejorada		2
Gestión de descargas		3
Pruebas puntuales con usuarios de forma individual		1

Fase 4	Implementación de aplicativo NodeJS	60 días
Tarea		Puntuación
Esqueleto siguiendo la arquitectura		2
Gestión de conexiones a la base de datos		1
Gestión de peticiones realizadas por la extensión		3
Desarrollo de portal de administración para profesores		4
Desarrollo de portal de administración para administradores		4
Mejora de la interfaz gráfica de usuario		1
Pruebas puntuales con usuarios de forma individual		1

Fase 5	Administración del servidor web	15 días
Tarea		Puntuación
Elección de sistema operativo		1
Elegir herramientas y configuraciones a utilizar para asegurar el sistema		3
Desarrollar scripts que permitan configurar un servidor web		4
Probar scripts que permitan configurar servidor web		1
Solucionar problemas de los scripts		2
Preparar servidor de AWS		1
Documentar uso de scripts		1

Fase 6	Adaptar proyecto al entorno de despliegue	15 días
Tarea	Puntuación	
Añadir comprobación con LDAP de la universidad	3	
Adaptar formato de ficheros de carga	1	
Carga de datos mediante ficheros	4	
Definir reglas de seguridad de Chrome a aplicar en directorio activo	3	
Analizar cambios a realizar en los ordenadores del laboratorio de la EII	1	
Solicitud de permisos necesarios a la universidad	1	

Fase 7	Documentación	30 días
Tarea	Puntuación	
Definición final de la estructura de la documentación	1	
Desarrollo de los apartados de la documentación	4	

Fase 8	Despliegue en entorno final	5 días
Tarea	Puntuación	
Preparación del servidor	4	
Probar de forma individual el funcionamiento del aplicativo NodeJS	1	
Configurar ordenadores de los laboratorios	3	
Configurar directorio activo	2	
Probar el funcionamiento, de forma individual, del sistema completo	1	
Comprobar que el sistema sea seguro	2	

Fase 9	Pruebas con usuarios en entorno final	5 días
Tarea	Puntuación	
Elegir fecha para realizar la prueba con varios usuarios	1	
Solicitar permiso para realización de prueba con usuarios	1	
Desarrollo de la prueba con usuarios	2	
Descripción y análisis de los resultados obtenidos en la prueba	4	

Todas las estimaciones en días han sido realizadas en base a la duración que han necesitado otros proyectos personales de menor envergadura. Además, se indican las principales tareas por la dificultad de estimar más en detalle las tareas concretas que serán necesarias.

La fase 9 y algunas tareas de la fase 8 finalmente no fue posible llevarlas a cabo debido al COVID-19, no obstante, el resto de las tareas se han ajustado bastante bien con el tiempo y esfuerzo estimado.

Se muestra a continuación un diagrama de Grantt con la organización de tareas en el tiempo en base a los datos indicados anteriormente y teniendo en cuenta la existencia de los siguientes 7 roles en el equipo de desarrollo y cada rol representado por una única persona:



- Jefe de proyecto
- Analista de software
- Arquitecto de software
- Desarrollador
- Testeador
- Administrador de sistemas
- Experto en seguridad

Actividad	Día inicio	Día fin	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Analizar sistemas de seguridad y alternativas similares ya existentes	1	3																					
Analizar requisitos técnicos y funcionales	4	8																					
Analizar posibilidades de materializarlo (nuevo navegador...)	9	12																					
Analizar que la opción elegida sea segura	13	16																					
Analizar posibles roles de usuario necesarios	17	18																					
Analizar costes, licencias y plataformas a utilizar	13	13																					
Elección de tecnologías	19	26																					
Diseñar arquitectura	27	34																					
Definición final de roles de usuario	19	21																					
Diseño de entidades	35	38																					
Definición inicial de la documentación	39	40																					
Aprendizaje de cómo funcionan las herramientas elegidas	27	44																					
Programación inicial para redirigir peticiones	45	61																					
Diseño de API de prueba para desarrollar el cliente	49	52																					
Mejora de la redirección de peticiones analizando alternativas	62	70																					
Interfaz gráfica de usuario mejorada	71	79																					
Gestión de descargas	80	92																					
Pruebas puntuales con usuarios de forma individual	62	65																					
Esqueleto siguiendo la arquitectura	93	100																					
Gestión de conexiones a la base de datos	101	104																					
Gestión de peticiones realizadas por la extensión	105	115																					
Desarrollo de portal de administración para profesores	116	130																					
Desarrollo de portal de administración para administradores	131	145																					
Mejora de la interfaz gráfica de usuario	146	149																					
Pruebas puntuales con usuarios de forma individual	116	119																					
Elección de sistema operativo	27	27																					
Elegir herramientas y configuraciones a utilizar para asegurar el sistema	28	31																					
Desarrollar scripts que permitan configurar un servidor web	32	36																					
Probar scripts que permitan configurar servidor web	37	37																					
Solucionar problemas de los scripts	38	39																					
Preparar servidor de AWS	40	40																					
Documentar uso de scripts	41	41																					
Añadir comprobación con LDAP de la universidad	150	153																					
Adaptar formato de ficheros de carga	154	154																					
Carga de datos mediante ficheros	155	159																					
Definir reglas de seguridad de Chrome a aplicar en directorio activo	32	34																					
Analizar cambios a realizar en los ordenadores del laboratorio de la EII	42	42																					
Solicitud de permisos necesarios a la universidad	41	41																					
Definición final de la estructura de la documentación	42	46																					
Desarrollo de los apartados de documentación	48	169																					
Preparación del servidor	160	161																					
Probar de forma individual el funcionamiento del aplicativo NodeJS	162	162																					
Configurar ordenadores de los laboratorios	163	163																					
Configurar directorio activo	164	164																					
Probar el funcionamiento, de forma individual, del sistema completo	165	165																					
Comprobar que el sistema sea seguro	166	166																					
Elegir fecha para realizar la prueba con varios usuarios	155	155																					
Solicitar permiso para realización de prueba con usuarios	156	156																					
Desarrollo de la prueba con usuarios	167	167																					
Descripción y análisis de los resultados obtenidos en la prueba	168	169																					



Actividad	Día inicio	Día fin	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42
Analizar sistemas de seguridad y alternativas similares ya existentes	1	3																					
Analizar requisitos técnicos y funcionales	4	8																					
Analizar posibilidades de materializarlo (nuevo navegador...)	9	12																					
Analizar que la opción elegida sea segura	13	16																					
Analizar posibles roles de usuario necesarios	17	18																					
Analizar costes, licencias y plataformas a utilizar	13	13																					
Elección de tecnologías	19	26																					
Diseñar arquitectura	27	34																					
Definición final de roles de usuario	19	21																					
Diseño de entidades	35	38																					
Definición inicial de la documentación	39	40																					
Aprendizaje de cómo funcionan las herramientas elegidas	27	44																					
Programación inicial para redirigir peticiones	45	61																					
Diseño de API de prueba para desarrollar el cliente	49	52																					
Mejora de la redirección de peticiones analizando alternativas	62	70																					
Interfaz gráfica de usuario mejorada	71	79																					
Gestión de descargas	80	92																					
Pruebas puntuales con usuarios de forma individual	62	65																					
Esqueleto siguiendo la arquitectura	93	100																					
Gestión de conexiones a la base de datos	101	104																					
Gestión de peticiones realizadas por la extensión	105	115																					
Desarrollo de portal de administración para profesores	116	130																					
Desarrollo de portal de administración para administradores	131	145																					
Mejora de la interfaz gráfica de usuario	146	149																					
Pruebas puntuales con usuarios de forma individual	116	119																					
Elección de sistema operativo	27	27																					
Elegir herramientas y configuraciones a utilizar para asegurar el sistema	28	31																					
Desarrollar scripts que permitan configurar un servidor web	32	36																					
Probar scripts que permitan configurar servidor web	37	37																					
Solucionar problemas de los scripts	38	39																					
Preparar servidor de AWS	40	40																					
Documentar uso de scripts	41	41																					
Añadir comprobación con LDAP de la universidad	150	153																					
Adaptar formato de ficheros de carga	154	154																					
Carga de datos mediante ficheros	155	159																					
Definir reglas de seguridad de Chrome a aplicar en directorio activo	32	34																					
Analizar cambios a realizar en los ordenadores del laboratorio de la EII	42	42																					
Solicitud de permisos necesarios a la universidad	41	41																					
Definición final de la estructura de la documentación	42	46																					
Desarrollo de los apartados de documentación	48	169																					
Preparación del servidor	160	161																					
Probar de forma individual el funcionamiento del aplicativo NodeJS	162	162																					
Configurar ordenadores de los laboratorios	163	163																					
Configurar directorio activo	164	164																					
Probar el funcionamiento, de forma individual, del sistema completo	165	165																					
Comprobar que el sistema sea seguro	166	166																					
Elegir fecha para realizar la prueba con varios usuarios	155	155																					
Solicitar permiso para realización de prueba con usuarios	156	156																					
Desarrollo de la prueba con usuarios	167	167																					
Descripción y análisis de los resultados obtenidos en la prueba	168	169																					



Actividad	Día inicio	Día fin	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
Analizar sistemas de seguridad y alternativas similares ya existentes	1	3																					
Analizar requisitos técnicos y funcionales	4	8																					
Analizar posibilidades de materializarlo (nuevo navegador...)	9	12																					
Analizar que la opción elegida sea segura	13	16																					
Analizar posibles roles de usuario necesarios	17	18																					
Analizar costes, licencias y plataformas a utilizar	13	13																					
Elección de tecnologías	19	26																					
Diseñar arquitectura	27	34																					
Definición final de roles de usuario	19	21																					
Diseño de entidades	35	38																					
Definición inicial de la documentación	39	40																					
Aprendizaje de cómo funcionan las herramientas elegidas	27	44																					
Programación inicial para redirigir peticiones	45	61																					
Diseño de API de prueba para desarrollar el cliente	49	52																					
Mejora de la redirección de peticiones analizando alternativas	62	70																					
Interfaz gráfica de usuario mejorada	71	79																					
Gestión de descargas	80	92																					
Pruebas puntuales con usuarios de forma individual	62	65																					
Esqueleto siguiendo la arquitectura	93	100																					
Gestión de conexiones a la base de datos	101	104																					
Gestión de peticiones realizadas por la extensión	105	115																					
Desarrollo de portal de administración para profesores	116	130																					
Desarrollo de portal de administración para administradores	131	145																					
Mejora de la interfaz gráfica de usuario	146	149																					
Pruebas puntuales con usuarios de forma individual	116	119																					
Elección de sistema operativo	27	27																					
Elegir herramientas y configuraciones a utilizar para asegurar el sistema	28	31																					
Desarrollar scripts que permitan configurar un servidor web	32	36																					
Probar scripts que permitan configurar servidor web	37	37																					
Solucionar problemas de los scripts	38	39																					
Preparar servidor de AWS	40	40																					
Documentar uso de scripts	41	41																					
Añadir comprobación con LDAP de la universidad	150	153																					
Adaptar formato de ficheros de carga	154	154																					
Carga de datos mediante ficheros	155	159																					
Definir reglas de seguridad de Chrome a aplicar en directorio activo	32	34																					
Analizar cambios a realizar en los ordenadores del laboratorio de la EII	42	42																					
Solicitud de permisos necesarios a la universidad	41	41																					
Definición final de la estructura de la documentación	42	46																					
Desarrollo de los apartados de documentación	48	169																					
Preparación del servidor	160	161																					
Probar de forma individual el funcionamiento del aplicativo NodeJS	162	162																					
Configurar ordenadores de los laboratorios	163	163																					
Configurar directorio activo	164	164																					
Probar el funcionamiento, de forma individual, del sistema completo	165	165																					
Comprobar que el sistema sea seguro	166	166																					
Elegir fecha para realizar la prueba con varios usuarios	155	155																					
Solicitar permiso para realización de prueba con usuarios	156	156																					
Desarrollo de la prueba con usuarios	167	167																					
Descripción y análisis de los resultados obtenidos en la prueba	168	169																					



Actividad	Día inicio	Día fin	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84
Analizar sistemas de seguridad y alternativas similares ya existentes	1	3																					
Analizar requisitos técnicos y funcionales	4	8																					
Analizar posibilidades de materializarlo (nuevo navegador...)	9	12																					
Analizar que la opción elegida sea segura	13	16																					
Analizar posibles roles de usuario necesarios	17	18																					
Analizar costes, licencias y plataformas a utilizar	13	13																					
Elección de tecnologías	19	26																					
Diseñar arquitectura	27	34																					
Definición final de roles de usuario	19	21																					
Diseño de entidades	35	38																					
Definición inicial de la documentación	39	40																					
Aprendizaje de cómo funcionan las herramientas elegidas	27	44																					
Programación inicial para redirigir peticiones	45	61																					
Diseño de API de prueba para desarrollar el cliente	49	52																					
Mejora de la redirección de peticiones analizando alternativas	62	70																					
Interfaz gráfica de usuario mejorada	71	79																					
Gestión de descargas	80	92																					
Pruebas puntuales con usuarios de forma individual	62	65																					
Esqueleto siguiendo la arquitectura	93	100																					
Gestión de conexiones a la base de datos	101	104																					
Gestión de peticiones realizadas por la extensión	105	115																					
Desarrollo de portal de administración para profesores	116	130																					
Desarrollo de portal de administración para administradores	131	145																					
Mejora de la interfaz gráfica de usuario	146	149																					
Pruebas puntuales con usuarios de forma individual	116	119																					
Elección de sistema operativo	27	27																					
Elegir herramientas y configuraciones a utilizar para asegurar el sistema	28	31																					
Desarrollar scripts que permitan configurar un servidor web	32	36																					
Probar scripts que permitan configurar servidor web	37	37																					
Solucionar problemas de los scripts	38	39																					
Preparar servidor de AWS	40	40																					
Documentar uso de scripts	41	41																					
Añadir comprobación con LDAP de la universidad	150	153																					
Adaptar formato de ficheros de carga	154	154																					
Carga de datos mediante ficheros	155	159																					
Definir reglas de seguridad de Chrome a aplicar en directorio activo	32	34																					
Analizar cambios a realizar en los ordenadores del laboratorio de la EII	42	42																					
Solicitud de permisos necesarios a la universidad	41	41																					
Definición final de la estructura de la documentación	42	46																					
Desarrollo de los apartados de documentación	48	169																					
Preparación del servidor	160	161																					
Probar de forma individual el funcionamiento del aplicativo NodeJS	162	162																					
Configurar ordenadores de los laboratorios	163	163																					
Configurar directorio activo	164	164																					
Probar el funcionamiento, de forma individual, del sistema completo	165	165																					
Comprobar que el sistema sea seguro	166	166																					
Elegir fecha para realizar la prueba con varios usuarios	155	155																					
Solicitar permiso para realización de prueba con usuarios	156	156																					
Desarrollo de la prueba con usuarios	167	167																					
Descripción y análisis de los resultados obtenidos en la prueba	168	169																					



Actividad	Día inicio	Día fin	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100	101	102	103	104	105
Analizar sistemas de seguridad y alternativas similares ya existentes	1	3																					
Analizar requisitos técnicos y funcionales	4	8																					
Analizar posibilidades de materializarlo (nuevo navegador...)	9	12																					
Analizar que la opción elegida sea segura	13	16																					
Analizar posibles roles de usuario necesarios	17	18																					
Analizar costes, licencias y plataformas a utilizar	13	13																					
Elección de tecnologías	19	26																					
Diseñar arquitectura	27	34																					
Definición final de roles de usuario	19	21																					
Diseño de entidades	35	38																					
Definición inicial de la documentación	39	40																					
Aprendizaje de cómo funcionan las herramientas elegidas	27	44																					
Programación inicial para redirigir peticiones	45	61																					
Diseño de API de prueba para desarrollar el cliente	49	52																					
Mejora de la redirección de peticiones analizando alternativas	62	70																					
Interfaz gráfica de usuario mejorada	71	79																					
Gestión de descargas	80	92																					
Pruebas puntuales con usuarios de forma individual	62	65																					
Esqueleto siguiendo la arquitectura	93	100																					
Gestión de conexiones a la base de datos	101	104																					
Gestión de peticiones realizadas por la extensión	105	115																					
Desarrollo de portal de administración para profesores	116	130																					
Desarrollo de portal de administración para administradores	131	145																					
Mejora de la interfaz gráfica de usuario	146	149																					
Pruebas puntuales con usuarios de forma individual	116	119																					
Elección de sistema operativo	27	27																					
Elegir herramientas y configuraciones a utilizar para asegurar el sistema	28	31																					
Desarrollar scripts que permitan configurar un servidor web	32	36																					
Probar scripts que permitan configurar servidor web	37	37																					
Solucionar problemas de los scripts	38	39																					
Preparar servidor de AWS	40	40																					
Documentar uso de scripts	41	41																					
Añadir comprobación con LDAP de la universidad	150	153																					
Adaptar formato de ficheros de carga	154	154																					
Carga de datos mediante ficheros	155	159																					
Definir reglas de seguridad de Chrome a aplicar en directorio activo	32	34																					
Analizar cambios a realizar en los ordenadores del laboratorio de la EII	42	42																					
Solicitud de permisos necesarios a la universidad	41	41																					
Definición final de la estructura de la documentación	42	46																					
Desarrollo de los apartados de documentación	48	169																					
Preparación del servidor	160	161																					
Probar de forma individual el funcionamiento del aplicativo NodeJS	162	162																					
Configurar ordenadores de los laboratorios	163	163																					
Configurar directorio activo	164	164																					
Probar el funcionamiento, de forma individual, del sistema completo	165	165																					
Comprobar que el sistema sea seguro	166	166																					
Elegir fecha para realizar la prueba con varios usuarios	155	155																					
Solicitar permiso para realización de prueba con usuarios	156	156																					
Desarrollo de la prueba con usuarios	167	167																					
Descripción y análisis de los resultados obtenidos en la prueba	168	169																					



Actividad	Día inicio	Día fin	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126
Analizar sistemas de seguridad y alternativas similares ya existentes	1	3																					
Analizar requisitos técnicos y funcionales	4	8																					
Analizar posibilidades de materializarlo (nuevo navegador...)	9	12																					
Analizar que la opción elegida sea segura	13	16																					
Analizar posibles roles de usuario necesarios	17	18																					
Analizar costes, licencias y plataformas a utilizar	13	13																					
Elección de tecnologías	19	26																					
Diseñar arquitectura	27	34																					
Definición final de roles de usuario	19	21																					
Diseño de entidades	35	38																					
Definición inicial de la documentación	39	40																					
Aprendizaje de cómo funcionan las herramientas elegidas	27	44																					
Programación inicial para redirigir peticiones	45	61																					
Diseño de API de prueba para desarrollar el cliente	49	52																					
Mejora de la redirección de peticiones analizando alternativas	62	70																					
Interfaz gráfica de usuario mejorada	71	79																					
Gestión de descargas	80	92																					
Pruebas puntuales con usuarios de forma individual	62	65																					
Esqueleto siguiendo la arquitectura	93	100																					
Gestión de conexiones a la base de datos	101	104																					
Gestión de peticiones realizadas por la extensión	105	115																					
Desarrollo de portal de administración para profesores	116	130																					
Desarrollo de portal de administración para administradores	131	145																					
Mejora de la interfaz gráfica de usuario	146	149																					
Pruebas puntuales con usuarios de forma individual	116	119																					
Elección de sistema operativo	27	27																					
Elegir herramientas y configuraciones a utilizar para asegurar el sistema	28	31																					
Desarrollar scripts que permitan configurar un servidor web	32	36																					
Probar scripts que permitan configurar servidor web	37	37																					
Solucionar problemas de los scripts	38	39																					
Preparar servidor de AWS	40	40																					
Documentar uso de scripts	41	41																					
Añadir comprobación con LDAP de la universidad	150	153																					
Adaptar formato de ficheros de carga	154	154																					
Carga de datos mediante ficheros	155	159																					
Definir reglas de seguridad de Chrome a aplicar en directorio activo	32	34																					
Analizar cambios a realizar en los ordenadores del laboratorio de la EII	42	42																					
Solicitud de permisos necesarios a la universidad	41	41																					
Definición final de la estructura de la documentación	42	46																					
Desarrollo de los apartados de documentación	48	169																					
Preparación del servidor	160	161																					
Probar de forma individual el funcionamiento del aplicativo NodeJS	162	162																					
Configurar ordenadores de los laboratorios	163	163																					
Configurar directorio activo	164	164																					
Probar el funcionamiento, de forma individual, del sistema completo	165	165																					
Comprobar que el sistema sea seguro	166	166																					
Elegir fecha para realizar la prueba con varios usuarios	155	155																					
Solicitar permiso para realización de prueba con usuarios	156	156																					
Desarrollo de la prueba con usuarios	167	167																					
Descripción y análisis de los resultados obtenidos en la prueba	168	169																					



Actividad	Día inicio	Día fin	127	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144	145	146	147
Analizar sistemas de seguridad y alternativas similares ya existentes	1	3																					
Analizar requisitos técnicos y funcionales	4	8																					
Analizar posibilidades de materializarlo (nuevo navegador...)	9	12																					
Analizar que la opción elegida sea segura	13	16																					
Analizar posibles roles de usuario necesarios	17	18																					
Analizar costes, licencias y plataformas a utilizar	13	13																					
Elección de tecnologías	19	26																					
Diseñar arquitectura	27	34																					
Definición final de roles de usuario	19	21																					
Diseño de entidades	35	38																					
Definición inicial de la documentación	39	40																					
Aprendizaje de cómo funcionan las herramientas elegidas	27	44																					
Programación inicial para redirigir peticiones	45	61																					
Diseño de API de prueba para desarrollar el cliente	49	52																					
Mejora de la redirección de peticiones analizando alternativas	62	70																					
Interfaz gráfica de usuario mejorada	71	79																					
Gestión de descargas	80	92																					
Pruebas puntuales con usuarios de forma individual	62	65																					
Esqueleto siguiendo la arquitectura	93	100																					
Gestión de conexiones a la base de datos	101	104																					
Gestión de peticiones realizadas por la extensión	105	115																					
Desarrollo de portal de administración para profesores	116	130																					
Desarrollo de portal de administración para administradores	131	145																					
Mejora de la interfaz gráfica de usuario	146	149																					
Pruebas puntuales con usuarios de forma individual	116	119																					
Elección de sistema operativo	27	27																					
Elegir herramientas y configuraciones a utilizar para asegurar el sistema	28	31																					
Desarrollar scripts que permitan configurar un servidor web	32	36																					
Probar scripts que permitan configurar servidor web	37	37																					
Solucionar problemas de los scripts	38	39																					
Preparar servidor de AWS	40	40																					
Documentar uso de scripts	41	41																					
Añadir comprobación con LDAP de la universidad	150	153																					
Adaptar formato de ficheros de carga	154	154																					
Carga de datos mediante ficheros	155	159																					
Definir reglas de seguridad de Chrome a aplicar en directorio activo	32	34																					
Analizar cambios a realizar en los ordenadores del laboratorio de la EII	42	42																					
Solicitud de permisos necesarios a la universidad	41	41																					
Definición final de la estructura de la documentación	42	46																					
Desarrollo de los apartados de documentación	48	169																					
Preparación del servidor	160	161																					
Probar de forma individual el funcionamiento del aplicativo NodeJS	162	162																					
Configurar ordenadores de los laboratorios	163	163																					
Configurar directorio activo	164	164																					
Probar el funcionamiento, de forma individual, del sistema completo	165	165																					
Comprobar que el sistema sea seguro	166	166																					
Elegir fecha para realizar la prueba con varios usuarios	155	155																					
Solicitar permiso para realización de prueba con usuarios	156	156																					
Desarrollo de la prueba con usuarios	167	167																					
Descripción y análisis de los resultados obtenidos en la prueba	168	169																					



Actividad	Día inicio	Día fin	148	149	150	151	152	153	154	155	156	157	158	159	160	161	162	163	164	165	166	167	168	169
Analizar sistemas de seguridad y alternativas similares ya existentes	1	3																						
Analizar requisitos técnicos y funcionales	4	8																						
Analizar posibilidades de materializarlo (nuevo navegador...)	9	12																						
Analizar que la opción elegida sea segura	13	16																						
Analizar posibles roles de usuario necesarios	17	18																						
Analizar costes, licencias y plataformas a utilizar	13	13																						
Elección de tecnologías	19	26																						
Diseñar arquitectura	27	34																						
Definición final de roles de usuario	19	21																						
Diseño de entidades	35	38																						
Definición inicial de la documentación	39	40																						
Aprendizaje de cómo funcionan las herramientas elegidas	27	44																						
Programación inicial para redirigir peticiones	45	61																						
Diseño de API de prueba para desarrollar el cliente	49	52																						
Mejora de la redirección de peticiones analizando alternativas	62	70																						
Interfaz gráfica de usuario mejorada	71	79																						
Gestión de descargas	80	92																						
Pruebas puntuales con usuarios de forma individual	62	65																						
Esqueleto siguiendo la arquitectura	93	100																						
Gestión de conexiones a la base de datos	101	104																						
Gestión de peticiones realizadas por la extensión	105	115																						
Desarrollo de portal de administración para profesores	116	130																						
Desarrollo de portal de administración para administradores	131	145																						
Mejora de la interfaz gráfica de usuario	146	149																						
Pruebas puntuales con usuarios de forma individual	116	119																						
Elección de sistema operativo	27	27																						
Elegir herramientas y configuraciones a utilizar para asegurar el sistema	28	31																						
Desarrollar scripts que permitan configurar un servidor web	32	36																						
Probar scripts que permitan configurar servidor web	37	37																						
Solucionar problemas de los scripts	38	39																						
Preparar servidor de AWS	40	40																						
Documentar uso de scripts	41	41																						
Añadir comprobación con LDAP de la universidad	150	153																						
Adaptar formato de ficheros de carga	154	154																						
Carga de datos mediante ficheros	155	159																						
Definir reglas de seguridad de Chrome a aplicar en directorio activo	32	34																						
Analizar cambios a realizar en los ordenadores del laboratorio de la EII	42	42																						
Solicitud de permisos necesarios a la universidad	41	41																						
Definición final de la estructura de la documentación	42	46																						
Desarrollo de los apartados de documentación	48	169																						
Preparación del servidor	160	161																						
Probar de forma individual el funcionamiento del aplicativo NodeJS	162	162																						
Configurar ordenadores de los laboratorios	163	163																						
Configurar directorio activo	164	164																						
Probar el funcionamiento, de forma individual, del sistema completo	165	165																						
Comprobar que el sistema sea seguro	166	166																						
Elegir fecha para realizar la prueba con varios usuarios	155	155																						
Solicitar permiso para realización de prueba con usuarios	156	156																						
Desarrollo de la prueba con usuarios	167	167																						
Descripción y análisis de los resultados obtenidos en la prueba	168	169																						

4.3. Presupuesto

El coste de cada concepto está expresado en lo que se paga por día de trabajo y/o uso en base a los salarios observados en glassdoor e indeed, y a los costes de AWS para el servidor.

Unidad	Descripción	Precio
Día	Jefe de proyecto	112,63€
Día	Analista de software	67,32€
Día	Arquitecto de software	116,85€
Día	Desarrollador	55,95€
Día	Tester de software	59,99€
Día	Administrador de sistemas	48,64€
Día	Experto en seguridad	93,93€
Día	Servidor AWS T3.XLARGE	4,32€

En base a los precios anteriores y a los datos de la planificación se calcularán los precios estimados para el total del proyecto. Se tendrá en cuenta, además, que el jefe de proyecto, al ser el encargado de supervisar el curso del mismo, tendrá 1 día más cada 6 de proyecto, además de los indicados en la planificación para cubrir la supervisión y solución de problemas, así como el trato con el cliente.

Días	Descripción	Coste por día	Total
63	Jefe de proyecto	112,63€	7095,69€
10	Analista de software	67,32€	673,20€
20	Arquitecto de software	116,85€	2337,00€
133	Desarrollador	55,95€	7441,35€
13	Tester de software	59,99€	779,87€
16	Administrador de sistemas	48,64€	778,24€
20	Experto en seguridad	93,93€	1878,60€
169	Servidor AWS T3.XLARGE	4,32€	730,08€
TOTAL			21.714,03€

4.4. Requisitos funcionales

Al basarse en la idea general del proyecto, se trata de un sistema centralizado en el cual el profesor podrá imponer restricciones, basadas en listas blancas o negras, sobre los dominios, que no sean direcciones IP, a los que puede acceder un alumno durante un horario especificado. También podrá ver las infracciones cometidas por los mismos y si han iniciado sesión o no, lo que permitirá llevar un control de asistencia y uso del navegador.

También habrá administradores, que serán cuentas pertenecientes a la dirección de la escuela, los cuales podrán cargar la estructura de asignaturas-grupos-profesores-alumnos realizándose siempre un backup al mismo tiempo, además de poder visualizar notificaciones realizadas fuera de una

restricción, poder ver las últimas señales de vida de las extensiones para llevar un control de si han dejado o no de funcionar, gestionar la restauración / eliminado de copias de seguridad y configurar el tiempo que tardan en borrarse las señales de vida.

Los alumnos solo podrán acceder a los recursos permitidos por los profesores en función de la política marcada para un determinado alumno en una franja horaria concreta, siendo imprescindible que siempre que en un día no haya ninguna restricción programada la extensión no pida usuario ni contraseña de forma que el alumno no perciba ni que la extensión está ahí.

Al alumno se le mostrarán notificaciones de cuando comienza y/o finaliza una determinada restricción, además de las correspondientes indicaciones que eviten que por desconocimiento cometa una infracción.

Los datos de identificación utilizados por los alumnos, profesores y administradores para acceder al sistema serán los mismos a los utilizados en otros servicios de la Universidad de Oviedo, utilizando para ello el LDAP de la universidad. Siendo para los alumnos el usuario UOXXXXXX y para el resto usuario@uniovi.es. Además, en la extensión solamente podrán identificarse los alumnos y en el panel de administración solo profesores y administradores.

Un profesor tan solo podrá imponer limitaciones sobre los alumnos de una asignatura en la cual sea docente, siendo además la única forma de que acceda a las infracciones de un alumno, sino no tendrá acceso.

Al generar restricciones se corre el riesgo de que un alumno pueda tener ya otras en un horario determinado. Para evitar este conflicto, se excluirá automáticamente al alumno de la restricción y se le indicará al profesor. No obstante, esto no implica que la restricción no se vaya a crear, sino que ese alumno simplemente no estará. También ha de tenerse en cuenta que de tener todos los alumnos un conflicto en ese horario, la restricción como es lógico no se creará.

Las restricciones podrán modificarse y/o eliminarse siempre que no hayan finalizado ya, lo que hará que en el caso de eliminarse se eliminen también las notificaciones asociadas. En el caso de modificaciones dentro de las cuales un alumno pase a estar excluido, las notificaciones vinculadas a la restricción y al alumno, de haberlas, no se volverán a mostrar a no ser que se vuelva a incluir al alumno dentro de la restricción.

Los alumnos tan solo pertenecen a un grupo de una asignatura, nunca a varios. Siendo solamente relevantes los grupos de laboratorio.

4.5. Requisitos no funcionales

Se implementarán todas las medidas de seguridad necesarias para evitar que el sistema pueda ser evitado y/o manipulado teniendo en cuenta una gestión por roles basada en estudiantes, profesores y alumnos. En este sentido las notificaciones deberán contener la máxima información posible que permita identificar un ordenador en concreto dentro de la organización, así como detectar posibles

errores/manipulaciones en las mismas. En este sentido también debe asegurarse la comunicación, así como el servidor web en términos de confidencialidad, integridad y disponibilidad.

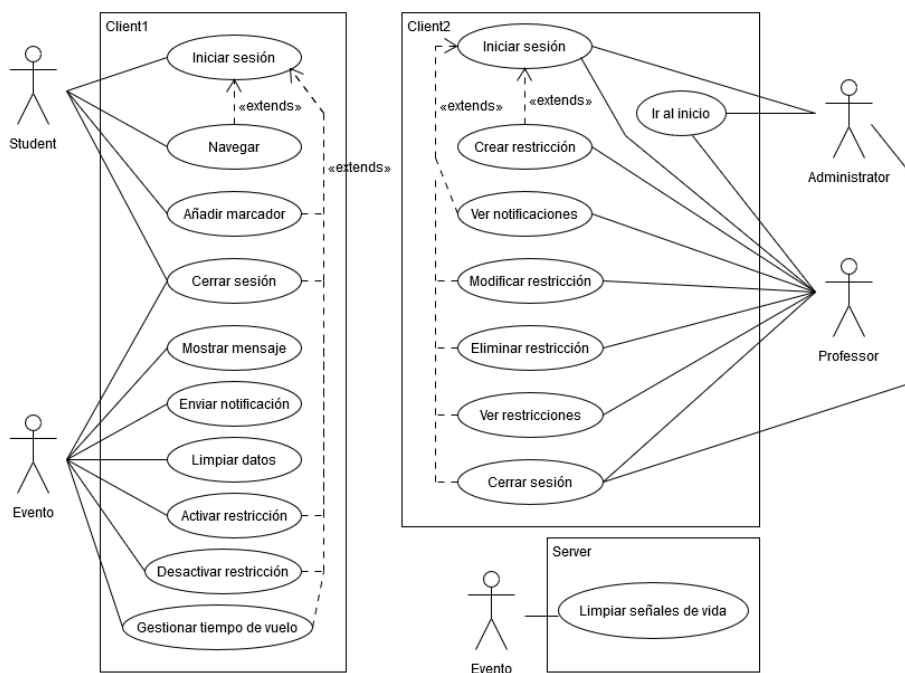
La forma de uso de la extensión, por parte de los alumnos, debe ser sencilla e intuitiva de forma que no requiera de mucha experiencia por parte del usuario inexperto. De igual forma, el panel de administración debe tener una interfaz gráfica de usuario que cumpla estos mismos criterios y a su vez permita realizar las acciones necesarias con un número de clics reducido.

Teniendo en cuenta el entorno donde va a ser desplegado, el prototipo debe ejecutarse de forma que se minimicen los recursos necesarios y permita una adecuada fluidez en su uso. En este sentido, debe asegurarse que el sistema provee un número de conexiones simultáneas suficientes como para cubrir la realización de un examen de laboratorio de la escuela. Relativo a la minimización de recursos necesarios debe tenerse en cuenta el espacio limitado de almacenamiento, dejando almacenados tan solo aquellos datos imprescindibles para el correcto funcionamiento del sistema, así como programando limpiezas periódicas de datos, que, pasados un tiempo, dejan de tener utilidad alguna.

Al igual que en todos los proyectos, ha de tenerse en cuenta el coste del mismo. En este caso, el coste es el relativo a las infraestructuras necesarias para proceder al despliegue. Dicho despliegue debe poder realizarse, dentro de lo posible, con los recursos propios de la universidad sin tener que adquirir material y/o licencias.

4.6. Casos de uso

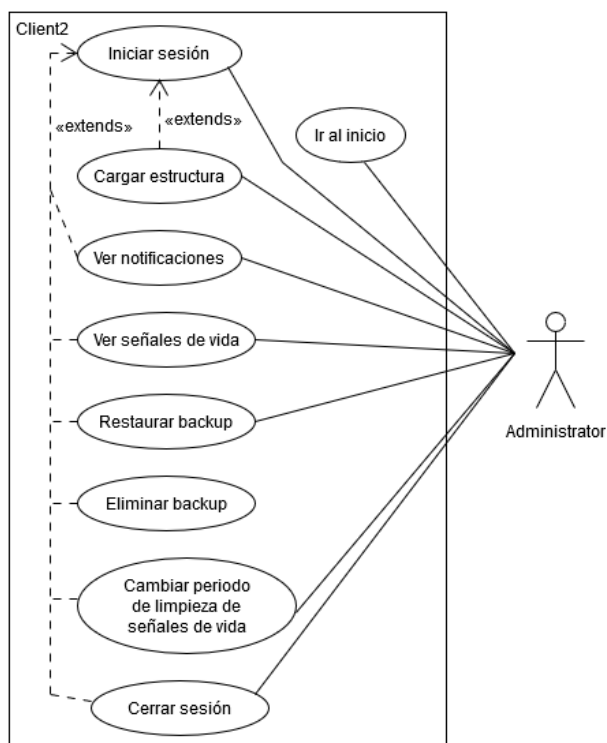
En este apartado se mostrarán los casos de uso principales para los diferentes actores del sistema, no obstante, algunos de ellos por ser muy genéricos, para no saturar el diagrama, se describirán en texto debajo de cada diagrama.



En el diagrama anterior se muestran todas las acciones realizables por los diferentes actores del sistema, a excepción del administrador. Para este actor, tan solo se han mostrado las acciones que comparte con el rol profesor, por lo que se mostrará en el siguiente diagrama el conjunto de acciones que faltan.

Como acciones genéricas indicadas en el diagrama están:

- Navegar: Esta acción incluye el realizar una búsqueda mediante la barra de direcciones o un buscador y/o navegar hacia atrás o hacia adelante.
- Mostrar mensaje: Contiene tanto mostrar mensajes emergentes mediante el sistema de Windows como mostrar páginas finales indicando alguna circunstancia.
- Enviar notificación: Incluye las notificaciones provocadas a raíz de acciones del usuario, así como las señales de vida del aplicativo cliente.
- Activar restricción: Gestiona el bloqueo de páginas y al provocar otros eventos, para el bloqueo y notificación se activarán también, según proceda, las acciones “Mostrar mensaje” y “Enviar notificación”.
- Gestionar tiempo de vuelo: Se encarga de gestionar los bloqueos en base a estar en este modo, así como de manejar las notificaciones que se produzcan y la desactivación.
- Ver notificaciones: Esta acción incluye el ver el resumen de notificaciones, así como los detalles de las mismas.
- Ver restricciones: Incluye ver el resumen de restricciones, así como los detalles.



Al igual que en el primer diagrama de casos de uso se han generalizado alguna de las acciones, de forma que el diagrama esté más claro.



Se muestra, a continuación, una descripción más detallada de las diferentes acciones del segundo diagrama:

- Cargar estructura: Incluye el cargar la estructura de alumnos-profesores-grupos base para el funcionamiento del sistema.
- Ver notificaciones: Abarca las notificaciones que se producen fuera de una restricción, como pueden ser las producidas en tiempo de vuelo o cuando no se ha solicitado inicio de sesión por no haber restricciones ese día.
- Ver señales de vida: Incluye tanto el resumen de señales de vida de los diferentes dispositivos que utilizan el sistema como el detalle de dichas señales.

Capítulo 2 DISEÑO Y ARQUITECTURA



VISTA DE BLOQUES

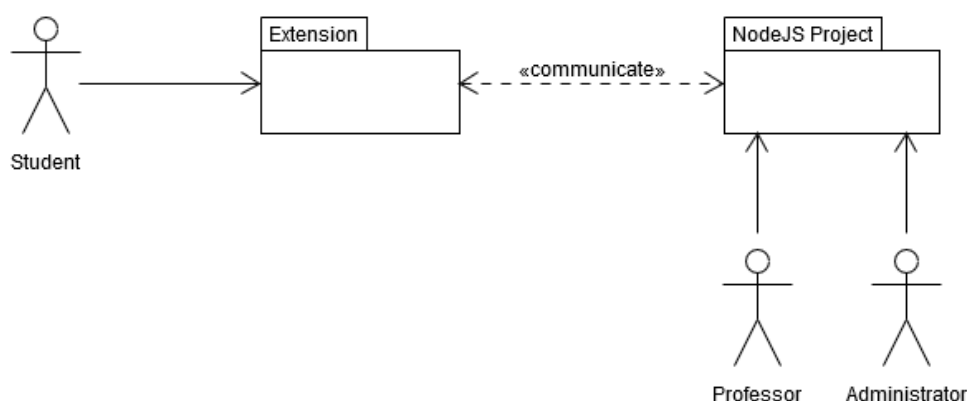
1.1. Diagrama de caja negra

El sistema software está formado por dos aplicativos, uno que actúa de cliente tratándose de una extensión para Google Chrome y la otra aplicación, basada en NodeJS, con una parte servidora y otra parte cliente.

A la extensión se pueden conectar usuarios con el rol “student” y su función será la de controlar y restringir la navegación del usuario en función de lo programado a través del proyecto NodeJS.

Adicionalmente, otros usuarios con el rol “professor” o “administrator” podrán acceder al aplicativo cliente que provee el proyecto NodeJS. Desde ahí en función del rol podrán aplicar restricciones, controlar notificaciones de acciones indebidas, cargar la estructura de datos y gestionar las copias de seguridad entre otras cosas.

Como se puede observar, el proyecto NodeJS provee una API REST a la cual se conecta la extensión y a su vez ofrece un aplicativo web a modo de panel de administración.



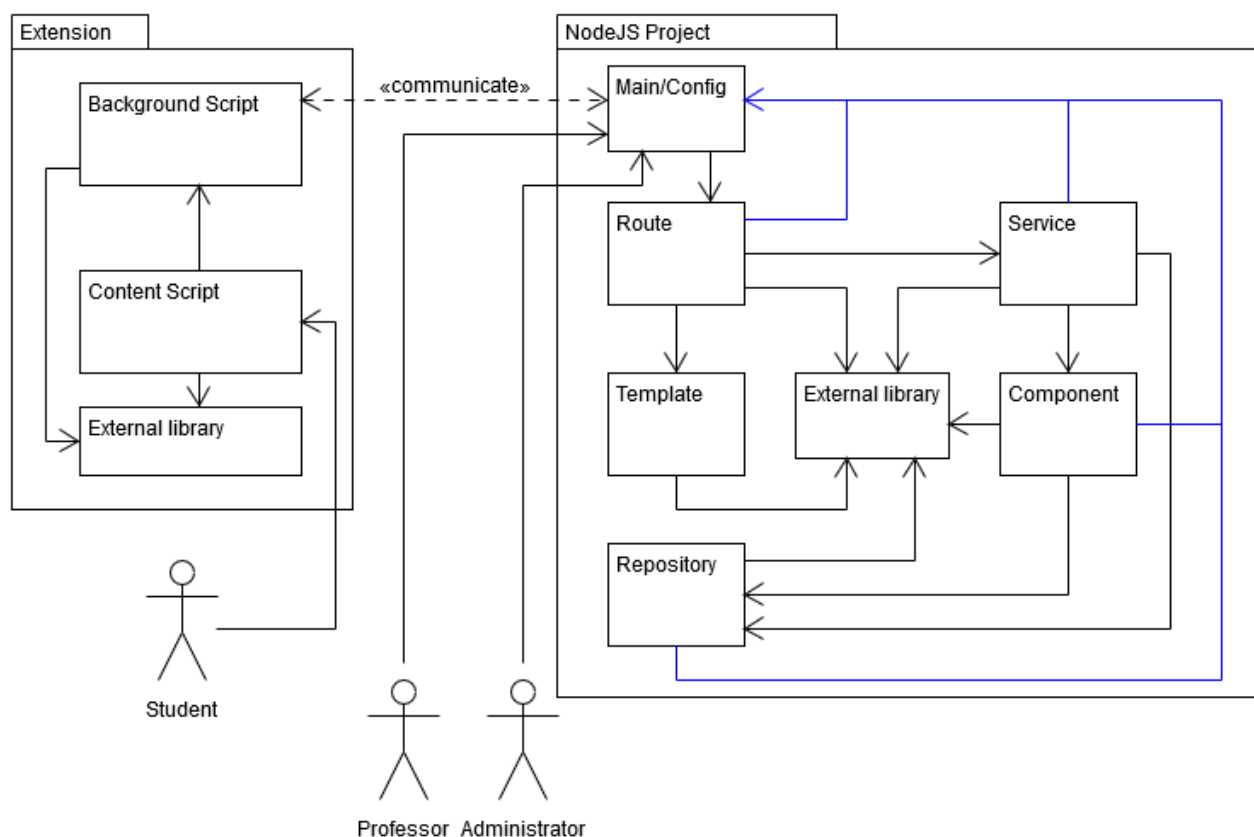
1.2. Diagrama de caja blanca 1

En el primero de los dos diagramas de caja blanca se desglosan los diferentes componentes principales que conforman cada uno de los 2 aplicativos.

La extensión, siguiendo la estructura que recomienda Chrome de cara a realizar una aplicación segura, está formada por los ficheros de código que se cargan en segundo plano y los ficheros de contenido. Los ficheros de contenido son los encargados de mostrar una interfaz gráfica al usuario, recoger los datos que el usuario ingrese y comunicarlos mediante la API de Google Chrome “Runtime” y la función “sendMessage” a los ficheros de código en segundo plano. Esto permite, que, aunque en el navegador este habilitado el cuadro de mandos para desarrolladores, el usuario no pueda acceder a los scripts de procesamiento los cuales llevan a cabo comunicaciones con el exterior, procesos de

criptología y estabilidad del sistema. De forma complementaria a ambos tipos de ficheros de código, se utilizan librerías externas y las diferentes APIs que provee Google Chrome para desarrollar extensiones.

El proyecto NodeJS mantiene una estructura similar a la arquitectura en capas mezclada con el modelo-vista-controlador, no obstante, incluye alguna variación producto de la tecnología utilizada y el proyecto en sí. Además, se utiliza como gestor de dependencias NPM que sirve a su vez como auditor, en cuanto a estabilidad y brechas de seguridad se refiere, ya que analiza las versiones de las librerías utilizadas en busca de problemas reportados. Aunque no se aprecie en la estructura de componentes, el proyecto de forma interna a través de los diferentes endpoints que proporciona se divide en una parte con función de API REST y otra como aplicación web.



Descripción del diagrama relativo a la extensión:

- Background Script: Páginas internas de procesamiento de las extensiones de Chrome. Se encuentran dentro de la carpeta coreJS.
- Content Script: Páginas internas para interactuar con el usuario en las extensiones de Chrome. Se encuentran en la carpeta principal de la extensión.
- External library: Librerías externas utilizadas en la extensión.

Descripción del diagrama relativo al proyecto NodeJS:

- **Main/Config:** Se encarga de gestionar la configuración general del aplicativo NodeJS además de ser el punto de entrada de la aplicación. Incluye declaración de variables de aplicación, inicialización de librerías, enrutamiento mediante privilegios, arranque del servidor y manejo de excepciones para recursos no disponibles.
- **Route:** Ficheros JavaScript, dentro de la carpeta routes, encargados de gestionar las peticiones realizadas por el usuario.
- **Service:** Ficheros JavaScript, dentro de la carpeta services, encargados de obtener y cargar datos en la base de datos además de realizar algunos procesamientos. La conexión con la base de datos la hace a través de los ficheros Repository, no directamente.
- **Component:** Fichero JavaScript, dentro de la carpeta validators, parecidos a un servicio que se encargan de realizar una tarea concreta. En este caso se usa para gestionar la validación de un formulario.
- **Repository:** Fichero JavaScript, dentro de la carpeta modules, encargado de realizar los accesos directos con el sistema de bases de datos.
- **Template:** Plantillas HTML, dentro de la carpeta views, mediante las cuales se muestra el contenido al usuario de forma dinámica.
- **External library:** Librerías externas utilizadas en el aplicativo NodeJS.

1.3. Diagrama de caja blanca 2

Añadiendo más detalle al diagrama de caja blanca anterior, se muestra a continuación otro en el cual se puede apreciar más la interacción entre las diferentes partes de ambos aplicativos.

Cabe reseñar que en el proyecto de NodeJS no se han vinculado las librerías en concreto con cada uno de los scripts que las utilizan para no complicar más la lectura con más líneas, además, las librerías que se muestran, aunque son prácticamente todas, no son la totalidad de las utilizadas, sino las que más influyen. Algo similar ocurre en los “background scripts” que puede faltar alguna relación.

Se describen a continuación los ficheros contenidos en el siguiente diagrama sobre el aplicativo NodeJS.

Main/Config:

- **app.js:** Se encarga de gestionar la configuración general del aplicativo NodeJS además de ser el punto de entrada de la aplicación. Incluye declaración de variables de aplicación, inicialización de librerías, enrutamiento mediante privilegios, arranque del servidor y manejo de excepciones para recursos no disponibles.

Route:

- **rapp.js:** Se encarga de gestionar las peticiones del usuario relativas a páginas generales de la aplicación. En este caso tan solo la página principal del panel de administración.

- `administrator.js`: Se encarga de gestionar las peticiones relativas al rol de administrador.
- `rprofessor.js`: Se encarga de gestionar las peticiones relativas al rol de profesor.
- `rstudentapi.js`: Se encarga de gestionar las peticiones a la API del rol estudiante.
- `ruser.js`: Se encarga de gestionar los inicios y cierres de sesión del panel de administración.
- `rusersapi.js`: Se encarga de gestionar la obtención del token para las futuras peticiones a la API REST.

Service:

- `rappService.js`: Se utiliza para gestionar las operaciones necesarias para obtener información de la base de datos o del archivo de configuración desde el archivo de entrada/configuración general.
- `administratorService.js`: Utilizado para obtener, almacenar y procesar datos relativos a las operaciones realizadas por el rol administrador.
- `rldapConnectionService.js`: Utilizado para gestionar las conexiones y validaciones con el LDAP de la universidad.
- `rprofessorService.js`: Se utiliza para obtener, almacenar y procesar datos relativos a las operaciones realizadas por el rol professor.
- `rstudentapiService.js`: Utilizado para obtener, almacenar y procesar datos relativos a las operaciones realizadas a la API REST desde el rol student.
- `rusersapiService.js`: Se utiliza para obtener la información necesaria cuando el usuario solicita un token a través de la API REST.
- `ruserService.js`: Se utiliza para obtener la información necesaria cuando el usuario inicia sesión a través del panel de administración.

Component:

- `slotValidator.js`: Encargado de validar la información referente al alta y modificación de restricciones.

Repository:

- `bdManagement.js`: Encargado de gestionar las conexiones directas con la base de datos.

Template:

- `public resources folder`: Contiene los ficheros CSS, imágenes y JavaScript que no requieren privilegios para ser accedidos. Se encuentran dentro de la carpeta `public`.
- `base.html`: Base de todas las plantillas que se encarga de gestionar los menús y apariencia básica del panel de administración.
- `professor`: Carpeta con las plantillas HTML encargadas de mostrar al usuario las peticiones relativas al rol professor.
- `admin`: Carpeta con las plantillas HTML encargadas de mostrar al usuario las peticiones relativas al rol administrator.
- `main`: Carpeta con las plantillas HTML encargadas de mostrar al usuario la página principal, así como la página de inicio de sesión.

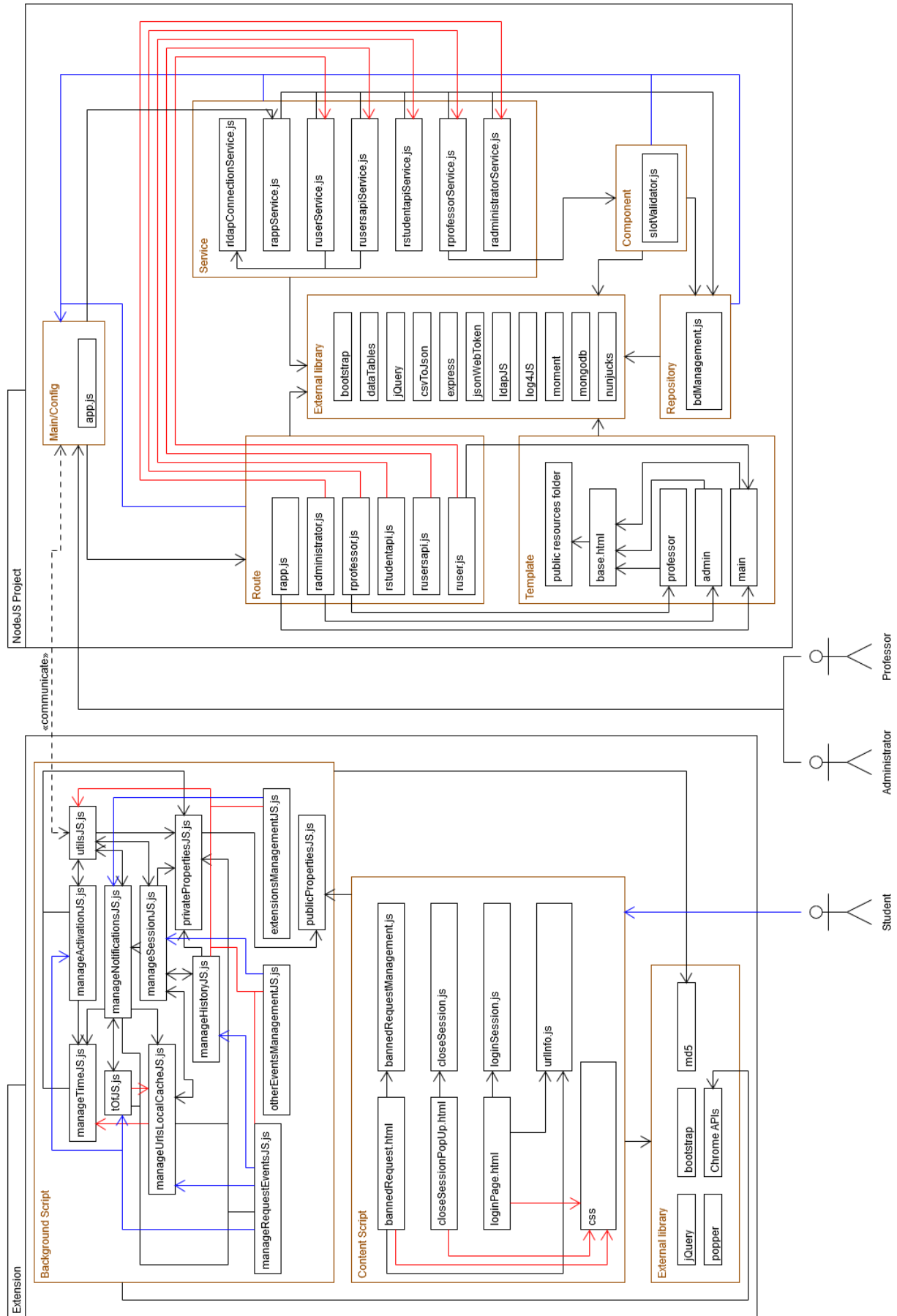
Se describen a continuación los ficheros contenidos en el siguiente diagrama sobre la extensión.

Background Script:

- `manageTimeJS.js`: Se encarga de gestionar el manejo de la hora recibida por el servidor, así como de validarla.
- `extensionsManagementJS.js`: Se encarga de gestionar los eventos relacionados con cambios en la configuración de las extensiones: activaciones, instalaciones, desinstalaciones y desactivaciones.
- `manageActivationJS.js`: Se encarga de gestionar la activación o no de la extensión en función de que ese día haya o no restricciones. También valida su integridad.
- `manageHistoryJS.js`: Utilizado para gestionar la gestión de la navegación hacia atrás, así como los eventos relacionados con el historial del navegador.
- `manageNotificationsJS.js`: Se utiliza para gestionar las notificaciones a la API REST.
- `manageRequestEventsJS.js`: Utilizado para gestionar las peticiones de navegación.
- `manageSessionJS.js`: Se utiliza para gestionar la sesión, es decir, el inicio de sesión, el borrado de datos al abrir el navegador, la expiración del token y parar las acciones programadas.
- `manageUrlsLocalCacheJS.js`: Utilizado para gestionar las restricciones recibidas por la API REST, así como programar las activaciones/desactivaciones de las restricciones, validar su integridad y comprobar si una URL está permitida o no.
- `otherEventsManagementJS.js`: Se utiliza para gestionar los eventos ocurridos al abrir el navegador y cuando se crean marcadores.
- `privatePropertiesJS.js`: Se incluyen las variables de la extensión que son utilizadas de forma privada en el procesamiento interno.
- `publicPropertiesJS.js`: Se incluyen las variables de la extensión que son utilizadas tanto de forma privada en el procesamiento interno como de forma pública mediante las páginas y scripts que interactúan con el usuario.
- `tOfJS.js`: Se encarga de gestionar el modo tiempo de vuelo.
- `utilsJS.js`: Usado para pequeñas acciones diversas como obtener el dominio principal de una URL, programar las notificaciones de señal de vida, obtener las direcciones IP internas, mostrar notificaciones mediante la bandeja del sistema, una función genérica para peticiones web y un método para actualizar las pestañas del navegador.

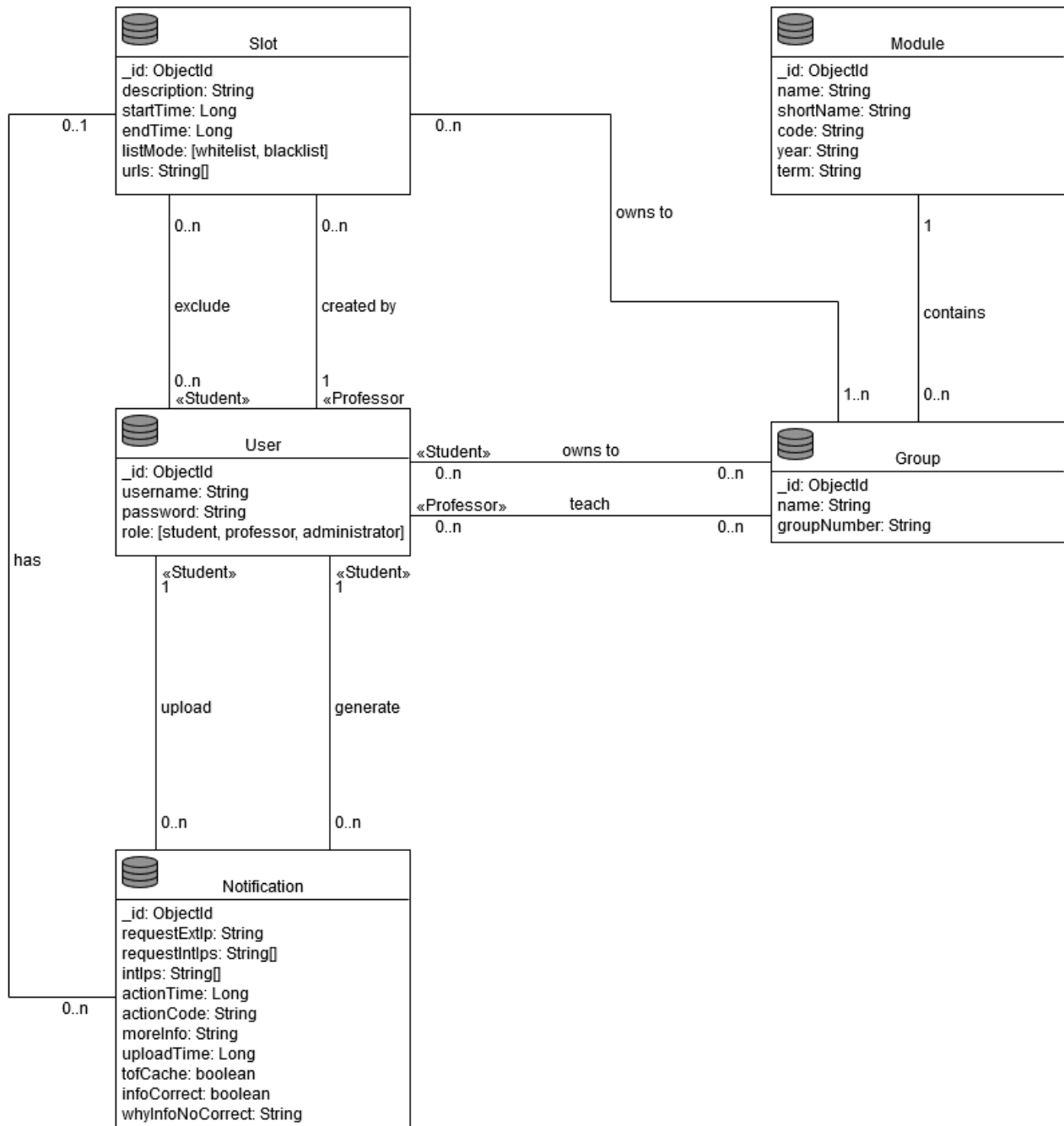
Content Script:

- `bannedRequest.html` y `bannedRequestManagement.js`: Utilizados para mostrar la página de acceso restringido.
- `closeSessionPopUp.html` y `closeSession.js`: Utilizados para cerrar la sesión al clicar sobre el icono de la extensión.
- `loginPage.html`, `loginSession.js` y `urlInfo.js`: Utilizados para mostrar el inicio de sesión al usuario.
- `css`: Incluye los estilos CSS propios utilizados, así como los propios de Bootstrap.



MODELO DE DOMINIO

Aunque la base de datos es no relacional, puesto que se trata de MongoDB, se muestra la estructura del modelo de dominio a modo de diagrama entidad-relación.



DIAGRAMAS DE SECUENCIA

En este apartado se mostrarán los principales diagramas de secuencia del proyecto software teniendo en cuenta que su principal objetivo es el de mostrar la interacción entre el cliente y el servidor a la hora de trabajar con datos, es decir, las peticiones de documentos HTML, CSS, JavaScript... no se mostrará en los mismos.

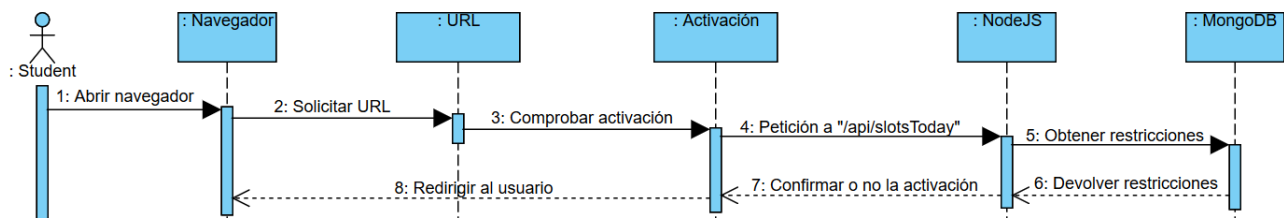
Para todos los diagramas de secuencia, a excepción de los referentes al inicio de sesión, se entenderá que el usuario ya ha iniciado sesión en su rol correspondiente, así como que se envía el token o la sesión almacenada, y es comprobado en el servidor.

De cara a generalizar los esquemas y hacer más sencilla su comprensión y extensión, no se indican, en las llamadas, los nombres de los métodos concretos, sino que se describe una acción. En este sentido, tampoco se indican algunas funciones secundarias que no son de mucha relevancia.

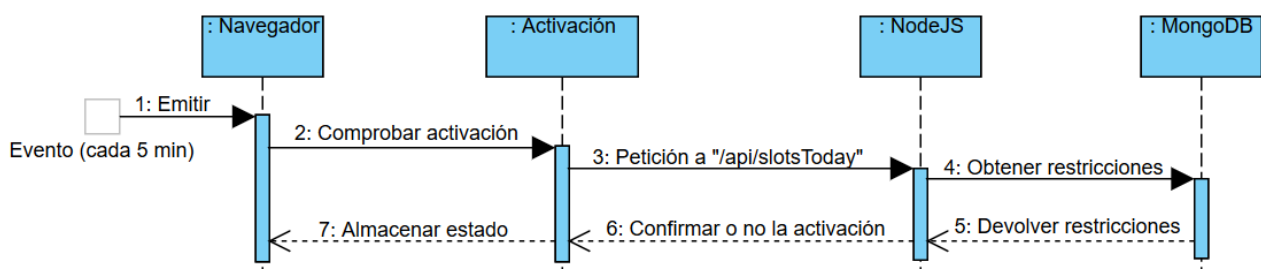
La principal información que se quiere transmitir con estos diagramas es la minimización de la carga al servidor a la hora de realizar peticiones el usuario. De esta forma se consigue el fin inicial de poder desplegarlo, y utilizarlo, en la facultad teniendo en cuenta los recursos de computación disponibles.

3.1. Rol student – Extensión de Chrome

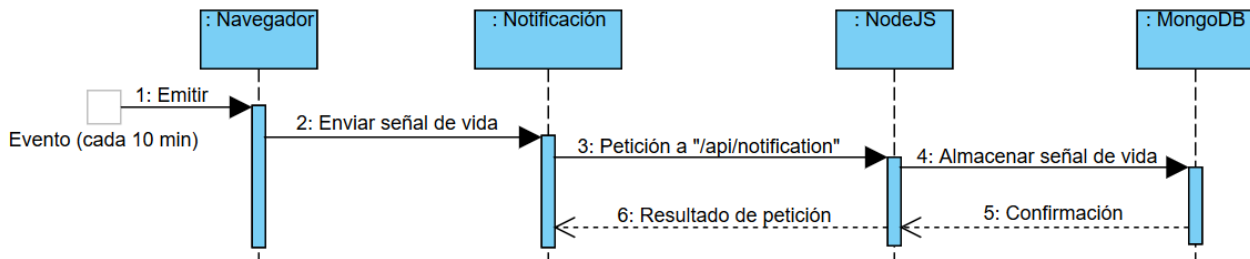
Al abrir el navegador



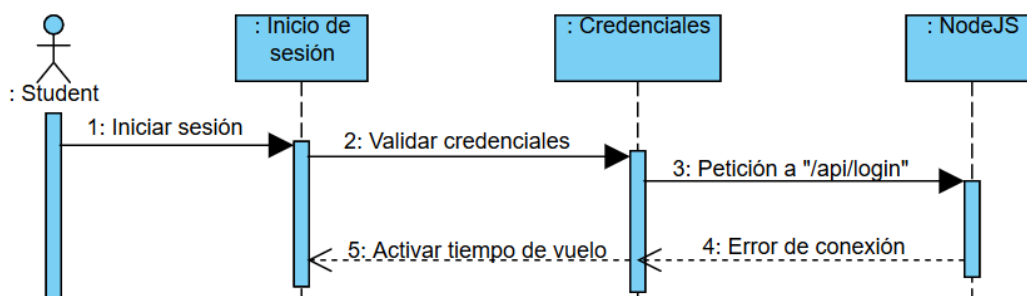
Comprobar activación periódicamente



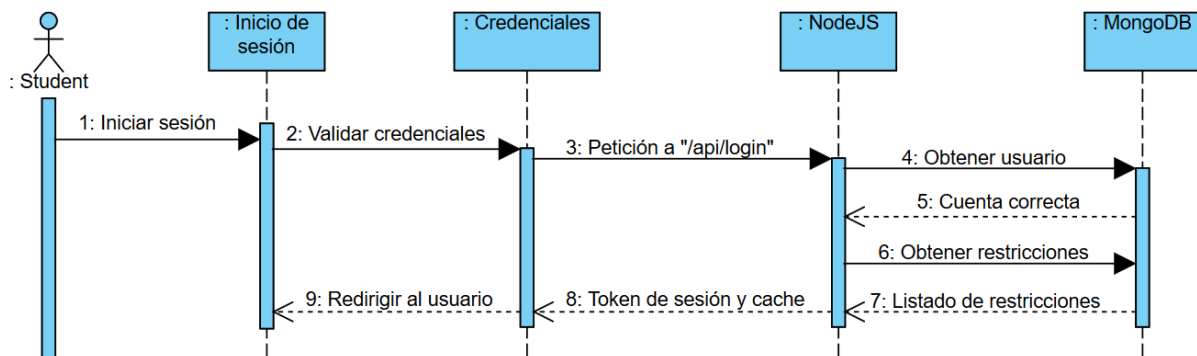
Señales de vida de la extensión



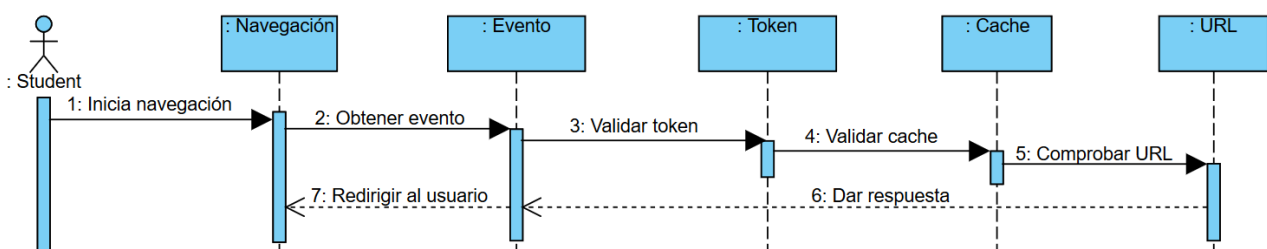
Inicio del tiempo de vuelo



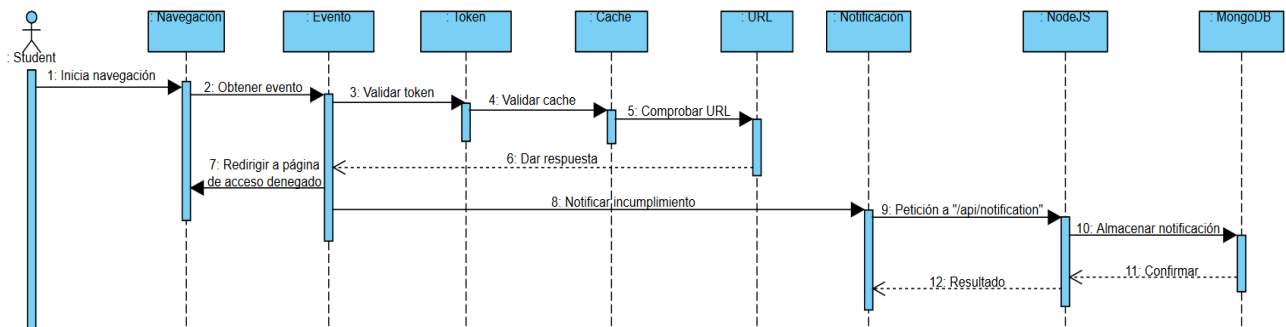
Inicio de sesión



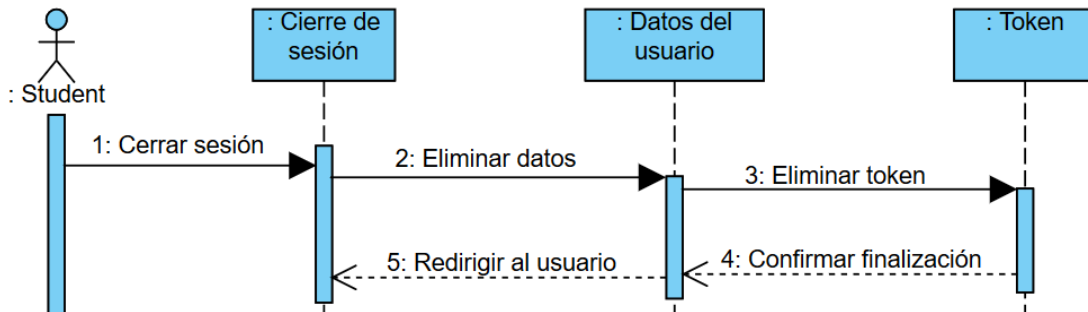
Petición web permitida



Petición web denegada

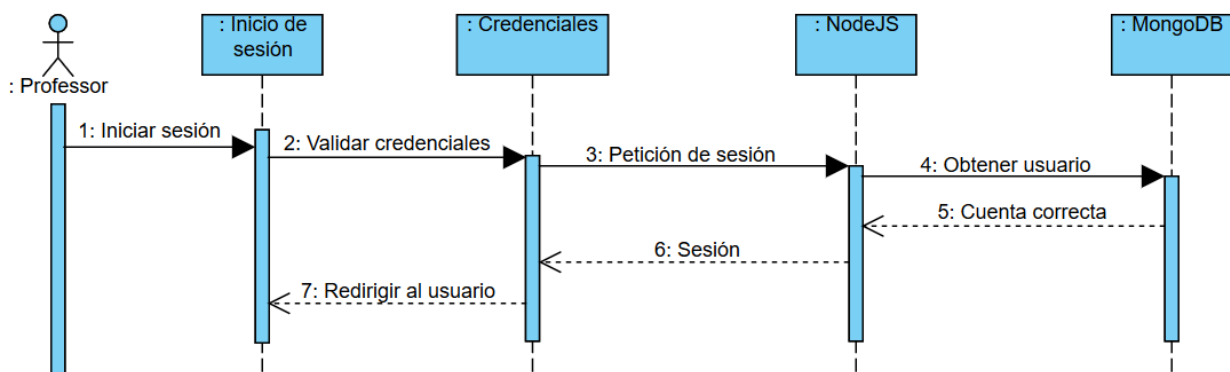


Cierre de sesión

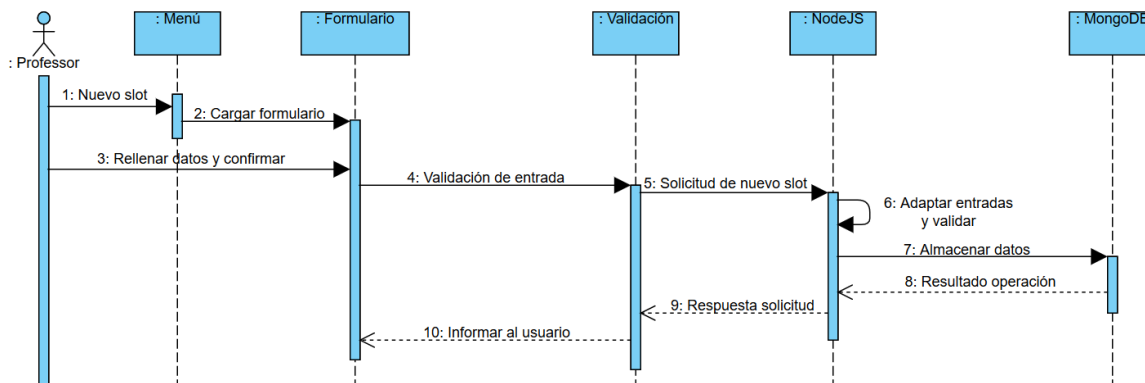


3.2. Rol professor – Portal administración

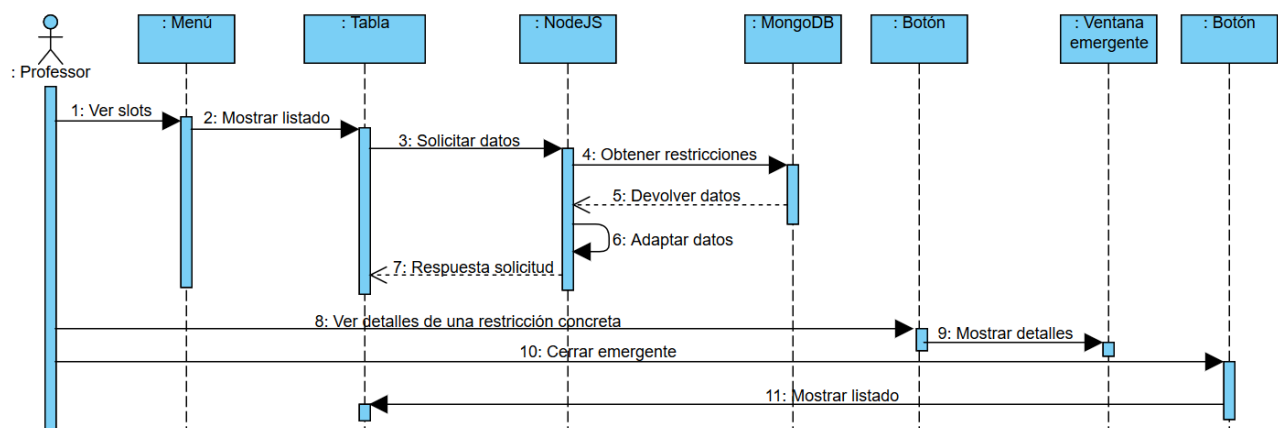
Inicio de sesión



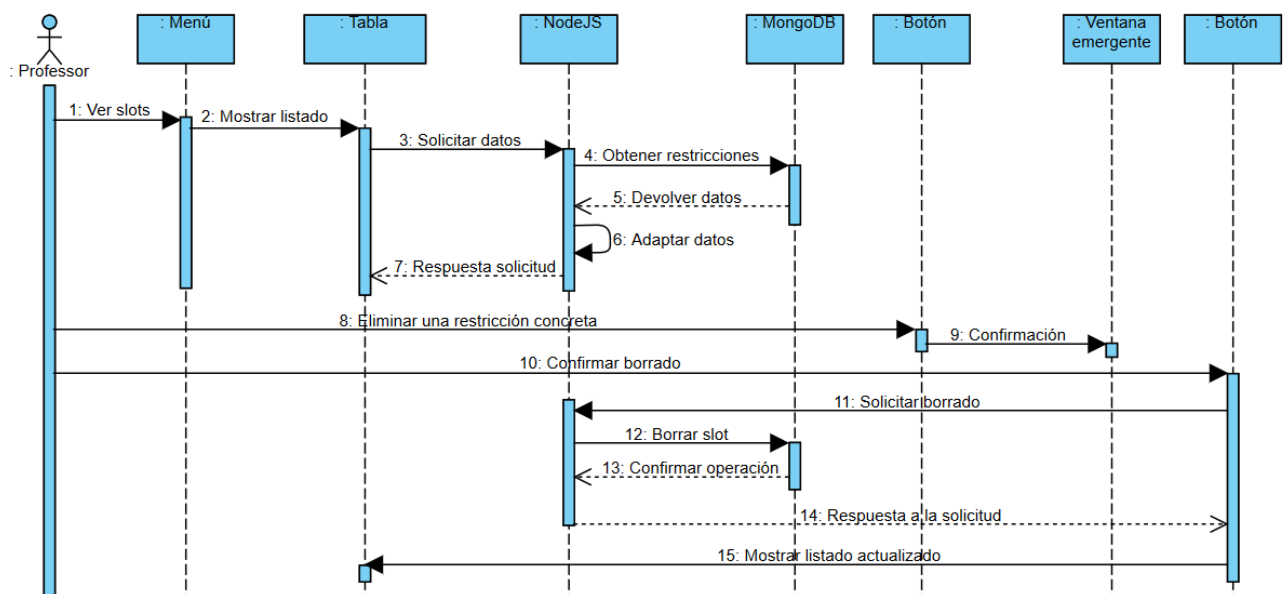
Añadir restricción



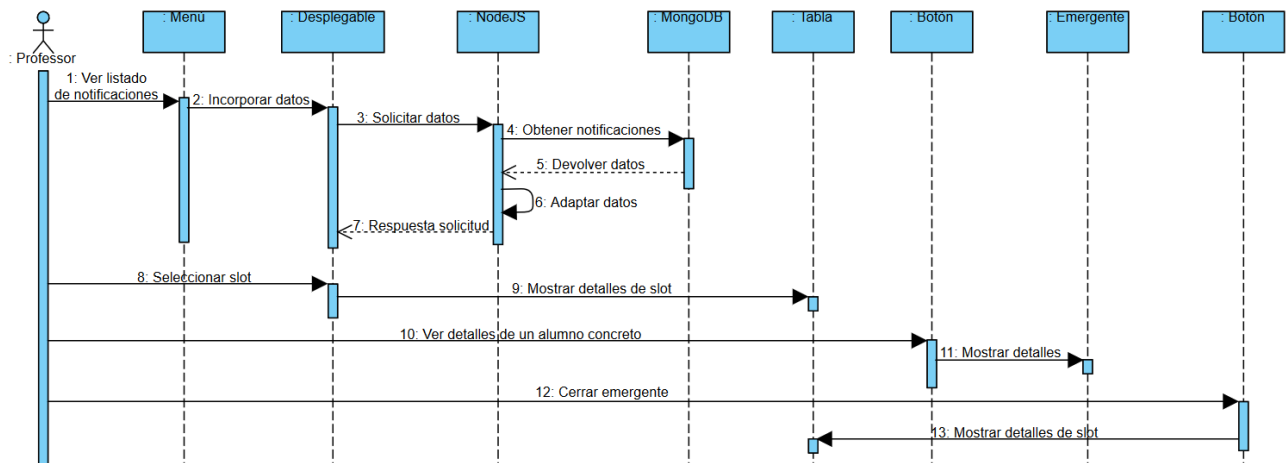
Ver restricciones – Opción “detalles”



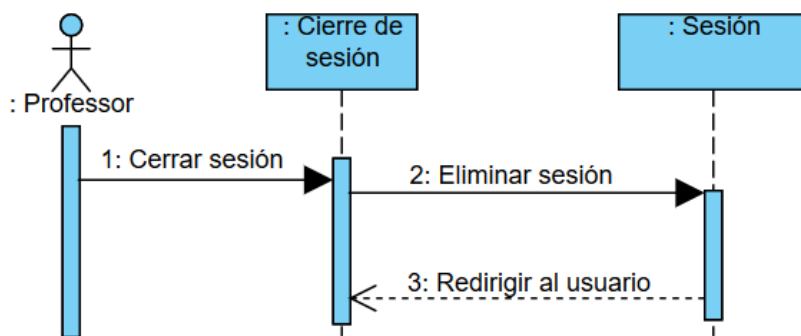
Ver restricciones – Opción “eliminar”



Ver notificaciones de alumnos

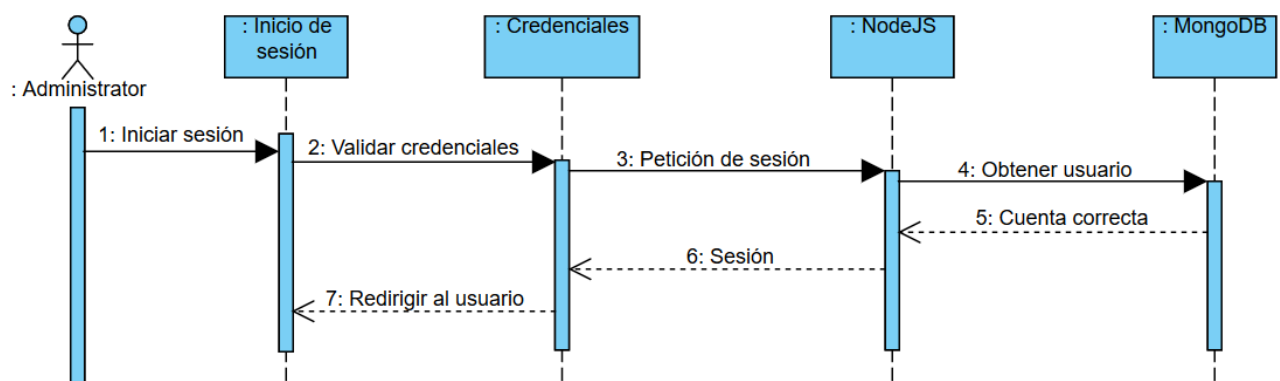


Cierre de sesión

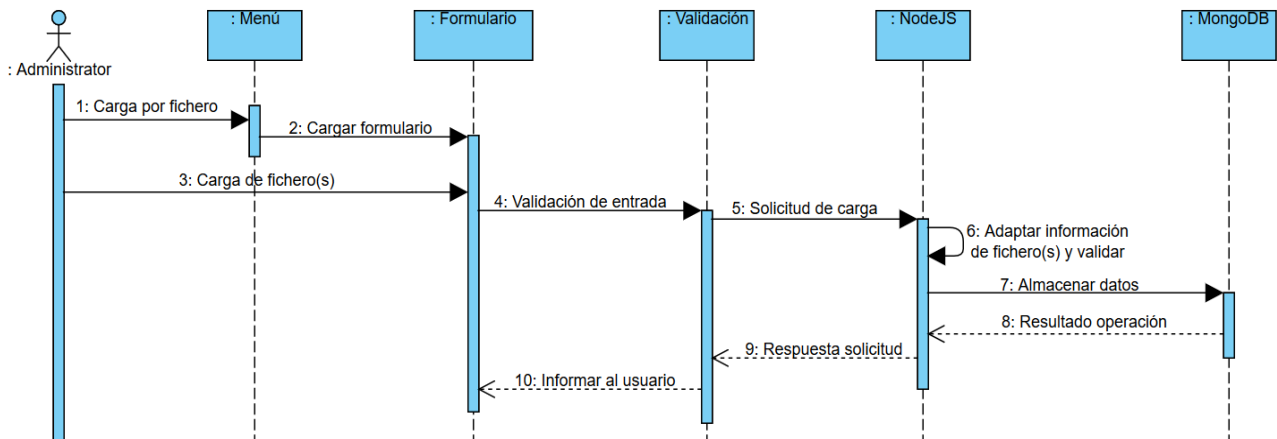


3.3. Rol administrador – Portal de administración

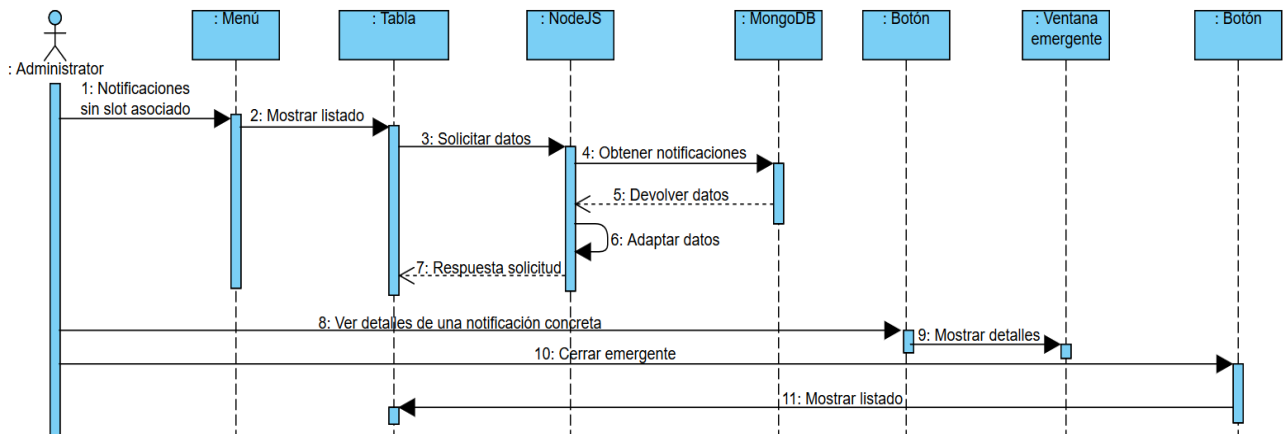
Inicio de sesión



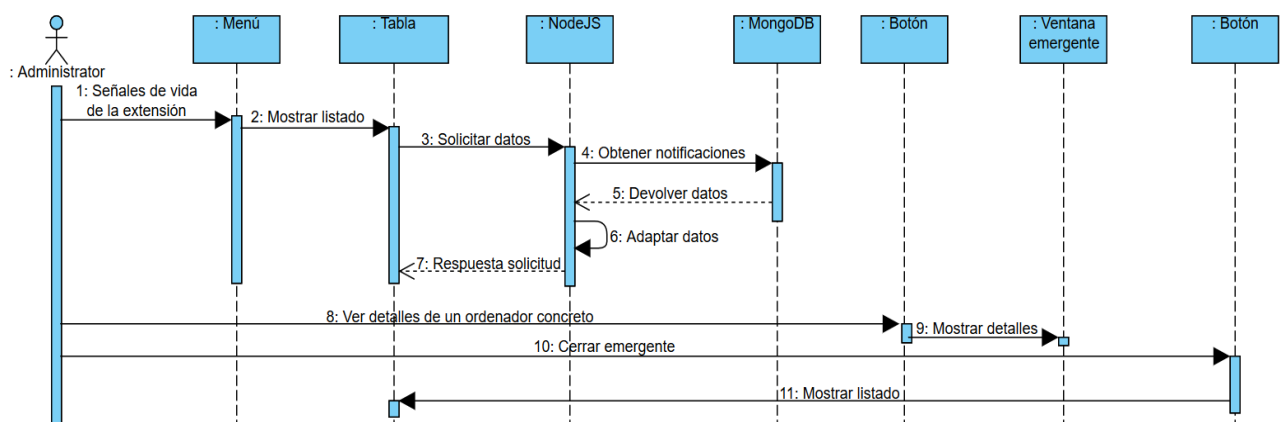
Cargar ficheros de grupos-alumnos-profesores



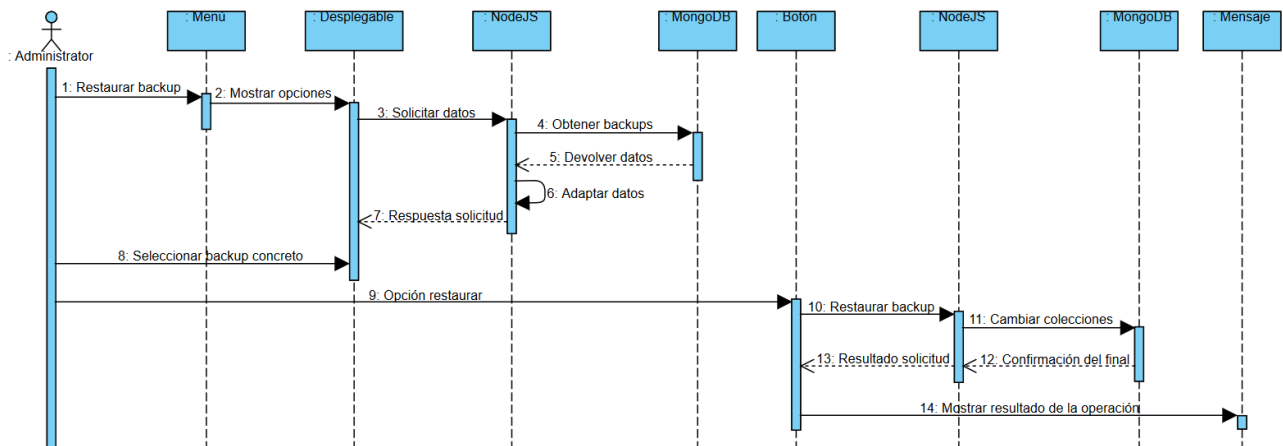
Ver notificaciones sin slot asociado



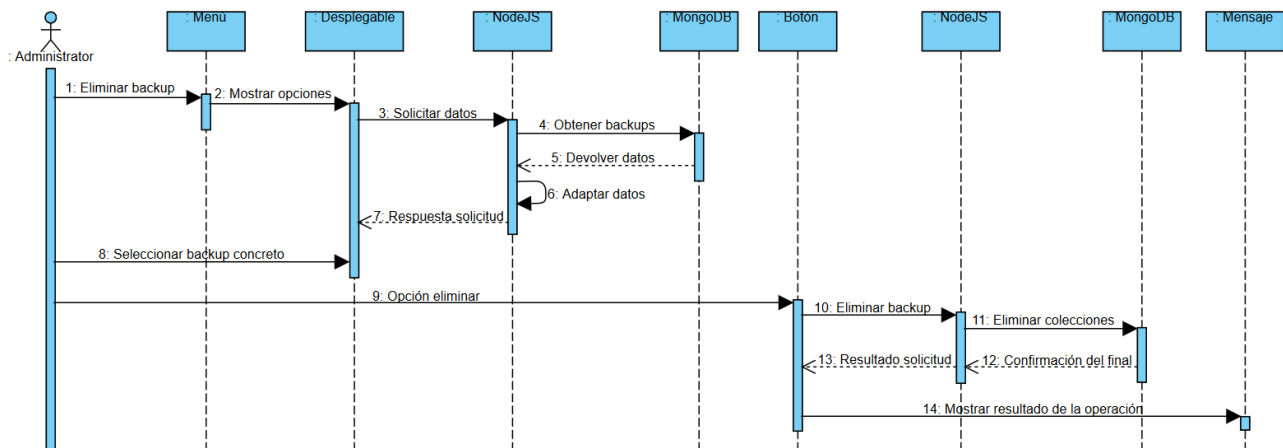
Ver señales de vida



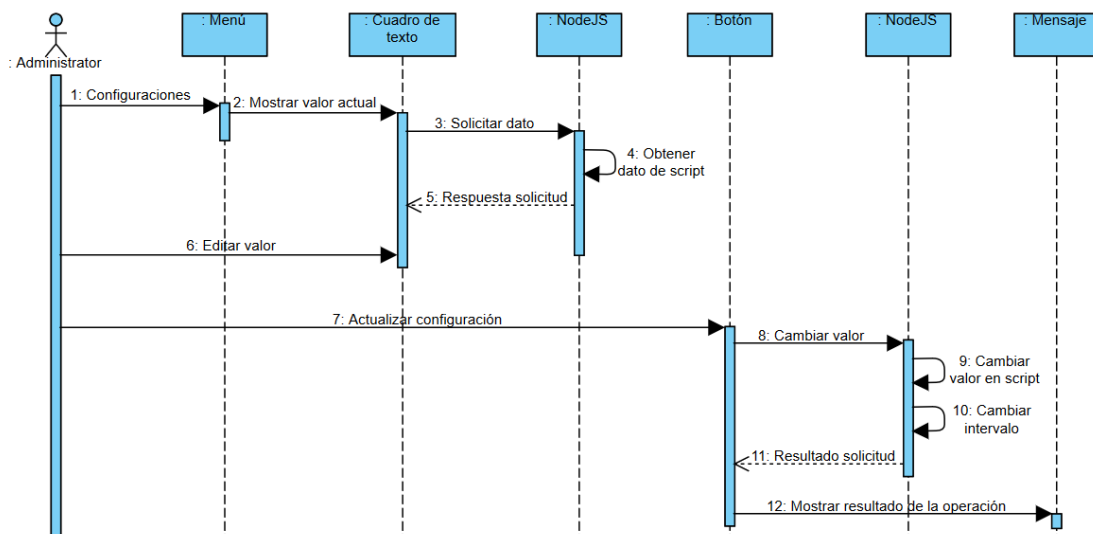
Restaurar copia de seguridad



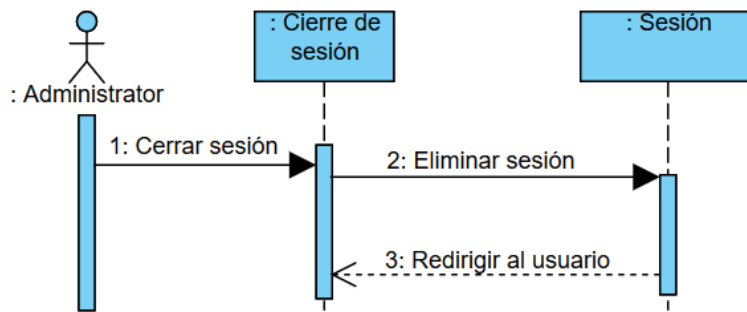
Eliminar copia de seguridad



Editar configuraciones



Cierre de sesión



Capítulo 3 DETALLES DE IMPLEMENTACIÓN



EXTENSIÓN PARA CHROME

1.1. Descripción breve

A lo largo del presente apartado, referente a la extensión de Chrome, se mostrarán diversos problemas surgidos durante el desarrollo de la misma. Los principales problemas, de forma resumida, son los siguientes:

- Google Chrome no permite bloquear peticiones a todas las URL existentes por seguridad.
- La forma de bloqueo, mediante la opción que provee Google, de peticiones es síncrona.
- Chrome no comprueba la integridad de extensiones instaladas de un origen externo a su tienda de aplicaciones.
- El usuario podría acceder a páginas intermedias utilizadas por la extensión si no se gestiona el historial.
- En la primera de las soluciones probadas se producen condiciones de carrera.
- Con la segunda solución no en todas las navegaciones se lanza el evento de petición que se produce prácticamente al instante de solicitar la navegación, sino que solo es común el evento que se produce al tener la página prácticamente cargada. Además, se producen problemas en páginas que requieren inicio de sesión.
- Si no hay restricciones en un día solicitará inicio de sesión no teniendo por qué hacerlo.
- Se sobrecarga el servidor por comprobación de peticiones en tiempo real.
- Con la segunda solución no se detectan de forma correcta las descargas.
- Si no se activa la extensión por no haber restricciones ese día, pero minutos más tarde se crea una restricción, el usuario que ya haya entrado no tendría que iniciar sesión.
- El usuario podría acceder a parte del código interno.
- Si el servidor de control de restricciones no responde, al intentar iniciar sesión, el usuario no podría navegar.
- Si se producen infracciones y en ese momento el servidor no responde, las notificaciones se perderían.

1.2. Análisis de peticiones

Inicialmente para detectar las peticiones realizadas por el usuario se utilizó la API WebRequest que provee Google Chrome. Esta API permite cancelar la petición o redirigirla a una URL.

El problema que tiene el uso de esta API es, que, según la política de seguridad de Chrome, para poder bloquear o redirigir a otra URL, la URL de la solicitud del usuario debe estar incluida en el manifiesto de la extensión. Esto hace que, como se van a cambiar las URL a las que puede acceder al usuario en tiempo real, se tengan que añadir todas las URL mediante una expresión regular al manifiesto. Y aunque funciona, a la hora de subir la extensión a la Google Chrome Store, que es la

única forma que tiene Chrome para poder instalar sus extensiones sin perder la comprobación de integridad de las extensiones, te indica que debes ser el propietario de las URL que has indicado en el manifiesto para poder incluirlas en el mismo, por lo que no aprueban su publicación.

Otro problema es que esta API de Chrome solo bloquea o redirige las peticiones de forma síncrona, por lo que al realizar una petición a la API REST o a la caché local y devolver la respuesta de forma asíncrona, ya no se puede bloquear en función de lo que permitan o no ambos métodos, lo que supone un gran problema.

Una solución a este último problema es redirigir la petición siempre por defecto a una página de espera y mediante el uso de la API de Chrome Tabs detectar cuando se actualiza una pestaña, para que en caso de ser la página de espera entonces empiece el proceso de forma asíncrona, el cual consiste en comprobar la integridad de la caché local y comprobar que la URL solicitada está o no permitida en ese momento. En caso de que no fuera válido porque aún no se inició sesión o porque se trató de corromper la caché, el usuario será redirigido a la página de inicio de sesión para, a continuación, analizar la petición y cargar al usuario la página solicitada o indicarle que no puede acceder a esa URL.

Realizando lo anterior el sistema funciona, pero el hecho de que haya un evento que se activa cada vez que se actualiza una pestaña ralentiza el sistema. Además, para permitir al usuario ir hacia atrás sin que vaya a la página de espera, hay que indicar otro evento que se encuentra dentro de la API WebNavigation mediante el cual se puede saber si el usuario ha tratado de ir hacia atrás en la navegación, y que en caso afirmativo si detecta que la página destino es una propia de la extensión, realice automáticamente otra vuelta atrás para ir a la página a la que querría ir el usuario. Esto tiene un problema, y es que cuando se carga momentáneamente la página de espera hasta que automáticamente se da hacia atrás, el evento que detecta si se carga la página de espera se activa y redirige la petición de nuevo a donde estaba el usuario inicialmente, por lo que se entra en una especie de bucle por condición de carrera que termina cuando el evento que detecta si se va hacia atrás finaliza antes de que se haya completado el evento que detecta si se está en la página de espera.

Debido a todo lo anterior se decidió analizar otras posibilidades y se optó por elegir la API WebNavigation, ya usada para gestionar la marcha atrás durante la navegación. No obstante, aunque se vaya a utilizar para un fin similar a la anterior mostrada, no está pensada exclusivamente para redirigir peticiones o anularlas, sino que trata de permitir actuar ante una petición de formas muy diversas, bien sea en el mismo momento de la solicitud, en otros estados intermedios o una vez completada.

La principal ventaja de esta API respecto a la anterior es que es asíncrona, por lo que no hay que andar utilizando páginas intermedias que luego son detectadas por otros eventos. Además, tampoco exigen que se incluyan las URL en el manifiesto, lo que permite que se pueda publicar en la Chrome Web Store.

Otra ventaja es el hecho de que esta API no solo detecta solicitudes que vayan a través de la red, si no que detecta todas las solicitudes realizadas al navegador en sí, como por ejemplo el acceso a la

administración de extensiones que tiene su propia URL interna. Esto permite que se pueda bloquear el acceso a la configuración, al historial, a la administración de extensiones para que no las puedan desactivar / desinstalar... No obstante, se aplicarán otras medidas de seguridad al margen de la extensión para evitar que el usuario la pueda manipular. La ventaja de esta API en este aspecto está en que hay páginas de configuración del navegador que se pueden querer permitir o no en función de los usuarios, horas... y de esta forma se podría hacer gestionándolo como cualquier otra URL.

Como desventaja de utilizar esta API está que, el evento que se produce en el momento de realizar una petición, no se lanza siempre como haría el de la API WebRequest, si no que por ejemplo, para las páginas propias del navegador, no se ejecuta, e incluso en Opera que permite la extensión, a la hora de realizar una búsqueda desde la barra de direcciones no se ejecuta el evento, porque esta API se basa más en los procesos por los que el navegador pasa en una petición (los que ve el usuario) respecto a las peticiones realizadas a través de la red como haría la API WebRequest. La solución a esto es utilizar el evento que se produce cuando prácticamente se han recibido todos los datos solicitados. Además, algunas páginas que requieren inicio de sesión, como la del campus virtual de la universidad, no funcionan a la primera si se utiliza el evento que se produce cuando se realiza la petición. Esto da lugar a que si, por ejemplo, la API REST tarda más en responder, la página llegue a pintarse en el navegador y el usuario pueda acceder momentáneamente al contenido. Además, si se trata de acceder a un enlace de descarga directo, como puede ser "<http://go.microsoft.com/fwlink/?LinkId=863262>" al no cargarse nada en el navegador, no se lanza el evento y por tanto se permite su descarga, lo que va en contra de la finalidad de este proyecto.

Una de las soluciones encontradas para lo primero ha sido el redirigir, según llega la petición, a la página de espera ya mencionada en la opción de la API WebRequest, ya que no permiten parar la petición a través de extensiones, que hubiera sido lo ideal. Obviamente, se vuelve a tener el problema a la hora de que el usuario vaya hacia atrás, pero esta vez no se produce una carrera entre dos eventos, ya que solo se lanza el que detecta si el usuario ha querido ir hacia atrás en la navegación, por lo que funciona sin problemas, y encima si falla el servidor, entre que pasa el tiempo máximo de solicitud y no, el navegador indicará al usuario que su solicitud está siendo procesada, por lo que no se le muestra ninguna información no permitida y se le mantiene informado. Esta primera solución es la utilizada en caso de querer analizar cada petición en tiempo real con la API REST. La opción funciona bien si solo se hacen unas pocas solicitudes, pero si hay varios usuarios utilizando el sistema, el servidor se sobrecarga de una forma bastante considerable sin llegar a ser necesario para el problema que se quiere resolver. Por ello se opta por descargar, cada vez que el usuario inicia sesión, a una cache local las páginas que tiene autorizadas / denegadas ese usuario. De esta forma la solicitud se resuelve mucho más rápido, con lo que no llega a cargarse la página nunca, antes de comprobar que el usuario tiene acceso a la misma, y, por tanto, no hace falta que haya ninguna página de espera entremedias. Esto hace, en un principio, que tampoco sea necesario el uso de ningún evento para controlar la navegación hacia atrás, siendo cierto que el usuario podría llegar a la página de inicio de sesión, lo cual no se considera un problema de seguridad ni funcional. No obstante, se ha observado tras implantar la solución, que, al ir hacia atrás el navegador, habiendo alguna página no permitida entremedias, la extensión te permite acceder ya que no se valoran las páginas a las que se accede

yendo hacia atrás por rapidez en el procesamiento de las peticiones, lo cual es un problema que afecta al fin principal de esta extensión que es poder restringir la navegación a ciertos sitios. La solución sería analizar también las páginas cuando se va hacia atrás, pero entonces nunca deja navegar hacia atrás porque en cuanto se llega a la página no permitida salta el evento que muestra la página de bloqueo y así una y otra vez, por lo que se ha optado por seguir incorporando la tabla hash propia que incluye las URL por las que navega el usuario en cada pestaña abierta en el navegador. El coste que tiene esta solución es el siguiente: si el profesor cambia una regla en tiempo real, los usuarios que ya estuvieran conectados tendrán que volver a iniciar sesión, y como máximo tendrán las reglas atrasadas durante los 45 minutos que dura la sesión.

Para solucionar el problema de las descargas, se ha optado por detectar el evento que se produce cuando se realiza la petición y almacenar la URL de dicha petición mediante la API Storage de html5. De esta forma si se produce una solicitud web, dicha URL almacenada será borrada en el evento de página completada, y en caso contrario se quedará almacenada a la espera de que sea procesada por el evento, de la API Downloads de Chrome, que detecta si se ha iniciado una descarga. De esta forma, se cancela la descarga y se inicia el proceso de comprobar que el usuario tiene acceso al dominio en el que se encuentra alojado ese recurso.

1.3. Más detalles de implementación

La extensión, siempre que se arranca el navegador comprueba si existe alguna restricción para algún usuario durante ese día, de esta forma, en caso de que no haya ninguna, no solicitará inicio de sesión y el usuario no se percatará de que la extensión está ahí. Esto además se complementa con una comprobación cada 5 minutos para evitar ataques contra este aspecto, cambiando alguno de los valores almacenados para controlar este funcionamiento o por si se da la situación de que minutos más tarde se ha añadido una restricción para ese día.

Además de las API principales utilizadas para gestionar las peticiones, se utilizan también, entre otras:

- API History: Permite detectar cuando una página se carga en el historial, para de esta forma, borrarla del mismo si se trata de una página propia de la extensión.
- API Bookmarks: Para impedir que el usuario añada marcadores.
- API Management: Al margen de que se pueda gestionar con una directiva la instalación, desinstalación, activación y desactivación de extensiones en Google Chrome, se hace uso de esta API (no funciona en Opera) que permite detectar eventos acerca de estas acciones. De esta forma se puede notificar a la API REST y dejarlo ahí registrado, además de avisar al usuario, aunque no se impide la acción, ya que, si el usuario instala una extensión al tratar de desinstalarla Chrome muestra un mensaje de confirmación al usuario y sería él finalmente quien elegiría. Algo similar ocurre cuando se desinstala, pero en este caso el problema surge de que al desinstalarse ya no se dispone del paquete para instalarla de nuevo.
- API Storage: Se borran todos los datos de navegación cada vez que el usuario cierra sesión, de forma manual o cuando cierra el navegador, para que de esta forma el siguiente usuario

disponga del navegador limpio sin ningún tipo de dato, lo cual incrementa la privacidad y la seguridad, por ejemplo, ante un examen que requiera el acceso a determinadas páginas, para que de esta forma no tenga ya el trabajo hecho el siguiente usuario, en caso de que usen el mismo.

- API Runtime: Sistema de mensajes para separar claramente el código computacional del que sirve para mostrar la parte visible del usuario. Esto es necesario, puesto que en la parte visible muchas veces se deben hacer consultas a la parte computacional, y el hecho de que se hagan directamente desde las páginas de usuario, hace que sean visibles estructuras internas de la extensión cuando no deberían ser vistas por motivos de seguridad. De esta forma desde la parte del usuario se lanzan mensajes que son recogidos y procesados por las “background pages” que se encargan de realizar las operaciones y ofrecer una respuesta, para que en función de esta al usuario se le muestren unas cosas u otras. Así se consigue que se realicen todas las operaciones necesarias, sin que el usuario pueda ver cuáles son esas operaciones.

Otro problema que puede surgir es el hecho de que el servidor que gestiona los inicios de sesión no responda en un momento dado. Esto hará que el usuario no pueda navegar en nada, y, por tanto, si pasa durante un examen, dar lugar a un problema. La solución que se ha implementado para solucionar esto ha sido el utilizar el conocido “tiempo de vuelo”, de manera que, si el servidor no responde, el usuario podrá navegar en unas páginas por defecto, las cuales en el caso del prototipo son las pertenecientes al dominio “uniovi.es”, asegurando de esta forma el funcionamiento del campus virtual.

Para enviar las notificaciones de infracciones cometidas por el usuario, como son, en el caso del prototipo: acceder a URL no permitidas o tratar de modificar las extensiones disponibles en el navegador, se procede a enviarlas según se producen a la API, realizando un máximo de 3 intentos, en caso de no poder llegar a notificarse a la primera. Lo mismo se hace, aunque solo con la modificación de las extensiones disponibles, si se encuentra en modo tiempo de vuelo, pero en este caso, si no se llega a notificar en los 3 intentos, estas son almacenadas en una cache local que se intenta enviar cada vez que un usuario inicia sesión.



APLICATIVO NODEJS

2.1. Descripción breve

Al tener los aplicativos desarrollados en NodeJS una estructura bastante general sobre todo vinculada a razones de seguridad, se detallarán las particularidades llevadas a cabo en el Capítulo 5, en lugar de en el presente título.

Capítulo 4 PROCESOS DE CALIDAD



EXTENSIÓN PARA CHROME

1.1. Casos de test

Durante el desarrollo de la extensión se han llevado a cabo multitud de pruebas para comprobar su correcto funcionamiento. Si bien son pruebas que no se han automatizado por la naturaleza de la solución y por limitaciones de tiempo, se han llegado a ejecutar de forma manual para obtener resultados y modificar el desarrollo cuando fuera necesario.

Para la ejecución de los casos de test hay que generar las condiciones externas necesarias para que la extensión funcione de la forma descrita. Es decir, hay que tener usuarios, las restricciones oportunas indicadas en cada test, no tener iniciada la sesión en el perfil de Chrome y no activar el modo incógnito.

Los casos de test, **cuando no se está en tiempo de vuelo y ese día hay restricciones**, diseñados y ejecutados son los siguientes:

- No habiendo iniciado sesión, solicitar el acceso a una página web e indicar datos de identificación erróneos en la página de inicio de sesión. Tras ello, debe indicarse que los datos proporcionados no son válidos.
- No habiendo iniciado sesión, dejar en blanco el usuario, la contraseña y/o ambos a la vez. Comprobar que al tratar de iniciar sesión notifica el error.
- No habiendo iniciado sesión, solicitar el acceso a una página web e indicar datos de identificación válidos, estando el usuario con una mezcla de minúsculas y mayúsculas. Tras ello, debe cargarse la página o el mensaje de acceso restringido y notificado. Es decir, la sesión debe haberse iniciado de forma correcta.
- No habiendo iniciado sesión solicitar acceso a una página no permitida. Tras ello, debe indicarse el usuario y la contraseña para comprobar que se muestra la página interna de acceso denegado y notificado.
- No habiendo iniciado sesión solicitar acceso a una página permitida. Tras ello, debe indicarse el usuario y la contraseña para comprobar que se muestra la página solicitada.
- No habiendo iniciado sesión solicitar la descarga de un recurso no permitido. Tras ello, debe indicarse el usuario y la contraseña para comprobar que la descarga no se realiza y notifica de acceso restringido.

Ejemplo de descarga: <http://go.microsoft.com/fwlink/?LinkId=863262>

- No habiendo iniciado sesión solicitar la descarga de un recurso permitido. Tras ello, debe verse como se cancela automáticamente, para a continuación, tras indicar el usuario y la contraseña, ver cómo se ha reanudado y se descarga de forma correcta.

Ejemplo de descarga: <http://go.microsoft.com/fwlink/?LinkId=863262>

- No habiendo iniciado sesión tratar de crear un marcador. Tras ello, debe apreciarse como no se añade y se avisa mediante un mensaje de Windows.

- No habiendo iniciado sesión tratar de acceder a la página de administración de extensiones. Tras ello, debe indicarse el usuario y la contraseña para comprobar que notifica de acceso restringido.
- Una vez iniciada la sesión tratar de realizar al menos una navegación mediante el buscador de Google para a continuación cerrar el navegador. Tras ello, abrir de nuevo el navegador y comprobar que el historial está vacío, así como comprobar, volviendo a iniciar sesión, que en el buscador de Google no se encuentra guardada la búsqueda realizada.
- Una vez iniciada la sesión acceder a la página del campus virtual de la universidad, introducir los datos de identificación de usuario y comprobar que se carga de forma correcta.
- Una vez iniciada la sesión, acceder a una página que requiera de autenticación, introducir los datos de identificación de usuario, elegir la opción de recordar contraseña y sin cerrar la sesión cerrar el navegador. Tras ello, abrir de nuevo el navegador, iniciar sesión y acceder al mismo servicio para comprobar que ni la contraseña ni el usuario están almacenados.
- Habiendo iniciado sesión solicitar acceso a una página no permitida. Tras ello, debe mostrarse la página interna de acceso denegado y notificado.
- Habiendo iniciado sesión solicitar acceso a una página permitida. Tras ello, debe mostrarse la página solicitada.
- Habiendo iniciado sesión solicitar la descarga de un recurso no permitido. Tras ello, debe comprobarse que la descarga no se realiza y notifica de acceso restringido y notificado.
Ejemplo de descarga: <http://go.microsoft.com/fwlink/?LinkId=863262>
- Habiendo iniciado sesión solicitar la descarga de un recurso permitido. Tras ello, debe comprobarse que la descarga se realiza de forma correcta.
Ejemplo de descarga: <http://go.microsoft.com/fwlink/?LinkId=863262>
- Habiendo iniciado sesión tratar de crear un marcador. Tras ello, debe apreciarse como no se añade y te avisa mediante un mensaje de Windows.
- Habiendo iniciado sesión tratar de acceder a la página de administración de extensiones. Tras ello, debe notificar de acceso restringido y notificado.
- Habiendo iniciado sesión, esperar 45 minutos y solicitar acceso a un recurso web. Tras ello, debe solicitarse inicio de sesión.
- Habiendo iniciado sesión, no teniendo ninguna restricción para ese usuario en ese justo momento y habiendo varias restricciones, con prohibiciones diferentes y siendo seguidas, que comenzarán y finalizarán dentro de los 45 minutos posteriores, esperar hasta que finalice la última restricción. Comprobar como notifica del comienzo y fin de cada una de las restricciones, así como admite el acceso a diferentes recursos, permitidos y no, en cada franja de restricciones.
- Sin haber iniciado sesión, teniendo ya una restricción para ese usuario en ese justo momento y habiendo varias restricciones, con prohibiciones diferentes y siendo seguidas, que comenzarán y finalizarán dentro de los 45 minutos posteriores, esperar hasta que finalice la última restricción. Comprobar cómo según se inicia sesión se notifica del inicio de una restricción, así como va notificándose el cambio de las restricciones a lo largo de los 45

minutos. Comprobar también el acceso a diferentes recursos, permitidos y no, en cada franja de restricciones.

- Habiendo iniciado sesión, no teniendo ninguna restricción para ese usuario en ese justo momento y habiendo varias restricciones, con prohibiciones diferentes y dejando huecos sin restricciones, que comenzarán y finalizarán dentro de los 45 minutos posteriores, esperar hasta que finalice la última restricción. Comprobar como notifica del comienzo y fin de cada una de las restricciones, así como admite el acceso a diferentes recursos, permitidos y no, en cada franja de restricciones, y cuando no las hay comprobar como permite navegar.
- Sin haber iniciado sesión, teniendo ya una restricción para ese usuario en ese justo momento y habiendo varias restricciones, con prohibiciones diferentes y dejando huecos sin restricciones, que comenzarán y finalizarán dentro de los 45 minutos posteriores, esperar hasta que finalice la última restricción. Comprobar cómo según se inicia sesión se notifica del inicio de una restricción, así como va notificándose el cambio de las restricciones a lo largo de los 45 minutos. Comprobar también el acceso a diferentes recursos, permitidos y no, en cada franja de restricciones, y cuando no las hay comprobar como permite navegar.
- Sin haber iniciado sesión solicitar el acceso a una web permitida. Tras ello, indicar los datos de identificación, comprobar cómo se carga la página y una vez cargada ir hacia atrás. Debe comprobarse cómo se indica que no hay más páginas hacia donde ir, así como muestra la página de inicio de Google Chrome.
- Sin haber iniciado sesión solicitar el acceso a varias páginas web, permitida-noPermitida-permitida. Tras ello, indicar los datos de identificación, comprobar cómo se cargan las páginas permitidas y cómo se avisa y notifica de las restricciones en las prohibidas. Una vez en la última página, estando ya cargada, ir hacia atrás. Debe mostrarse la primera página permitida hacia la que se navegó. A continuación, volver a ir hacia atrás y comprobar cómo se indica que no hay más páginas hacia donde ir, así como muestra la página de inicio de Google Chrome.
- Habiendo iniciado sesión y sin haber pasado 45 minutos, clicar sobre el icono de la extensión. Debe mostrarse una opción de cerrar sesión. Clicar sobre esa opción y volver a clicar sobre el icono. Debe mostrarse un cuadro vacío. Probar a solicitar el acceso a algún sitio web y comprobar que solicita inicio de sesión.
- Nada más abrir el navegador clicar sobre el icono de la extensión. Comprobar como muestra un cuadro vacío.
- Habiendo iniciado sesión, cerrar el navegador, volver a abrirlo y clicar sobre el icono de la extensión. Comprobar como muestra un cuadro vacío.
- Una vez iniciada la sesión, acceder a una página que requiera de autenticación, introducir los datos de identificación de usuario, elegir la opción de recordar contraseña y cerrar la sesión clicando sobre el icono de la extensión y el botón cerrar sesión. Tras ello, iniciar de nuevo la sesión y acceder al mismo servicio para comprobar que ni la contraseña ni el usuario están almacenados.
- Una vez iniciada la sesión tratar de realizar al menos una navegación mediante el buscador de Google para a continuación cerrar la sesión clicando sobre el icono de la extensión y el

botón cerrar sesión. Tras ello, iniciar de nuevo la sesión y comprobar que el historial está vacío, así como comprobar, que en el buscador de Google no se encuentra guardada la búsqueda realizada.

- Sin haber iniciado sesión tratar de iniciarla con datos de un profesor y/o administrador. Debe indicar que los datos proporcionados son incorrectos.

Los casos de test, **cuando no hay conexión con el servidor y ese día hay o no restricciones**, diseñados y ejecutados son los siguientes:

- No habiendo iniciado sesión, solicitar el acceso a una página web e indicar datos de identificación erróneos en la página de inicio de sesión. Tras ello, debe indicarse que no se puede conectar con el servidor y que se inicia el modo de tiempo de vuelo.
- No habiendo iniciado sesión, dejar en blanco el usuario, la contraseña y/o ambos a la vez. Comprobar que al tratar de iniciar sesión notifica el error de campos en blanco y/o indica que no se puede conectar con el servidor y que se inicia el modo de tiempo de vuelo.
- No habiendo iniciado sesión, solicitar el acceso a una página web de las no permitidas por defecto e indicar datos de identificación válidos, estando el usuario con una mezcla de minúsculas y mayúsculas. Tras ello, debe indicarse que no se puede conectar con el servidor y que se inicia el modo de tiempo de vuelo, mostrando además una página que notifica de la restricción sin notificación.
- No habiendo iniciado sesión, solicitar el acceso a una página web de las permitidas por defecto e indicar datos de identificación válidos, estando el usuario con una mezcla de minúsculas y mayúsculas. Tras ello, debe indicarse que no se puede conectar con el servidor y que se inicia el modo de tiempo de vuelo, mostrando además la página solicitada.
- No habiendo iniciado sesión solicitar la descarga de un recurso no permitido en las URL por defecto. Tras ello, debe indicarse el usuario y la contraseña para comprobar que la descarga no se realiza y notifica de acceso restringido sin notificación asociada.
- No habiendo iniciado sesión solicitar la descarga de un recurso permitido en las URL por defecto. Tras ello, debe verse como se cancela automáticamente, para a continuación, tras indicar el usuario y la contraseña, ver cómo se ha reanudado y se descarga de forma correcta.
- No habiendo iniciado sesión tratar de crear un marcador. Tras ello, debe apreciarse como no se añade y te avisa mediante un mensaje de Windows.
- No habiendo iniciado sesión tratar de acceder a la página de administración de extensiones. Tras ello, debe indicarse el usuario y la contraseña para comprobar que notifica de acceso restringido sin notificación alguna por estar en tiempo de vuelo.
- Una vez iniciada la sesión acceder a la página del campus virtual de la universidad, introducir los datos de identificación de usuario y comprobar que se carga de forma correcta.
- Habiendo iniciado sesión solicitar acceso a una página no permitida en las URL por defecto. Tras ello, debe mostrarse la página interna de acceso denegado, pero no notificado por estar en tiempo de vuelo.
- Habiendo iniciado sesión solicitar acceso a una página permitida en las URL por defecto. Tras ello, debe mostrarse la página solicitada.

- Habiendo iniciado sesión tratar de crear un marcador. Tras ello, debe apreciarse como no se añade y te avisa mediante un mensaje de Windows.
- Habiendo iniciado sesión tratar de acceder a la página de administración de extensiones. Tras ello, debe avisar de acceso restringido sin notificación asociada.
- Habiendo iniciado sesión, esperar 45 minutos y solicitar acceso a un recurso web. Tras ello, debe solicitarse inicio de sesión.
- Sin haber iniciado sesión solicitar el acceso a una web permitida dentro de las URL por defecto. Tras ello, indicar los datos de identificación, comprobar cómo se carga la página y una vez cargada ir hacia atrás. Debe comprobarse cómo se indica que no hay más páginas hacia donde ir, así como muestra la página de inicio de Google Chrome o una página de acceso restringido y no notificado en caso de no estar entre las URL por defecto la página de inicio de Google Chrome.
- Sin haber iniciado sesión solicitar el acceso a varias páginas web, permitida-noPermitida-permitida dentro de las URL por defecto. Tras ello, indicar los datos de identificación, comprobar cómo se cargan las páginas permitidas y cómo se avisa y no notifica de las restricciones en las prohibidas. Una vez en la última página, estando ya cargada, ir hacia atrás. Debe mostrarse la primera página permitida hacia la que se navegó. A continuación, volver a ir hacia atrás y comprobar cómo se indica que no hay más páginas hacia donde ir, así como muestra la página de inicio de Google Chrome o una página de acceso restringido y no notificado en caso de no estar entre las URL por defecto la página de inicio de Google Chrome.
- Habiendo iniciado sesión y sin haber pasado 45 minutos, clicar sobre el icono de la extensión. Debe mostrarse una opción de finalizar tiempo de vuelo. Clicar sobre esa opción y volver a clicar sobre el icono. Debe mostrarse un cuadro vacío. Probar a solicitar el acceso a algún sitio web y comprobar que solicita inicio de sesión.
- Nada más abrir el navegador clicar sobre el icono de la extensión. Comprobar como muestra un cuadro vacío.
- Habiendo iniciado sesión, cerrar el navegador, volver a abrirlo y clicar sobre el icono de la extensión. Comprobar como muestra un cuadro vacío.

Los casos de test, **cuando hay conexión con el servidor y ese día no hay restricciones**, diseñados y ejecutados son los siguientes:

- Tratar de navegar a cualquier URL y comprobar que no solicita inicio de sesión además de mostrarse el contenido solicitado.
- Tratar de añadir un marcador. Comprobar que no muestra notificación emergente pero no deja realizar la tarea.
- Realizar una navegación múltiple y comprobar yendo hacia atrás que funciona de forma correcta esta acción. Sin mostrar notificación emergente al llegar al final del historial de marcha atrás.
- Clicar sobre el icono de la extensión y comprobar que muestra un cuadro en blanco.
- Habiendo navegado por alguna URL cerrar el navegador y volver a abrirlo. Comprobar que en el historial no se guarda dicha información.

- Habiendo accedido a una web que requiere contraseña y marcando los datos de acceso como recordables, cerrar el navegador. Comprobar que al volver a abrirlo y acceder al mismo sitio web no recuerda los datos.
- Solicitar la descarga de un recurso. Tras ello, debe comprobarse que la descarga se realiza de forma correcta.
Ejemplo de descarga: <http://go.microsoft.com/fwlink/?LinkId=863262>
- Estando navegando sin haber solicitado inicio de sesión, por no haber restricciones, crear una restricción para ese mismo día y comprobar como en un periodo máximo de 5 minutos, en cuanto se solicite un recurso web va a pedir inicio de sesión.

1.2. Pruebas con usuarios en entorno reducido

Desde el inicio del desarrollo de la extensión se han estado probando las diferentes versiones en un entorno reducido. Estas pruebas fueron realizadas en una navegación diaria común para observar posibles fallos, así como mejoras en la usabilidad de la extensión.

La primera observación fue referente a la usabilidad en el inicio de sesión. El usuario que realizó la prueba, durante varios días, echaba en falta la posibilidad de al pulsar la tecla “intro” solicitar el inicio de sesión al servidor directamente sin tener que utilizar el ratón.

Lo siguiente observado fue una irregularidad referente a la acción de ir hacia atrás. El problema era que al no almacenar ese historial en una tabla hash propia, sino que se trataba de predecir el número de veces que había que ir hacia atrás para evitar las páginas intermedias de la extensión, a veces no acababa en la página correcta.

Otro problema fue el hecho de utilizar algunas librerías JavaScript y CSS de forma online, lo que ocasionaba que a veces no se cargaran de forma correcta las páginas internas de la extensión si la conexión no era buena. Se solucionó incluyéndolas en local.

En relación con la usabilidad, se observó que tras aplicar la tabla hash en la navegación hacia atrás, el botón del navegador seguía activo, aunque se llegara al final. Esto ocasionaba dudas en el usuario que creía que podía seguir haciéndolo. La solución fue indicar un mensaje emergente.

APLICATIVO NODEJS

2.1. Estructura facilitadora de pruebas

Para facilitar el cambio del entorno de desarrollo / pruebas al entorno de producción se facilita una variable booleana de aplicación que permite pasar de utilizar el LDAP para autenticación a usar los datos por defecto incluidos en la base de datos.

En este sentido también se facilita una variable de aplicación que contiene la ruta a la base de datos MongoDB de forma que para cambiar de desarrollo / pruebas a producción tan solo se necesita cambiar ese valor.

Como entorno de pruebas se utiliza una máquina alojada en AWS y una máquina virtual en local para comprobar cambios en velocidades de transferencia de datos entre la base de datos, la extensión y el aplicativo NodeJS.

2.2. Casos de test

Durante el desarrollo del aplicativo NodeJS se han llevado a cabo multitud de pruebas para comprobar su correcto funcionamiento. Si bien son pruebas que no se han automatizado por la naturaleza de la solución y por limitaciones de tiempo, se han llegado a ejecutar de forma manual para obtener resultados y modificar el desarrollo cuando fuera necesario.

Las pruebas referentes a la API REST al estar ligadas con la prueba del funcionamiento de la extensión no se incluyen aquí por estar en el punto anterior. Lo que se incluye en este apartado son las pruebas del funcionamiento del panel de administración.

Al igual que en los casos de test indicados para la extensión, a la hora de probar estos nuevos casos de test deben darse las condiciones adecuadas para que se puedan ejecutar. Estas condiciones incluyen: tener usuarios cargados de los 3 roles, tener asignaturas, grupos y relaciones entre ellos según se indique en cada test.

Los casos de test diseñados y ejecutados, para **todos los roles**, han sido los siguientes:

- Estando en la página de inicio y sin haber iniciado sesión clicar sobre “Iniciar sesión”. Comprobar que se carga la página de inicio de sesión.
- Intentar iniciar sesión con el identificador de un estudiante. Comprobar que indica que los datos introducidos son incorrectos.
- Iniciar sesión con el identificador de un profesor. Comprobar que carga una página privada con las opciones: “Inicio”, “Gestión de Slots”, “Gestión de Informes” y “Cerrar sesión”.
- Iniciar sesión con el identificador de un administrador. Comprobar que carga una página privada con las opciones: “Inicio”, “Gestión de Grupos”, “Gestión de Informes”, “Gestión de Backups”, “Configuración” y “Cerrar sesión”.

Los casos de test diseñados y ejecutados, para el **rol de profesor**, han sido los siguientes:

- Habiendo iniciado sesión clicar sobre “Cerrar sesión”. Comprobar que te ha dirigido a la página de inicio de sesión y los menús superiores han desaparecido.
- Habiendo iniciado sesión, clicar sobre “Inicio”. Comprobar que se va a la página mostrada tras haber iniciado sesión.
- Habiendo iniciado sesión, clicar sobre “Gestión de Slots”. Comprobar que se abre un menú desplegable con las siguientes opciones: “Nuevo slot” y “Ver slots”.
- Habiendo iniciado sesión, clicar sobre “Gestión de Informes”. Comprobar que se abre un menú desplegable con la opción “Ver listado”.
- Habiendo iniciado sesión, clicar sobre “Gestión de Slots” > “Nuevo slot”. Comprobar que se carga el formulario para dar de alta nuevas restricciones.
- Habiendo iniciado sesión, clicar sobre “Gestión de Slots” > “Nuevo slot” y dejar algunos campos vacíos. Comprobar que resalta los campos como vacíos y no crea la restricción. Realizar esta comprobación con todas las combinaciones posibles.
- Habiendo iniciado sesión, no teniendo grupos asignados, clicar sobre “Gestión de Slots” > “Nuevo slot” y comprobar como en el desplegable de “Asignatura” indica que no tiene disponible ninguna y no deja crear la restricción.
- Habiendo iniciado sesión, clicar sobre “Gestión de Slots” > “Nuevo slot”, rellenar correctamente todos los campos y excluir a todos los grupos. Se debe comprobar como no deja crear la restricción por ese motivo.
- Habiendo iniciado sesión, clicar sobre “Gestión de Slots” > “Nuevo slot”, rellenar correctamente todos los campos y excluir a todos los alumnos. Se debe comprobar como no deja crear la restricción por ese motivo.
- Habiendo iniciado sesión, clicar sobre “Gestión de Slots” > “Nuevo slot” y rellenar correctamente todos los campos. Se debe comprobar como se muestra un listado con todas las restricciones además de un mensaje indicando que el slot se ha creado correctamente.
- Habiendo iniciado sesión, clicar sobre “Gestión de Slots” > “Nuevo slot”, crear un slot para un horario, un grupo y una asignatura en particular, y tratar de volver a crear otro slot dentro de ese mismo horario para todos los grupos de la misma asignatura. Se debe comprobar cómo se crea de forma correcta la restricción mostrando el listado de slots, además de, un mensaje indicando las colisiones y exclusiones automáticas de los alumnos del grupo para el que ya había una restricción. Realizar la misma prueba en franjas horarias que cojan parte del horario de la restricción ya disponible, el resultado debe ser el mismo.
- Habiendo iniciado sesión, clicar sobre “Gestión de Slots” > “Nuevo slot”, crear un slot para un horario, un grupo y una asignatura en particular, y tratar de volver a crear otro slot dentro de ese mismo horario para ese mismo grupo y asignatura. Se debe comprobar cómo no se crea la restricción mostrando el listado de slots, además de, un mensaje indicando las colisiones y que no se crea por encontrarse todos los alumnos excluidos. Realizar la misma prueba en franjas horarias que cojan parte del horario de la restricción ya disponible, el resultado debe ser el mismo.

- Habiendo iniciado sesión, clicar sobre “Gestión de Slots” > “Nuevo slot”, crear un slot para un horario, un grupo y una asignatura en particular con un alumno excluido, y tratar de volver a crear otro slot dentro de ese mismo horario para el mismo grupo y asignatura y dejar solo al alumno excluido anteriormente. Se debe comprobar cómo se crea de forma correcta la restricción mostrando el listado de slots además de un mensaje indicando que se realizó la operación de forma correcta.
- Habiendo iniciado sesión, clicar sobre “Gestión de Slots” > “Nuevo slot” y crear un slot para una asignatura y un grupo para el que no haya alumnos asociados. Comprobar cómo indica que no hay alumnos para los grupos indicados y no deja crear la restricción.
- Habiendo iniciado sesión, clicar sobre “Gestión de Slots” > “Nuevo slot” y rellenar la parte de URL con un formato diferente al indicado. Comprobar como muestra un mensaje de error al tratar de añadirla.
- Habiendo iniciado sesión y no teniendo restricciones aún creadas para ninguna asignatura donde el profesor identificado imparte alguno de los grupos, clicar sobre “Gestión de Slots” > “Ver slots” y comprobar que indica que no hay registros para mostrar.
- Habiendo iniciado sesión y teniendo restricciones creadas para alguna de las asignaturas donde el profesor identificado imparte alguno de los grupos, clicar sobre “Gestión de Slots” > “Ver slots” y comprobar que se muestra el listado, al menos, junto a un botón “Ver detalles”.
- Habiendo iniciado sesión y teniendo restricciones creadas con fecha de fin posterior a la fecha actual. Ir a “Gestión de Slots” > “Ver slots” y comprobar como aparecen en el listado junto a las opciones: “Eliminar”, “Modificar” y “Ver detalles”.
- Habiendo iniciado sesión y teniendo restricciones creadas con fecha de fin anterior a la fecha actual. Ir a “Gestión de Slots” > “Ver slots” y comprobar como aparecen en el listado solamente junto a la opción “Ver detalles”.
- Habiendo iniciado sesión y teniendo más de 10 restricciones creadas. Ir a “Gestión de Slots” > “Ver slots” y probar a utilizar los diferentes filtros, ordenaciones y la paginación. Debe mostrarse el contenido esperado.
- Habiendo iniciado sesión y teniendo al menos 1 restricción creada. Ir a “Gestión de Slots” > “Ver slots” y clicar sobre “Ver detalles” de cualquiera de las restricciones. Debe mostrarse una ventana con los datos particulares de la misma. Así mismo clicando a continuación sobre cualquier parte de la ventana, externa al cuadro sobre el que se muestra la información, debe cerrarse dicha ventana. Comprobar el mismo comportamiento clicando sobre el símbolo “X”.
- Habiendo iniciado sesión y teniendo al menos 1 restricción creada con fecha de fin posterior al momento actual. Ir a “Gestión de Slots” > “Ver slots” y clicar sobre “Modificar” de cualquiera de las restricciones disponibles con esta opción. Debe mostrarse un formulario idéntico al que se muestra cuando se dan de alta las restricciones con la excepción de que tendrá ya los campos rellenados y no deja cambiar la asignatura.
- Habiendo iniciado sesión y teniendo al menos 1 restricción creada con fecha de fin posterior al momento actual. Ir a “Gestión de Slots” > “Ver slots” y clicar sobre “Modificar” de cualquiera de las restricciones disponibles con esta opción. Tras ello, dejar campos vacíos y/o inválidos y probar a modificarla. Deben mostrarse los errores producidos y no modificarse.

- Habiendo iniciado sesión y teniendo al menos 1 restricción creada con fecha de fin posterior al momento actual. Ir a “Gestión de Slots” > “Ver slots” y clicar sobre “Modificar” de cualquiera de las restricciones disponibles con esta opción. Tras ello, cambiar campos, de forma correcta, y comprobar cómo se redirige a la página con el listado de slots indicando que se ha modificado de forma correcta. Se recomienda hacer las mismas comprobaciones que para la creación de slots.
- Habiendo iniciado sesión y teniendo al menos 1 restricción creada con fecha de fin posterior al momento actual. Ir a “Gestión de Slots” > “Ver slots” y clicar sobre “Eliminar” de cualquiera de las restricciones disponibles con esta opción. Tras ello, comprobar que se muestra un mensaje emergente solicitando confirmación de la operación.
- Habiendo iniciado sesión y teniendo al menos 1 restricción creada con fecha de fin posterior al momento actual. Ir a “Gestión de Slots” > “Ver slots” y clicar sobre “Eliminar” de cualquiera de las restricciones disponibles con esta opción. Tras ello, clicar sobre la opción “No”, “X” o sobre cualquier parte de la ventana sobre la que no esté el mensaje emergente. Debe seguir todo tal cual estaba sin eliminar.
- Habiendo iniciado sesión y teniendo al menos 1 restricción creada con fecha de fin posterior al momento actual. Ir a “Gestión de Slots” > “Ver slots” y clicar sobre “Eliminar” de cualquiera de las restricciones disponibles con esta opción. Tras ello, clicar sobre la opción “Sí”. Debe cargarse el listado de restricciones sin estar la eliminada, además de mostrarse un mensaje confirmando su eliminación de forma correcta.
- Habiendo iniciado sesión y teniendo al menos 1 restricción creada. Ir a “Gestión de Informes” > “Ver listado”. Comprobar que se muestra un desplegable con los nombres de las restricciones disponibles.
- Habiendo iniciado sesión y no teniendo restricciones creadas. Ir a “Gestión de Informes” > “Ver listado”. Comprobar que se muestra en el desplegable que no hay restricciones.
- Habiendo iniciado sesión y teniendo al menos 1 restricción creada. Ir a “Gestión de Informes” > “Ver listado” y seleccionar una restricción que no tenga notificaciones. Comprobar que se muestra una página indicando que no hay registros para mostrar.
- Habiendo iniciado sesión y teniendo al menos 1 restricción creada. Ir a “Gestión de Informes” > “Ver listado” y seleccionar una restricción que tenga notificaciones solo de inicios de sesión de alumnos. Comprobar que se muestra una página con un listado de alumnos, en negro, junto a un resumen de sus notificaciones.
- Habiendo iniciado sesión y teniendo al menos 1 restricción creada. Ir a “Gestión de Informes” > “Ver listado” y seleccionar una restricción que tenga notificaciones de inicios de sesión de alumnos e infracciones. Comprobar que se muestra una página con un listado de alumnos, en negro o rojo en función de que haya infracciones o no, junto a un resumen de sus notificaciones.
- Habiendo iniciado sesión y teniendo al menos 1 restricción creada. Ir a “Gestión de Informes” > “Ver listado” y seleccionar una restricción que tenga notificaciones de inicios de sesión e infracciones de 10 o más alumnos. Comprobar que se muestra una página con un listado de alumnos, en negro o rojo en función de que haya infracciones o no, junto a un resumen de

sus notificaciones. A continuación, filtrar, ordenar e ir entre las diferentes páginas comprobando que se muestra todo como debería.

- Habiendo iniciado sesión y teniendo al menos 1 restricción creada. Ir a “Gestión de Informes” > “Ver listado” y seleccionar una restricción que tenga notificaciones de inicios de sesión de alumnos e infracciones. Clicar sobre una de las opciones “Ver detalles” de alguno de los alumnos y comprobar que se abre una ventana emergente mostrando cada una de las notificaciones de forma individual. Probar también el funcionamiento de los diferentes filtros, ordenación y paginación.
- Habiendo iniciado sesión y teniendo al menos 1 restricción creada. Ir a “Gestión de Informes” > “Ver listado” y seleccionar una restricción que tenga notificaciones de inicios de sesión de alumnos e infracciones. Clicar sobre una de las opciones “Ver detalles” de alguno de los alumnos y comprobar que al clicar sobre la opción “X” o sobre cualquiera de las partes, externas al mensaje emergente, se cierra la ventana emergente.

Los casos de test diseñados y ejecutados, para el **rol de administrador**, han sido los siguientes:

- Habiendo iniciado sesión clicar sobre “Cerrar sesión”. Comprobar que te ha dirigido a la página de inicio de sesión y los menús superiores han desaparecido.
- Habiendo iniciado sesión, clicar sobre “Inicio”. Comprobar que se va a la página mostrada tras haber iniciado sesión.
- Habiendo iniciado sesión clicar sobre “Gestión de Grupos”. Comprobar que se abre un menú desplegable con la opción “Carga por fichero”.
- Habiendo iniciado sesión clicar sobre “Gestión de Informes”. Comprobar que se abre un menú con las opciones: “Notificaciones sin slot asociado” y “Señales de vida de la extensión”.
- Habiendo iniciado sesión clicar sobre “Gestión de Backups”. Comprobar que se abre un menú con las opciones: “Restaurar Backup” y “Eliminar Backup”.
- Habiendo iniciado sesión clicar sobre “Configuración”. Comprobar que se abre un menú con la opción “Configuraciones”.
- Habiendo iniciado sesión clicar sobre “Gestión de Grupos” > “Carga por fichero”. Comprobar que se muestra una ventana con opción a subir 2 ficheros y cargar ambos con un formato incorrecto. Debe avisar del error y no cargarlos.
- Habiendo iniciado sesión clicar sobre “Gestión de Grupos” > “Carga por fichero”. Comprobar que se muestra una ventana con opción a subir 2 ficheros y cargar ambos con un formato correcto. Debe cargar ambos y crear una copia de seguridad del estado anterior.
- Habiendo iniciado sesión clicar sobre “Gestión de Grupos” > “Carga por fichero”. Comprobar que se muestra una ventana con opción a subir 2 ficheros y cargar uno bien y el otro mal. Debe avisar del error y no cargar ninguno.
- Habiendo iniciado sesión clicar sobre “Gestión de Grupos” > “Carga por fichero”. Comprobar que se muestra una ventana con opción de subir 2 ficheros y cargar solamente el de profesores de forma correcta. Debe cargarse, crear una copia de seguridad y han de encontrarse disponibles los nuevos datos de haberlos.

- Habiendo iniciado sesión clicar sobre “Gestión de Grupos” > “Carga por fichero”. Se ha de tener ya la estructura de profesores-grupos cargada previamente. Comprobar que se muestra una ventana con opción de subir 2 ficheros y cargar solamente el de alumnos de forma correcta. Debe cargarse, crear una copia de seguridad y han de encontrarse disponibles los nuevos datos de haberlos.
- Habiendo iniciado sesión clicar sobre “Gestión de Grupos” > “Carga por fichero”. No se ha de tener la estructura de profesores-grupos cargada previamente, debe estar sin nada. Comprobar que se muestra una ventana con opción de subir 2 ficheros y cargar solamente el de alumnos de forma correcta. Debe indicar un error al no encontrarse la estructura de profesores-grupos.
- Habiendo iniciado sesión clicar sobre “Gestión de Grupos” > “Carga por fichero”. Sin añadir ningún fichero tratar de clicar sobre Cargar Ficheros. Debe mostrar un error.
- Habiendo iniciado sesión y no teniendo datos de notificaciones, clicar sobre “Gestión de Informes” > “Notificaciones sin slot asociado”. Debe mostrarse una pantalla que indica que no hay registros.
- Habiendo iniciado sesión y teniendo datos de notificaciones que no tengan slot asociado, por ejemplo, inicios de sesión no teniendo restricción aplicada, clicar sobre “Gestión de Informes” > “Notificaciones sin slot asociado”. Debe mostrarse una pantalla que indica los diferentes registros disponibles por alumno.
- Habiendo iniciado sesión y teniendo datos de notificaciones, de 10 o más alumnos diferentes, que no tengan slot asociado, por ejemplo, inicios de sesión no teniendo restricción aplicada, clicar sobre “Gestión de Informes” > “Notificaciones sin slot asociado”. Debe mostrarse una pantalla que indica los diferentes registros disponibles. A continuación, probar a ordenar, filtrar, paginar y buscar por los diferentes campos para comprobar el correcto funcionamiento de las herramientas de visualización.
- Habiendo iniciado sesión y teniendo datos de notificaciones que no tengan slot asociado, por ejemplo, inicios de sesión no teniendo restricción aplicada, clicar sobre “Gestión de Informes” > “Notificaciones sin slot asociado”. Debe mostrarse una pantalla que indica los diferentes registros disponibles junto a una opción “Ver detalles”. Clicar sobre esa opción y comprobar que se muestran las notificaciones disponibles para ese alumno. Comprobar también que la fecha máxima coincide con la mostrada en la ventana anterior, así como el número de notificaciones y si tiene acciones prohibidas o no, además de probar los diferentes filtros de configuración que ofrece esta nueva ventana.
- Habiendo iniciado sesión y teniendo datos de notificaciones que no tengan slot asociado y sean acciones prohibidas, por ejemplo, deshabilitar la extensión no teniendo restricción aplicada, clicar sobre “Gestión de Informes” > “Notificaciones sin slot asociado”. Deben estar los alumnos con notificaciones de infracciones en rojo, así como el indicador de acciones prohibidas a “Sí”. Clicar sobre “Ver detalles” y comprobar que hay al menos alguna infracción entre las diversas notificaciones.

- Habiendo iniciado sesión y no teniendo datos de señales de vida, clicar sobre “Gestión de Informes” > “Señales de vida de la extensión”. Debe mostrarse una pantalla que indica que no hay registros.
- Habiendo iniciado sesión y teniendo datos de señales de vida, clicar sobre “Gestión de Informes” > “Señales de vida de la extensión”. Debe mostrarse una pantalla que indica para cada IP externa las interfaces internas propias de un ordenador junto al número de señales de vida recibidas, la fecha de la última señal, así como un indicador que avisa si es necesario revisar la extensión en ese ordenador.
- Habiendo iniciado sesión y teniendo datos de señales de vida de 10 o más ordenadores, clicar sobre “Gestión de Informes” > “Señales de vida de la extensión”. Deben mostrarse los diferentes registros. A continuación, aplicar filtros, búsquedas, ordenaciones y paginaciones para comprobar que funcionan de forma correcta el conjunto de herramientas de visualización.
- Habiendo iniciado sesión y teniendo datos de señales de vida, clicar sobre “Gestión de Informes” > “Señales de vida de la extensión”. Comprobar que se muestran los registros junto a una opción “Ver detalles”. Clicar sobre esa opción y comprobar que se muestra el número de señales de vida esperadas, así como que coinciden la fecha máxima y el número de notificaciones con lo mostrado en la ventana anterior.
- Habiendo iniciado sesión y teniendo datos de señales de vida a revisar, es decir, señales de vida con fecha máxima por debajo de la media del resto de equipos, clicar sobre “Gestión de Informes” > “Señales de vida de la extensión”. Deben mostrarse los diferentes registros que cumplan la condición junto al indicador “Sí” en la columna “¿Necesario revisar?”.
- Habiendo iniciado sesión y teniendo datos de señales de vida en tiempo de vuelo, clicar sobre “Gestión de Informes” > “Señales de vida de la extensión”. A continuación, clicar sobre la opción “Ver detalles” del registro que cumpla las condiciones y comprobar que dispone de notificaciones con la columna “Tiempo de vuelo” en el estado “Activado”.
- Habiendo iniciado sesión y no teniendo backups, clicar sobre “Gestión de Backups” > “Restaurar Backup”. Debe indicarse en el desplegable que no hay backups. Comprobar también que el botón restaurar está desactivado.
- Habiendo iniciado sesión y teniendo backups, clicar sobre “Gestión de Backups” > “Restaurar Backup”. Debe indicarse en el desplegable que se debe elegir una opción. Comprobar también que el botón restaurar está desactivado mientras no se selecciona ninguna opción de backup.
- Habiendo iniciado sesión y teniendo backups, clicar sobre “Gestión de Backups” > “Restaurar Backup”. Elegir una opción en el desplegable y comprobar que el botón restaurar está activado. A continuación, clicar sobre el botón y comprobar que se muestra un mensaje indicando que se ha finalizado el proceso de forma correcta, así como aparece un nuevo backup disponible con la fecha actual.
- Habiendo iniciado sesión y no teniendo backups, clicar sobre “Gestión de Backups” > “Eliminar Backup”. Debe indicarse en el desplegable que no hay backups. Comprobar también que el botón eliminar está desactivado.

- Habiendo iniciado sesión y teniendo backups, clicar sobre “Gestión de Backups” > “Eliminar Backup”. Debe indicarse en el desplegable que se debe elegir una opción. Comprobar también que el botón eliminar está desactivado mientras no se selecciona ninguna opción de backup.
- Habiendo iniciado sesión y teniendo backups, clicar sobre “Gestión de Backups” > “Eliminar Backup”. Elegir una opción en el desplegable y comprobar que el botón eliminar está activado. A continuación, clicar sobre el botón y comprobar que se muestra un mensaje indicando que se debe seleccionar el checkbox de confirmar borrado.
- Habiendo iniciado sesión y teniendo backups, clicar sobre “Gestión de Backups” > “Eliminar Backup”. Elegir una opción en el desplegable y comprobar que el botón eliminar está activado. A continuación, clicar sobre el botón y comprobar que se muestra un mensaje indicando que se debe seleccionar el checkbox de confirmar borrado.
- Habiendo iniciado sesión y teniendo backups, clicar sobre “Gestión de Backups” > “Eliminar Backup”. Elegir una opción en el desplegable y comprobar que el botón eliminar está activado. Marcar el checkbox de confirmar borrado y clicar sobre el botón de eliminar. Comprobar que se muestra un mensaje indicando que el proceso se ha llevado a cabo de forma correcta, así como ya no aparece en el desplegable dicho backup.
- Habiendo iniciado sesión clicar sobre “Configuración” > “Configuraciones”. Comprobar que se abre una ventana con un campo selector dando la posibilidad de cambiar el número de días que permanecen las notificaciones en tiempo de vuelo. Debe indicar 24 si no se ha modificado nunca.
- Habiendo iniciado sesión clicar sobre “Configuración” > “Configuraciones”. En el selector intentar poner un número negativo. Debe quitarse automáticamente o dar error.
- Habiendo iniciado sesión clicar sobre “Configuración” > “Configuraciones”. En el selector intentar poner un número mayor a 24. Debe quitarse automáticamente o dar error.
- Habiendo iniciado sesión clicar sobre “Configuración” > “Configuraciones”. En el selector poner un valor diferente del actual y clicar sobre “Guardar cambios”. Debe mostrarse un mensaje indicando que la acción se ha llevado a cabo de forma correcta.



PRUEBAS CON USUARIOS EN ENTORNO FINAL

3.1. Descripción de la prueba

La prueba planteada consistía en probar el sistema en un examen práctico de bases de datos de la facultad de informática de Oviedo. No obstante, debido al COVID-19 y al no desarrollarse el curso de manera presencial, ha sido imposible realizar hasta este momento las pruebas indicadas. Por ello, esta labor se dejará para más adelante.

3.2. Análisis de resultados obtenidos

Al no poder realizar la prueba no se dispone de estos datos.

Capítulo 5 MEDIDAS DE SEGURIDAD



MEDIDAS DE SEGURIDAD GLOBALES

Al tratarse de un proyecto cuyo fin es el de aumentar la seguridad de ciertos sistemas críticos de tratamiento y/o almacenamiento de la información, a lo largo del mismo se han aplicado varias medidas de seguridad, tanto de forma global al conjunto de los aplicativos que conforman el sistema software como a nivel particular en cada una de las aplicaciones que intervienen en el proceso.

A nivel global principalmente son acciones realizadas en el proceso de despliegue para asegurar el correcto funcionamiento del sistema y que no se puedan saltar las restricciones aplicadas.

Al margen de las configuraciones en el servidor y los clientes, como medida de seguridad para dificultar que ataquen el sistema software realizado, se ha procedido a ofuscar el código y quitar los comentarios en la versión desplegada de la extensión, de forma que se dificulte la tarea de encontrar vulnerabilidades.

1.1. Políticas locales en los clientes

Se utilizan políticas locales y por directorio activo para bloquear acciones en Google Chrome y el sistema operativo.

En el sistema operativo son las propias que la Universidad de Oviedo aplica mediante su dominio, las cuales impiden a los estudiantes modificar programas y acceder a partes que requieren permisos de administrador dentro del sistema operativo. En concreto es importante que se impida el cambio de hora en los ordenadores cliente para impedir que haya problemas al indicar el tiempo en las notificaciones, ya que las restricciones se lanzan en función de la hora dada por el servidor.

Como se va a ver en los siguientes puntos, dentro del presente capítulo, la corrección de las notificaciones se comprueba mediante un pequeño sistema que simula ser un sistema de detección de intrusos (IDS), el cual indica posibles irregularidades en las mismas en función de diversos aspectos contemplados.

Para la gestión de Google Chrome se aplican las siguientes:

- Impedir finalizar procesos en el administrador de tareas interno de Google Chrome que podría hacer que el usuario en menos de 5 segundos desactive la extensión.
- Instalar la extensión como forzada para evitar que el usuario pueda desinstalarla y/o deshabilitarla.
- La extensión se localiza en la Chrome Web Store para que, de esta forma si el usuario accede a la carpeta donde el navegador almacena las extensiones, sea el propio Chrome quien en cada arranque compruebe la integridad de las mismas, ya que si se hace desde el directorio activo proveyéndola fuera de la Chrome Web Store esta comprobación no se realiza y el usuario podría modificar el funcionamiento de la extensión.

- Como capa de seguridad adicional a lo anterior se ha cambiado la localización por defecto de la carpeta de perfil de Chrome a otra menos “intuitiva” para dificultar al usuario el encontrar una forma de saltarse las restricciones aplicadas por la extensión. La ruta pasa de ser la original “%appdata%/local/Google/Chrome” a “%appdata%/Adobe/ManageReader/Flash Player/.jdk/.logs/.errors”. Aunque podría haberse cambiado a otra ruta que no sea %appdata%, no se ha podido hacer ya que Chrome necesita que el usuario actual tenga permisos de lectura/escritura sobre la misma, y teniendo en cuenta que el contenido de dicha ruta no se indexa, el usuario no podrá realizar búsquedas a ver si da con la localización.
- Se bloquea el acceso a la Chrome Web Store, así como a conexiones ftp, sftp y ssh entre otras desde el propio navegador con el fin de evitar que el usuario pueda instalar nuevas extensiones que puedan tratar de evadir la que aplica las restricciones y para asegurarse de que no consiga establecer conexiones no permitidas mediante el navegador en un examen.
- Evitar el uso del modo incógnito, que, aunque es soportado por la extensión, bajo ciertas situaciones podría hacer que la misma no funcionara correctamente al aplicar Chrome una especie de sandbox con los procesos y la memoria.
- Se impide el inicio de sesión para evitar problemas que surgen con el almacenamiento del historial y marcadores, ya que no se permite para que el usuario no pueda acceder a páginas propias de la extensión.
- No se permite el uso de herramientas para desarrolladores en el navegador para que no puedan acceder a la memoria de la extensión, que, aunque está protegida, podría darse alguna situación en la que pudieran acceder.
- Se evita que puedan arrancar el navegador Chrome con argumentos que disminuyan la seguridad de su sistema e incluso llegando a deshabilitar para ese arranque el funcionamiento de las extensiones.
- Además de las anteriores se aplican otras que, no teniendo que ver directamente con el funcionamiento del presente sistema software, guardan mucha relación con la privacidad y la seguridad en los sistemas de almacenamiento y tratamiento de la información, como es el auto relleno de formularios, guardado de contraseñas, predicción de búsquedas...

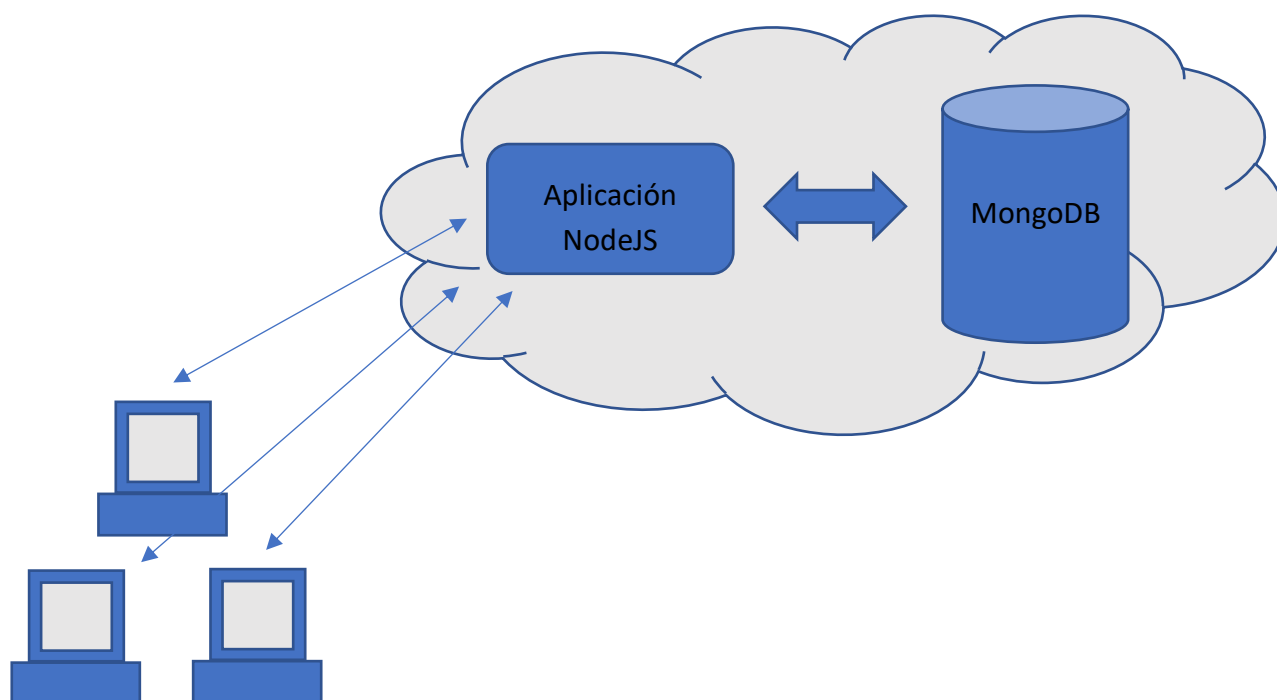
1.2. Configuración del servidor

Se utiliza https para que no se puedan capturar ni modificar los datos enviados y recibidos por la API REST y la aplicación de administración. Para ello se ha tenido que crear una autoridad certificadora que permita firmar un certificado autogenerado. De esta forma los navegadores reconocen el aplicativo como un sitio seguro y no aplican las restricciones en las comunicaciones, como es el bloquear comunicaciones desde JavaScript a una API REST que no tenga un certificado SSL correcto. Ciertamente se podría haber utilizado una autoridad certificadora como Let's Encrypt que es gratuita, pero no sirve al tratarse de un servidor localizado de forma interna en la Universidad. La propia autoridad certificadora creada ha tenido que añadirse al conjunto de entidades certificadoras de los diferentes ordenadores cliente.

Al encontrarse dentro del mismo aplicativo NodeJS la API REST y los endpoints, el puerto utilizado para ambos es el mismo, por lo que para evitar que un usuario cliente, al observar comunicaciones entrantes/salientes a la API REST pueda llegar a acceder al aplicativo de administración, se hace uso de Nginx como reverse proxy, lo que permite que desde fuera se vean dos puertos, pero internamente sea el mismo. De esta forma el usuario tendrá más difícil llegar a la aplicación de administración que además está protegida por usuario, contraseña y 2 roles con permisos diferentes.

Como configuraciones más generales, se actualiza la máquina, se eliminan los servicios innecesarios y se crea un usuario exclusivo para correr la aplicación del lado del servidor. También se activa el firewall UFW de Ubuntu para permitir solamente los dos puertos disponibles y se configura mongo para que solicite una contraseña al trabajar con él, ya que por defecto no lo hace. Adicionalmente a esto, a mongo solamente se puede acceder internamente desde la aplicación NodeJS, no se puede conectar directamente desde el exterior.

Debido a que, como ya se indicó en el punto anterior, las restricciones se aplican según el horario proporcionado por el servidor para evitar un posible fraude por parte de un cambio de hora en los clientes, el servidor debe tener la hora correctamente actualizada a la hora española ya que será la que se utilice. También debe configurarse el DST de forma automática para permitir el cambio de hora.



MEDIDAS DE SEGURIDAD EN LA EXTENSIÓN

La extensión, al tratarse del aplicativo utilizado por los clientes y el más propenso a ataques para anular sus restricciones, ha sido una parte de especial cuidado en cuanto a medidas de seguridad a aplicar se refiere.

Las medidas en materia de seguridad aplicadas son las siguientes:

- Se estructura la implementación de la extensión de forma que la parte que el usuario ve se comunica con la parte interna mediante un sistema de mensajes que provee el propio Chrome. De esta forma el usuario nunca ve la manera de enviar información al servidor ni puede modificar la forma en la que se hace mediante el inspector de desarrolladores, tan solo se ven los HTML mostrados y los JavaScript que se comunican con la parte interna.
- Al almacenarse en una caché interna las diferentes URL a las que tiene o no acceso el usuario, una vez recibida la información de la API REST se genera una hash de dicha información que permite garantizar la integridad del contenido cada vez que se consulta. No obstante, el usuario tras haber conseguido saltarse las medidas de seguridad que impone Chrome en su sistema de memoria podría llegar a darse cuenta de la existencia de esa comprobación y generar por el mismo una nueva hash, para a continuación almacenarla en el lugar correspondiente, pero ya se estaría aumentando el tiempo que requeriría el saltarse las restricciones.
- Se obtienen las direcciones IP de las diferentes interfaces de red de los clientes para tratar de identificar dentro de la red interna los dispositivos desde los que se conectan los usuarios. Esta información se vincula con el token de forma que, si se emite una notificación con un token y se envían unas direcciones IP internas diferentes, se indicará en el panel de administración.
- Se utiliza una tabla hash para controlar la navegación hacia atrás del usuario, de esta forma se evita que pueda ir hacia una página interna de la extensión en un momento que no deba y con ello llegar a analizar parte de su funcionamiento.
- Se impide el acceso a las páginas internas de Google Chrome como el historial y la página de extensiones por defecto.
- Se comprueba el formato de la URL con el objeto URL de JavaScript para obtener el dominio al que hace referencia, de esta forma se evita que inyecten información indebida. También es la forma en la que se bloquean por defecto las URL internas de Chrome, que al no tener un formato estándar no las admite.
- Se envía cada 10 minutos una notificación al servidor para indicar que está en funcionamiento lo cual se analiza desde el perfil de administrador. Esto permite saber al administrador si una extensión es posible que lleve tiempo sin funcionar por un fallo, manipulación o simplemente porque no se ha usado ese equipo.
- La extensión al iniciarse comprueba si hay alguna restricción para ese día de forma que si no la hay el usuario no tendría ningún tipo de restricción y usaría el navegador como si no

estuviera. De esta forma además de evitar al usuario iniciar sesión en un día que no haya exámenes, se evita el que sepa que hay algo antes del examen correspondiente. Dentro de este sistema, la extensión comprueba cada 5 minutos que siga no habiendo restricciones ese día para en caso contrario solicitar credenciales y comenzar a funcionar.

- Cuando el usuario inicia sesión en la extensión se notifica para que se pueda llevar un control de los usuarios que se han conectado. También se hace cuando el usuario conectado cambia a una nueva restricción.
- La extensión recibe el tiempo de la API REST para evitar que el usuario, cambiando la hora del ordenador cliente, evite las restricciones aplicadas. Para ello se almacena en una caché la hora recibida por el servidor y la diferencia con la hora del ordenador cliente, de esta forma se realiza la diferencia siempre que se tenga que indicar la hora en una notificación. No obstante, como se programa el comienzo y fin de las restricciones en el momento en el que se reciben del servidor, esta programación se realiza utilizando solamente la hora del servidor. Por ello, en caso de que el usuario consiguiera cambiar la hora en el ordenador cliente después de haber iniciado sesión, las notificaciones tendrían la hora de ocurrencia mal. Otra protección que se ha aplicado en este sentido es el de generar una hash a partir de la hora recibida para de esta forma comprobar su integridad cada vez que se solicita la información de hora almacenada en caché. También se recibe del servidor el tiempo que tarda en caducar el token para que en ese momento la extensión lo elimine junto a la información de URL almacenada en caché, por lo que se le vuelve a solicitar inicio de sesión al usuario tras ese tiempo que está establecido en 45 minutos.
- Se añade un modo tiempo de vuelo a la extensión que en caso de no estar disponible el servidor solamente se permitirá el acceso a la página web de la universidad. De esta forma se evita que un usuario interrumpa la conexión con la API REST para evitar las restricciones y además se permite a los usuarios seguir utilizando el navegador de forma limitada en caso de error en la conexión.
- Por seguridad de la información almacenada en Caché por la extensión esta información se elimina cada vez que el usuario cierra sesión o abre el navegador de nuevo. Además, como complemento se elimina el resto de información del navegador, para evitar que otro usuario que venga detrás pueda acceder a recursos en los que se haya conectado el anterior.
- Las notificaciones en caso de error al notificarse al servidor se intentan enviar 3 veces más, pero de no conseguirlo tras esos intentos, se almacenan en una caché y se intentan enviar en el siguiente inicio de sesión, tal y como pasa con las notificaciones que se producen durante el modo tiempo de vuelo.

MEDIDAS DE SEGURIDAD EN APLICATIVO NODEJS

El aplicativo NodeJS es el encargado de gestionar la API REST y el panel de administración para profesores y administradores, es por lo que es como si estuviera formado a su vez por dos aplicativos cada uno con algunas medidas de seguridad diferentes en función de su fin.

3.1. Seguridad global

Como principal medida de seguridad y estabilidad se ha decidido utilizar el framework “Express” para NodeJS que provee diversos sistemas para evitar ataques que se suelen producir en aplicaciones web.

Se han elegido librerías con un número suficiente de valoraciones y comprobando que la versión utilizada no tiene reportada ninguna brecha de seguridad en ese momento. En relación con esto, se dejó de utilizar la versión 2.4 de mongo y se pasó a utilizar la 3.3.3, lo mismo ocurrió con la librería Swig que dejó de utilizarse al no tener ninguna versión más reciente sin vulnerabilidades, fue sustituida por la librería Nunjucks.

Se utilizan 3 roles en todo el aplicativo: “student”, “professor” y “administrator”. Como se verá en los siguientes puntos, el primero es solamente utilizado en la API REST y los otros dos en el panel de administración de restricciones.

También se registran en un log las diferentes peticiones a la API REST y al panel de administración. En este log siempre se almacena el nombre de usuario y la IP externa.

Se dispone de un sistema de backup que en caso de cargar nuevos ficheros por parte del administrador del sistema permita no perder los datos antiguos. Además, permite que se pueda restaurar una copia de seguridad anterior y/o eliminarla.

La identificación se realiza mediante el LDAP de la universidad que da más seguridad que un sistema propio al ser exclusivamente dedicado para ello.

3.2. Seguridad en la API REST

Se utiliza un token que se provee en el inicio de sesión, el cual debe ser enviado junto a cualquier futura petición. El mismo está encriptado y contiene el nombre de usuario, las IP internas y externa del cliente que ha solicitado el inicio de sesión y la hora en la que se genera, para de esta forma invalidarlo tras 45 minutos.

La encriptación del token se realiza añadiendo una marca de tiempo del día en el que se solicita el inicio de sesión, de forma que si un atacante consigue descifrar la clave de cifrado solamente le servirá para el día en el que obtuvo el token y no para los posteriores, ya que la clave cambia cada día.

A la API REST al igual que en el panel de administración se concede acceso mediante roles, en cuyo caso solamente pueden acceder los estudiantes, es decir los usuarios que disponen del rol “student”.

Al recibir notificaciones se comprueba que las IP internas y externas de la notificación coincidan, de haberlo, con las del token incluido en la petición. En caso contrario se marca como problema en la notificación para que el profesor o administrador lleve a cabo las acciones que considere oportunas, no obstante, si es una notificación en tiempo de vuelo no se considera este aspecto al cargarse esas notificaciones con el siguiente usuario que inicia sesión en el sistema. Lo mismo se realiza con el nombre de usuario que se comprueba que coincida con el incluido token y con los tiempos de subida de la notificación y de creación de la misma, que en caso de no ser parecidos lo marca como posible problema.

3.3. Seguridad en el panel de administración

Se realiza una gestión de roles en la que se da acceso a funciones diferentes dentro del panel en función de que el usuario sea un profesor o un administrador. Los roles son “professor” y “administrator”.

Se analizan los tiempos de vida enviados por las extensiones en función de sus IP externas y las IP internas, lo que permite realizar una media de la última notificación recibida y para todas aquellas extensiones que estén por debajo señalarlas como revisables ya que puede que no estén funcionando correctamente.

Para las diferentes operaciones realizadas en función del rol, se comprueba siempre que el usuario tenga permisos para realizar esa tarea, por ejemplo, si se trata de un profesor que trata de eliminar una restricción, la misma debe estar asociada a un grupo de una asignatura en la que es profesor. En esta misma línea también se comprueba si se trata de eliminar una restricción pasada, ya que el sistema no lo permite y si el usuario intenta esquivar la restricción la operación no se llegará a realizar. Resumiendo, se restringen las operaciones en el lado del cliente, pero se comprueban siempre en el lado del servidor para evitar “trampas” por parte del usuario.

Se comprueban todas las entradas del usuario reconstruyéndolas y/o codificándolas con caracteres solamente admitidos en URL para evitar inyección de código, XSS u otras técnicas para intentar corromper el sistema. También se limita en este sentido el tamaño de las entradas, en el caso de las URL se utiliza el objeto URL de JavaScript para comprobar que tenga un formato correcto y entre otras comprobaciones se chequea que existan los datos en caso de ser opciones para escoger.

La sesión del usuario está encriptada y tiene una duración, hasta su expiración, de 45 minutos lo que resulta de utilidad si el usuario se deja la sesión abierta o si alguien consigue robar su sesión. Dentro de ella se almacena el nombre de usuario y su rol.

Capítulo 6 PROCESO DE DESPLIEGUE



SITUACIÓN IDEAL

1.1. Requisitos y configuración de los clientes

Los clientes deben estar configurados con el sistema operativo Windows en una versión que soporte las políticas por directorio activo necesarias. No obstante, aunque la casuística no se ha analizado y probado en profundidad, es posible que se pueda desplegar en cualquier sistema operativo que soporte el navegador Google Chrome, cambiando para ello la forma de aplicar las políticas indicadas en la presente memoria, y las cuales son requisito para que un usuario no pueda saltarse las restricciones impuestas. También ha de tenerse en cuenta que no todas las funciones que ofrecen las librerías de Chrome están disponibles en todos los sistemas operativos, por lo que debiera comprobarse su funcionamiento.

La versión de Google Chrome mínima necesaria para que funcionen tanto las políticas como las funcionalidades implementadas, haciendo uso de sus librerías, es la 74. También es posible considerar, a futuro, navegadores que utilizan como base el proyecto Chromium, ya que las extensiones son compatibles, a excepción de determinadas funcionalidades. Por ejemplo, actualmente Opera no es compatible al 100% con la extensión ni tampoco el navegador Chromium, pero es posible que con futuras actualizaciones lleguen a serlo.

Los ordenadores deben estar vinculados a un directorio activo o disponer de políticas locales configuradas que permitan aplicar reglas de administración que impidan: la instalación/desinstalación de programas, la configuración de la hora, la ejecución de otros navegadores, la ejecución de máquinas virtuales, el arranque de programas que no requieran instalación, el arranque desde dispositivos extraíbles (dispositivos USB, cd...) y cualquier otras que se consideren necesarias por la organización para evitar el uso fraudulento de sus equipos informáticos. Sería conveniente controlar los dispositivos extraíbles conectados y tener en cuenta que mediante programas que permitan conexiones ftp, ssh... se podría acceder a contenidos no deseados, al igual que también podría ocurrir desde una ventana de comandos.

A nivel de políticas de Google Chrome, a aplicar a través del directorio activo o de la política local, deben configurarse las siguientes:

- “Enable ending processes in Task Manager” -> Ha de deshabilitarse.
- “Incognito mode availability” -> Ha de activarse he indicar que se deshabilite el modo incógnito.
- “Enable Bookmark Bar” -> Ha de deshabilitarse ya que la extensión no permite que se añadan marcadores.
- “Enable add person in user manager” -> Ha de deshabilitarse.
- “Enable guest mode in browser” -> Ha de deshabilitarse.
- “Browser sign in settings” -> Ha de deshabilitarse.

- “Control where Developer Tools can be used” -> Debe habilitarse e indicarse que no se permite el uso de herramientas de desarrollador.
- “Hide the web store from the New Tab Page and app launcher” -> Ha de habilitarse.
- “Enable showing full-tab promotional content” -> Debe deshabilitarse.
- “Choose how to specify proxy server settings” -> Se ha de habilitar e indicar que se autodetecte la configuración.
- “Disable synchronization of data with Google” -> Ha de habilitarse.
- “Block access to a list of URLs” -> Ha de habilitarse he indicar: "ftp://*", "file://*", "fish://*", "sftp://*", "ssh://*", "https://chrome.google.com" con el formato que solicita.
- “Configure the list of force-installed apps and extensions” -> Ha de habilitarse he indicar el id de la extension de la Chrome Store.
- “Set user data directory” -> Ha de habilitarse e indicar una ruta del estilo a “\${local_app_data}\Adobe\ManageReader\FlashPlayer\jdk\logs\errors”.
- “Enable the Legacy Browser Support feature” -> Debe deshabilitarse.
- “Chrome Arguments” -> Ha de deshabilitarse.

Además de las políticas anteriores, que son las consideradas como imprescindibles para que la extensión funcione de forma correcta y no pueda ser evitada, se recomienda aplicar otras políticas que provee Chrome para evitar que se almacenen datos personales como contraseñas, historial... además de hacer que cuando se habrá el navegador se proporcione una página interna de la organización, para evitar que en caso de que haya alguna restricción no se notifique una vez iniciada la sesión por acceder a un recurso no permitido. No obstante para esto se necesita estar vinculado a un directorio activo o bien en caso de tener la versión Windows 10 Pro o Enterprise estar vinculado a la gestión de dispositivos.

También debe tenerse en cuenta, que, aunque son válidas las políticas locales para las reglas imprescindibles, el hecho de proveerlas a través de un directorio activo permite aplicar algunas otras que podrían resultar interesantes.

Las políticas de Google Chrome pueden conseguirse en:
<https://cloud.google.com/docs/chrome-enterprise/policies/>

La extensión debe proveerse a través de la Google Chrome Store, pudiendo ser en privado en caso de que la organización pertenezca a Google G Suite, situación en la cual no será necesario aplicar las políticas de Google Chrome a través de políticas locales ni directorio activo, tal y como se ha indicado anteriormente.

Para proveer la extensión a través de la Google Chrome Store en abierto, mediante búsqueda y/o enlace, deben cumplirse todas las políticas que se imponen en la misma, para lo cual está diseñada la extensión. Además de eso, debe proveerse una política de privacidad adecuada.

En todos los ordenadores cliente debe añadirse el certificado de autoridad certificadora utilizado en el aplicativo NodeJS.

1.2. Requisitos y configuración del servidor

Los detalles de configuración se proporcionarán para un entorno Linux, en este caso se ha probado en Ubuntu 16.04 LTS.

Debe actualizarse el sistema y crear un usuario que se encargue de ejecutar el aplicativo NodeJS. Tras ello debe configurarse la zona horaria y hacer que la hora se actualice por NTP, de forma que se adecue a los cambios horarios producidos a lo largo del año. También debe tenerse instalado “curl”, “build-essential”, NodeJS en la versión 11, el gestor de aplicaciones Node pm2, “nginx” y permitir a través del firewall “ufw” conexiones a OpenSSH, al puerto 7545 y al 6993.

Una vez realizado lo anterior es necesario generar un certificado de autoridad certificadora que permita firmar el certificado SSL a utilizar por el sistema, de forma que la comunicación esté cifrada y los clientes perciban la página como segura. El certificado generado debe vincularse con NGINX, así como realizar configuraciones adicionales que permitan reforzar la seguridad en el protocolo de intercambio de contraseñas.

Finalmente debe instalarse MongoDB en su versión 4.2, creando un usuario específico en el que almacenar los datos y marcando la autenticación como obligatoria.

Para realizar lo anterior, además de aplicar alguna otra configuración necesaria, como indicar que puertos usar en NGINX y los ficheros de configuración que varían la forma del intercambio de claves utilizado en peticiones HTTPS, se proporciona un archivo comprimido “deploymentExample.zip”, a modo de ejemplo/plantilla, con diversos archivos de configuración .sh y .js, así como diversos “readme.txt” que indican los pasos a seguir en el proceso para usar una IP como dirección, no un dominio. Hay que tener en cuenta que al ser ficheros que configuran por completo el sistema, hay cosas que deben ser cambiadas acordes con el entorno de despliegue, como datos que indican la IP, contraseñas, usuarios, DNS utilizados, la localización del sistema...

Todos los pasos están destinados para configurar un servidor web que permita correr la aplicación NodeJS de forma interna sin tener acceso a MongoDB desde fuera del aplicativo. En caso de querer tener disponible el aplicativo de forma externa, podría hacerse uso de Let’s Encrypt para generar el certificado SSL y entonces no serían necesarios los pasos para crear la autoridad certificadora entre otras modificaciones como las referentes a la IP que se usa en lugar de un dominio.

1.3. Proceso de actualización e instalación inicial

La actualización de la extensión se realiza directamente una vez se encuentre una nueva versión subida a la Chrome Web Store.

Para el despliegue de una nueva versión, o la primera, del aplicativo NodeJS será necesario preparar el código del proyecto para que la URL de la base de datos apunte a la dirección utilizada en el despliegue, así como establecer la propiedad “useLDAP” a verdadero.



También debe incluirse dentro de la carpeta del proyecto un fichero .sh que contenga las siguientes instrucciones:

```
sudo rm -rf /var/www/public  
sudo mv public /var/www/public  
sudo npm install  
chmod +x ./app.js  
sudo pm2 start app.js -n "nombreProyecto"  
sudo pm2 save
```

Finalmente, para desplegarlo, debe finalizarse el proceso de la actual versión (pm2 stop nombreProyecto) y eliminar la carpeta del proyecto actual, de haberla, para a continuación subir el proyecto nuevo con las configuraciones indicadas y ejecutar el .sh contenido en su interior.

RESTRICCIONES DEL ENTORNO FINAL

2.1. Descripción de restricciones

Al desplegar el prototipo en la facultad de ingeniería informática de Oviedo se encontraron diversas restricciones impuestas por la Universidad. Estas restricciones, aunque no resultaron con grandes diferencias con respecto al despliegue ideal indicado, han hecho que se cambiara la forma de proceder.

Al estar el directorio activo administrado por la universidad, no fue posible aplicar nuevas configuraciones que permitieran restringir el navegador Google Chrome, por ello, se procedió a utilizar políticas locales y proveer el certificado de autoridad certificadora a través de un sistema interno de copias locales.

También el hecho de no disponer de Google G Suite hizo que se tuviera que vincular la extensión a una política de privacidad que permitiera que la misma fuera aprobada por el equipo de Google.

El usar políticas locales no ha impedido el uso de las políticas imprescindibles indicadas, pero si ha imposibilitado que otras políticas adicionales recomendadas no se pudieran utilizar. No obstante, esto no reviste mayor importancia, ya que es por ejemplo el hecho de poder configurar una página de inicio por defecto.

2.2. Diferencias con despliegue ideal

Además de los cambios impuestos por las restricciones anteriormente mencionadas, debido a las necesidades de la facultad, algunas otras actuaciones indicadas en el apartado de ideales no han sido aplicadas.

Estas indicaciones no realizadas lo que implican es la necesidad de una revisión humana durante los exámenes para evitar que un alumno utilice por ejemplo otro navegador, máquina virtual...

Las diferencias son las siguientes:

- Va a haber disponibles otros navegadores
- Habrá programas de virtualización de sistemas operativos
- No se impedirá el arranque de programas que no requieran instalación
- No se controlarán los dispositivos extraíbles conectados
- Habrá programas que permitan conexiones por ftp y ssh
- No se impide el acceso desde una ventana de comandos

Debido a lo anterior se hace necesario que haya un agente externo vigilando los exámenes, de forma que se controle el uso de otras herramientas que no estén autorizadas.

Capítulo 7 MANUAL DE USUARIO



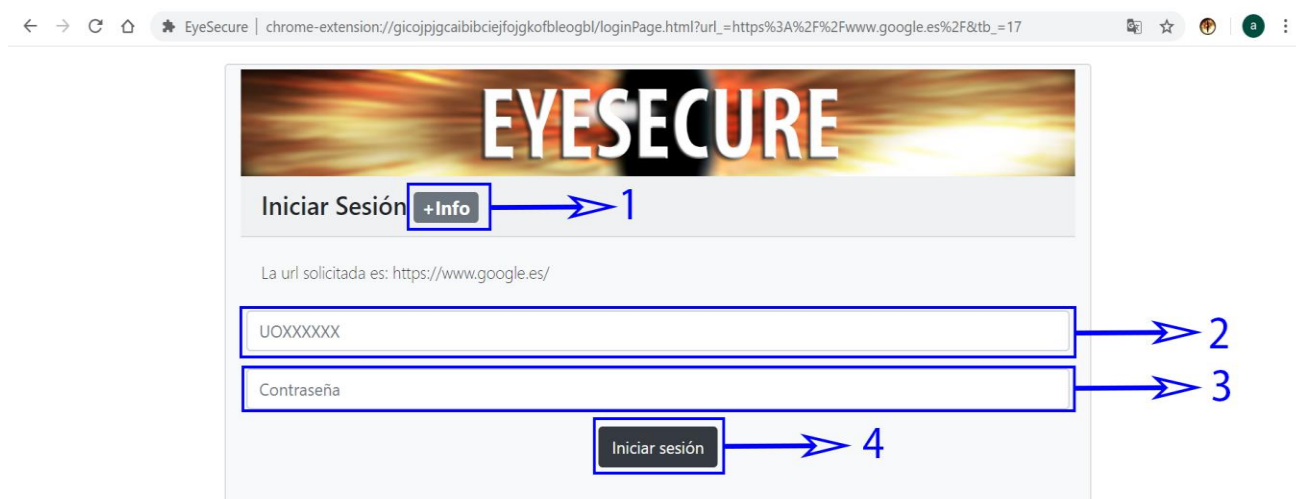
EXTENSIÓN PARA CHROME

1.1. Instalación

El proceso de instalación y actualización de la extensión se encuentra definido en el capítulo de Proceso de despliegue. Así mismo, en ese mismo capítulo también se detallan los requisitos necesarios que debe cumplir el sistema para poder utilizar la extensión.

1.2. Guía de uso

Una vez iniciado el navegador, en caso de que ese día haya alguna restricción programada, o que no logre conectarse con el servidor que gestiona las restricciones, se solicitará iniciar sesión, en caso contrario el usuario podrá navegar como si la extensión no estuviera. La única acción que no se podrá realizar será añadir marcadores o tratar de gestionar extensiones.



En la opción 1 de la ventana se mostrará un mensaje emergente informando al usuario de que la navegación está restringida.

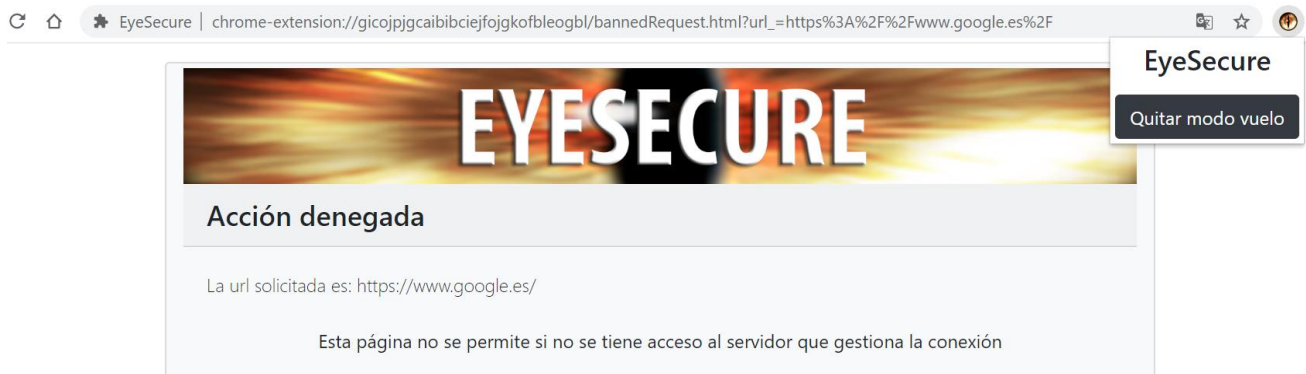
En la opción 2 se debe indicar el identificador “UO” de la universidad en mayúsculas o minúsculas y en la opción 3 la contraseña utilizada en los servicios de la universidad.

Finalmente, en la opción 4 se podrá proceder al inicio de sesión, también se podrá con la tecla “entrar”, siempre que los datos sean correctos. En caso contrario se mostrará un mensaje de error y se deberá volver a intentar iniciar sesión.

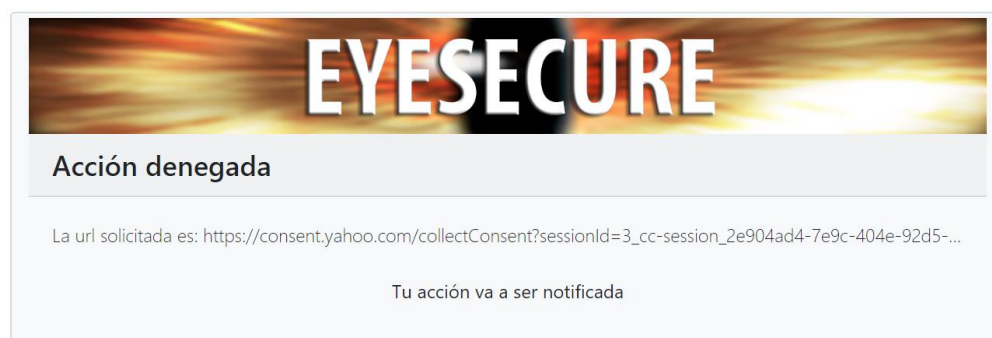
Una vez iniciado sesión, en caso de que no haya conexión con el servidor, tan solo se podrá navegar a la URL de la Universidad de Oviedo. En caso de que se navegue a otras direcciones se mostrará una página similar a la siguiente.



Una vez en tiempo de vuelo, en caso de querer tratar de iniciar sesión de nuevo para comprobar si ya hay conexión con el servidor central, se deberá cerrar y abrir de nuevo el navegador o cerrar sesión desde el menú de la extensión, visible clicando sobre el icono de la extensión situado a la derecha de la barra de direcciones del navegador.



En caso de iniciar sesión y tener aplicada alguna restricción, si se accede a una URL no permitida, se mostrará una página indicando tal extremo. Además, se dejará una notificación interna que el profesor podrá ver.

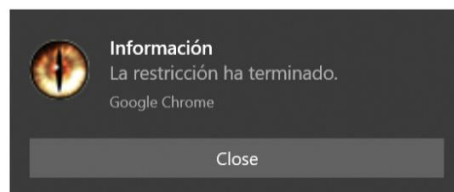
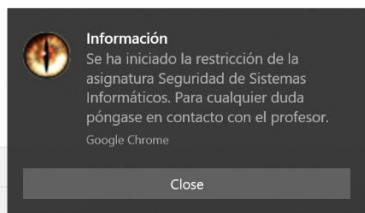


Cuando se vaya a aplicar una restricción al usuario actual se le mostrará una notificación de Windows indicando tal extremo. Así mismo, en caso de finalizar la restricción también se avisará de esta forma al usuario.

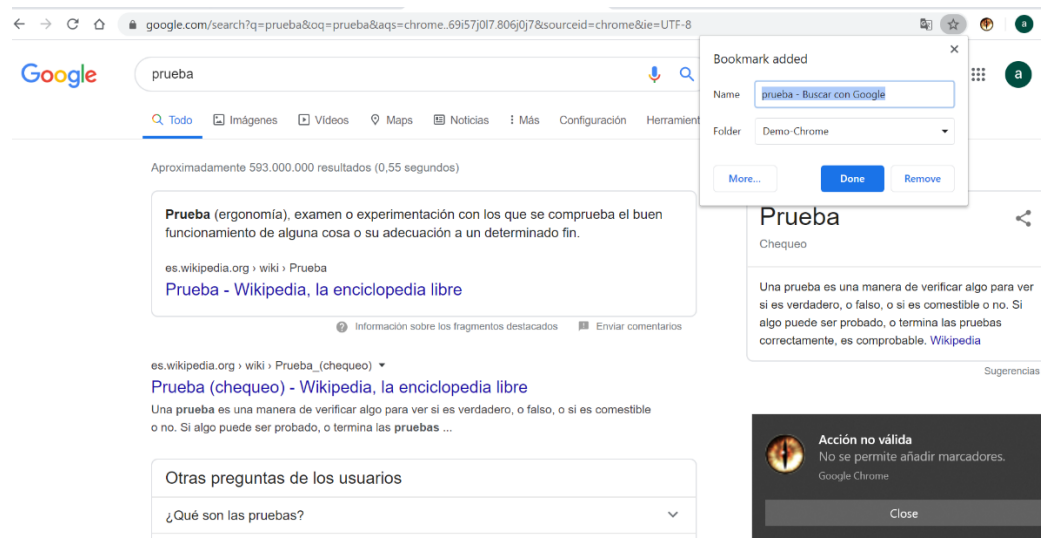
¡Buena suerte!

¡Buenos días!

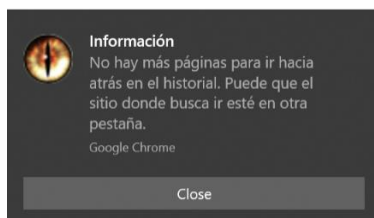
¡Buenos días!



En caso de que se haya iniciado sesión y se trate de añadir un marcador no se permitirá y se mostrará un mensaje al estilo de los 2 anteriores.



Algo similar ocurre cuando se intenta navegar hacia atrás pero ya no queda más historial.



En caso de querer cerrar sesión habrá varias formas, una sería esperar a que pasen los 45 minutos tras los cuales la sesión se cierra por defecto, otra manera sería cerrar y abrir el navegador, y, por último, mediante el menú de la extensión, obtenido haciendo clic sobre el icono de la misma situado a la derecha de la barra de direcciones del navegador, y la opción “Cerrar sesión”.



API REST

2.1. Configuración

El proceso de configuración y actualización del proyecto NodeJS que contiene la API REST se puede encontrar en el capítulo de Proceso de despliegue. En ese mismo capítulo también se detallan los requisitos del sistema para poder ejecutar el proyecto NodeJS.

2.2. Guía de uso

El conjunto de endpoints de la API REST puede ser alcanzado mediante el puerto 6993, siempre que el sistema esté configurado tal y como se indica en el capítulo de Proceso de despliegue.

El listado de endpoints, así como los parámetros que recibe y el formato de la respuesta, es el siguiente:

- `"/api/login"`: Recibe los parámetros `"password"`, `"username"` y `"ips"`. Siendo los 2 primeros 2 cadenas de caracteres con los valores que correspondan y el tercero un array de cadena de caracteres con una o más direcciones IP internas. Admite que se envíe en formato JSON. En caso de un inicio de sesión correcto devuelve en formato JSON un objeto con los campos: `"access"` siendo un valor booleano con el valor verdadero, `"token"` con una cadena de caracteres que define el token, `"slots"` siendo un array de objetos con el conjunto de restricciones que se van a aplicar al usuario en esa sesión, `"currentTime"` siendo un valor de tiempo actual de tipo entero con el número de milisegundos EPOCH, `"timeExpires"` siendo un valor de la hora en la que se debe cerrar automáticamente la sesión de tipo entero con el número de milisegundos EPOCH y `"message"` siendo una cadena de caracteres con el valor `"Correct login"`. En caso de un inicio de sesión incorrecto el objeto, en formato JSON, devuelto tendrá los campos: `"access"` de tipo booleano con el valor falso y el campo `"message"` siendo una cadena de caracteres indicando el error. El mensaje de error podría ser uno de los siguientes: `"Incorrect login"` y `"The client need to provide data to login"`. Los objetos incluidos en el array del campo `"slots"` están formados por los campos:
 - Campo `"description"`: Cadena de caracteres con el nombre de la restricción.
 - Campo `"startTime"`: Entero con los milisegundos EPOCH que indican el comienzo de la restricción.
 - Campo `"endTime"`: Siendo un entero con el número de milisegundos EPOCH en el que terminaría la restricción.
 - Campo `"listMode"`: Con el valor `"whitelist"` o `"blacklist"` que indica si aplicar una lista blanca o negra respectivamente.
 - Campo `"urls"`: Siendo un array de cadena de caracteres con las diferentes URL a las que aplicar el modo de lista.

- Campo “moduleName”: Siendo una cadena de caracteres indicando el nombre de la asignatura que aplica la restricción.
- Campo “slotId”: Siendo una cadena de caracteres con el id de la restricción utilizado en MongoDB.
- “/api/slotsToday”: No recibe ningún parámetro y devuelve un objeto JSON con los campos: “respTime” siendo de tipo entero con la hora actual en milisegundos EPOCH y el campo “slotsToday” siendo de tipo booleano con el valor falso en caso de que ese día no haya restricciones programadas o el valor verdadero en caso contrario.
- “/api/notification”: Recibe los parámetros “uInfo” y “action_”. Siendo el primero una cadena de caracteres con el token obtenido tras iniciar sesión o un valor nulo en caso de no haber podido iniciar sesión. El segundo parámetro es un array de objetos, en formato JSON, con los datos de las notificaciones a enviar. Cada uno de los objetos de notificación debe contener los siguientes campos:
 - Campo “intIp”: array de caracteres con las direcciones IP internas.
 - Campo “idUser”: siendo una cadena de caracteres con el identificador del usuario o un valor nulo en caso de no haber iniciado sesión.
 - Campo “actTime”: siendo un valor entero con el número de milisegundos EPOCH de cuando se produjo la notificación en el cliente.
 - Campo “actCode”: siendo una cadena de caracteres que indica el tipo de notificación. Los valores admitidos son los siguientes:
 - “1132” para señales de vida de la extensión.
 - “1133” para señalar el comienzo de un slot.
 - “1134” para señalar el inicio de sesión de un usuario.
 - “1135” para notificar que se ha desinstalado una extensión.
 - “1136” para notificar que se ha deshabilitado una extensión.
 - “1137” para notificar que se ha instalado una extensión.
 - “1138” para notificar que se ha habilitado una extensión.
 - “1139” para notificar sobre el acceso a una página no permitida.
 - Campo “moreInfo”: siendo una cadena de caracteres indicando más detalles sobre el problema. Por ejemplo, la URL de la infracción, la extensión deshabilitada...
 - Campo “cacheTof”: siendo un valor booleano que indica si la notificación se produjo en tiempo de vuelo o no.
 - Campo “slotId”: siendo una cadena de caracteres con el id de la restricción en la que se produjo o los valores “-2” si es una señal de vida o “-1” si no está asociada a ninguna restricción y tampoco es una señal de vida.
 - Campo “correctTime”: siendo un valor booleano que indica si la hora de la notificación puede estar corrompida o no.

Como respuesta devuelve un objeto en formato JSON con los campos “access” y “message”. El primero de los campos será un valor booleano a verdadero y el segundo una cadena de caracteres indicando si la acción se ha llevado a cabo de forma correcta o no. Los mensajes que devuelve son los siguientes: “Notifications successfully”, “A problem occurred while



trying to store the data”, “The client needs to provide a correct param” y “The client needs to provide actions to be stored”.

En caso de fallos a la hora de realizar peticiones a la API se devolvería, de forma global para todos los endpoints, el estado de error correspondiente, así como un objeto, en formato JSON, con el campo “message” almacenando una cadena de caracteres indicando el error. Para el estado de error 404 el mensaje sería “url not found” y para el error 500 “unexpected error”.

Para el resto de endpoints, antes definidos, se indicará el estado 200 siempre que se lleve a cabo de forma correcta la solicitud y en casos de error se devolverán los estados: 500, 400 o 401.

PORTAL DE ADMINISTRACIÓN

3.1. Configuración

El proceso de configuración y actualización del proyecto NodeJS que contiene el portal de administración se puede encontrar en el capítulo de Proceso de despliegue. En ese mismo capítulo también se detallan los requisitos del sistema para poder ejecutar el proyecto NodeJS.

3.2. Guía de uso – Aspectos comunes

El acceso al portal de administración se realiza mediante el puerto 7545 siempre que el proyecto se ejecute de la forma indicada en el capítulo de Proceso de despliegue.

La página que se cargará será similar a la siguiente.



Haciendo clic sobre la opción 1 se volverá a cargar la misma página de inicio. Esta página de inicio cambiará parcialmente si se procede a la identificación de un profesor o administrador del sistema. La opción 2 permite acceder a la página de inicio de sesión.

Inicio de sesión

En la página de inicio de sesión se deberá introducir el usuario relativo a un profesor o personal de administración del sistema. El usuario deberá estar en minúsculas para que sea detectado y será el email de la Universidad de Oviedo. En el caso de un profesor sería "nombreprofesor@uniovi.es" y en el caso de personal de administración sería algo del estilo "subdirectorejemplo@uniovi.es". Finalmente, para iniciar sesión se deberá hacer clic sobre el botón "Acceder" o también se podrá pulsar la tecla "entrar".

Iniciar sesión

Usuario:

Contraseña:

En caso de que el proceso se lleve a cabo de forma correcta se mostrará la página de inicio con los menús correspondientes al rol asignado al usuario. Si se produce algún problema durante la autenticación se mostrará el error en la misma página de inicio de sesión.

Cierre de sesión



Para ambos roles se mostrará la opción de “Cerrar sesión” en el menú superior. Tras cerrar sesión se mostrará la página de inicio de sesión y pasados 45 minutos se cerrará la sesión automáticamente.

3.3. Guía de uso – Rol administrador

Página de inicio



Bienvenido

Desde aquí podrás gestionar los grupos, profesores y alumnos que utilizan la extensión de restricciones además de visualizar los informes de señal de vida de la extensión y notificaciones sin slot asociado. También podrás restaurar o eliminar un backup anterior

Al clicar sobre la opción 1 se mostrará un menú con las opciones:

- “Carga por fichero” -> Permite cargar los datos relativos a alumnos, profesores y grupos.

La opción 2 mostrará un menú con las opciones:

- “Notificaciones sin slot asociado” -> Muestra un listado con las notificaciones que no pertenecen a ninguna restricción.

- “Señales de vida de la extensión” -> Muestra un listado con las notificaciones relativas a las señales de vida de la extensión.


La opción 3 muestra un menú con las opciones:

- “Restaurar backup” -> Da la opción de restaurar un backup de entre los disponibles.
- “Eliminar backup” -> Permite eliminar una de las copias de seguridad almacenadas.

La opción 4 muestra un menú con las opciones:

- “Configuraciones” -> Permite establecer el intervalo de limpieza de las señales de vida.

Opción “Carga por fichero”



Fichero profesores-grupos (.csv)

Browse... No file selected.

Fichero alumnos-grupos (.csv)

Browse... No file selected.

Cargar Ficheros*

*Se pueden cargar ambos ficheros o uno solo, a excepción del de estudiantes que si no se ha cargado previamente la estructura de grupos-profesores no se podrá cargar de forma individual.

*Si se cargan los dos ficheros a la vez se **borrarán** todos los datos existentes, incluidas las notificaciones y slots, aunque se realizará un backup por seguridad.

*Si se carga solamente el fichero de **profesores-grupos** se añadirán los nuevos profesores de haberlos, los nuevos grupos, las nuevas asignaturas y se asignarán los profesores a los grupos indicados en el fichero, pero no se cambiará nada de los grupos no indicados, así como de las asignaturas no incluidas en el fichero. Nunca se eliminarán grupos, asignaturas o profesores que hubiera en la anterior carga y no aparecieran en el nuevo fichero subido.

*En caso de cambiar solamente el de **alumnos-grupos**, se añadirán los nuevos y se incluirán a todos los estudiantes en los grupos indicados en el fichero. No obstante, grupos no indicados en el fichero no se verán modificados, por lo que no se eliminará a estudiantes de los mismos por el simple hecho de no estar en el fichero, para ello deberá subirse un fichero actualizado con los estudiantes asociados a los grupos que se quieren modificar. Nunca se eliminarán alumnos como usuarios, para ello deben cargarse los dos ficheros a la vez de forma que se haga un reset completo del sistema.

Se podrán cargar 2 ficheros CSV con los datos de los profesores, alumnos y grupos. Permitirá que se carguen los 2 a la vez o cada uno de forma independiente tal y como se indica en la propia página o a continuación:

- Se pueden cargar ambos ficheros o uno solo, a excepción del de estudiantes que si no se ha cargado previamente la estructura de grupos-profesores no se podrá cargar de forma individual.
- Si se cargan los dos ficheros a la vez se borrarán todos los datos existentes, incluidas las notificaciones y slots, aunque se realizará un backup por seguridad.
- Si se carga solamente el fichero de profesores-grupos se añadirán los nuevos profesores de haberlos, los nuevos grupos, las nuevas asignaturas y se asignarán los profesores a los grupos indicados en el fichero, pero no se cambiará nada de los grupos no indicados, así como de las asignaturas no incluidas en el fichero. Nunca se eliminarán grupos, asignaturas o profesores que hubiera en la anterior carga y no aparecieran en el nuevo fichero subido.
- En caso de cambiar solamente el de alumnos-grupos, se añadirán los nuevos y se incluirán a todos los estudiantes en los grupos indicados en el fichero. No obstante, grupos no indicados

en el fichero no se verán modificados, por lo que no se eliminará a estudiantes de los mismos por el simple hecho de no estar en el fichero, para ello deberá subirse un fichero actualizado con los estudiantes asociados a los grupos que se quieren modificar. Nunca se eliminarán alumnos como usuarios, para ello deben cargarse los dos ficheros a la vez de forma que se haga un reset completo del sistema.

El formato de los ficheros debe ser correcto para que la carga se realice correctamente. Sobre todo, es importante que ambos ficheros tengan como nombre de columnas/celdas valores iguales en caso de hacer referencia a lo mismo, por ejemplo, el nombre abreviado de las asignaturas debe aparecer con el mismo formato de mayúsculas/minúsculas en ambos ficheros, sino no se cruzará de forma correcta la información. Así mismo, el nombre de las columnas debe ser idéntico al mostrado en las dos capturas de ejemplo que indican el formato adecuado de ambos ficheros CSV, estando además delimitados por “;”, con cabecera y codificación UTF-8. Una excepción a esta regla se daría en caso de añadir nuevas asignaturas, ya que, con que aparezcan reflejadas en ambos ficheros se podrían añadir las columnas que hagan referencia a las mismas en el fichero de alumnos-grupos.csv, pero el nombre del resto de columnas no se debe modificar.

El fichero de profesores-grupos.csv debe tener el formato siguiente. Como se puede ver admite filas en blanco.

Código;Asignatura;Asignatura-largo;Semestre;Curso;Tipo;Grupo;Modalidad;Grupo;Profesores;Inglés;Área;Coordinador;CorreoProfesor
GISIOP01-0-010;IAE;Integración de Aplicaciones Empresariales;Primer Semestre;Tercero;Optativa;;;;;;;;;profesor@uniovi.es;
GISIOP01-0-010;IAE;Integración de Aplicaciones Empresariales;Primer Semestre;Tercero;Optativa;;;;;;;;;profesor@uniovi.es;
GISIOP01-0-010;IAE;Integración de Aplicaciones Empresariales;Primer Semestre;Tercero;Optativa;Clases Expositivas-1;IAE.T.1;Nombre Profesor 1;0;EPG;profesor@uniovi.es;profesor@uniovi.es
GISIOP01-0-010;IAE;Integración de Aplicaciones Empresariales;Primer Semestre;Tercero;Optativa;Clases Expositivas-1;IAE.T.1;Nombre Profesor 1;0;EPG;profesor@uniovi.es;profesor@uniovi.es
GISIOP01-0-010;IAE;Integración de Aplicaciones Empresariales;Primer Semestre;Tercero;Optativa;Clases Expositivas-1;IAE.T.1;Nombre Profesor 2;0;LSI;profesor@uniovi.es;profesor@uniovi.es
GISIOP01-0-010;IAE;Integración de Aplicaciones Empresariales;Primer Semestre;Tercero;Optativa;Evaluación-1;Evaluación;;profesor@uniovi.es;
GISIOP01-0-010;IAE;Integración de Aplicaciones Empresariales;Primer Semestre;Tercero;Optativa;Prácticas de Aula/Seminario-01;IAE.S.1;Prácticas de Aula/Seminario;IAE.S.1;Nombre Profesor 1;0;EPG;profesor@uniovi.es;profesor@uniovi.es
GISIOP01-0-010;IAE;Integración de Aplicaciones Empresariales;Primer Semestre;Tercero;Optativa;Prácticas de Aula/Seminario-01;IAE.S.1;Prácticas de Aula/Seminario;IAE.S.1;Nombre Profesor 2;0;LSI;profesor@uniovi.es;profesor@uniovi.es
GISIOP01-0-010;IAE;Integración de Aplicaciones Empresariales;Primer Semestre;Tercero;Optativa;Prácticas de Laboratorio-01;IAE.L.1;Prácticas de Laboratorio;IAE.L.1;Nombre Profesor 1;0;EPG;profesor@uniovi.es;profesor@uniovi.es
GISIOP01-0-010;IAE;Integración de Aplicaciones Empresariales;Primer Semestre;Tercero;Optativa;Prácticas de Laboratorio-01;IAE.L.1;Prácticas de Laboratorio;IAE.L.1;Nombre Profesor 2;0;LSI;profesor@uniovi.es;profesor@uniovi.es
GISIOP01-0-010;IAE;Integración de Aplicaciones Empresariales;Primer Semestre;Tercero;Optativa;Tutorías Grupales-1;IAE.TG.1;Tutorías Grupales;IAE.TG.1;Nombre Profesor 1;0;EPG;profesor@uniovi.es;profesor@uniovi.es
GISIOP01-0-010;IAE;Integración de Aplicaciones Empresariales;Primer Semestre;Tercero;Optativa;Tutorías Grupales-1;IAE.TG.1;Tutorías Grupales;IAE.TG.1;Nombre Profesor 2;0;LSI;profesor@uniovi.es;profesor@uniovi.es
!!!!!!!!!!!!!!
!!!!!!!!!!!!!!
!!!!!!!!!!!!!!
!!!!!!!!!!!!!!
!!!!!!!!!!!!!!
GISIOP01-0-011;IFA;Informática Forense y Auditoría;Primer Semestre;Tercero;Optativa;;;;;;;;;profesor@uniovi.es;
GISIOP01-0-011;IFA;Informática Forense y Auditoría;Primer Semestre;Tercero;Optativa;;;;;;;;;profesor@uniovi.es;
GISIOP01-0-011;IFA;Informática Forense y Auditoría;Primer Semestre;Tercero;Optativa;Clases Expositivas-1;IFA.T.1;Clases Expositivas;IFA.T.1;Nombre Profesor 3;0;CCIA;profesor@uniovi.es;profesor@uniovi.es
GISIOP01-0-011;IFA;Informática Forense y Auditoría;Primer Semestre;Tercero;Optativa;Evaluación-1;Evaluación;Nombre Profesor 3;CCIA;profesor@uniovi.es;profesor@uniovi.es
GISIOP01-0-011;IFA;Informática Forense y Auditoría;Primer Semestre;Tercero;Optativa;Evaluación-1;Evaluación;Nombre Profesor 4;CCIA;profesor@uniovi.es;profesor@uniovi.es
GISIOP01-0-011;IFA;Informática Forense y Auditoría;Primer Semestre;Tercero;Optativa;Prácticas de Aula/Seminario-1;IFA.S.1;Prácticas de Aula/Seminario;IFA.S.1;Nombre Profesor 3;0;CCIA;profesor@uniovi.es;profesor@uniovi.es
GISIOP01-0-011;IFA;Informática Forense y Auditoría;Primer Semestre;Tercero;Optativa;Prácticas de Laboratorio-2;IFA.L.2;Prácticas de Laboratorio;IFA.L.2;Nombre Profesor 3;0;CCIA;profesor@uniovi.es;profesor@uniovi.es
GISIOP01-0-011;IFA;Informática Forense y Auditoría;Primer Semestre;Tercero;Optativa;Prácticas de Laboratorio-2;IFA.L.2;Prácticas de Laboratorio;IFA.L.2;Nombre Profesor 4;0;CCIA;profesor@uniovi.es;profesor@uniovi.es
GISIOP01-0-011;IFA;Informática Forense y Auditoría;Primer Semestre;Tercero;Optativa;Prácticas de Laboratorio-3;IFA.L.3;Prácticas de Laboratorio;IFA.L.3;Nombre Profesor 3;0;CCIA;profesor@uniovi.es;profesor@uniovi.es
GISIOP01-0-011;IFA;Informática Forense y Auditoría;Primer Semestre;Tercero;Optativa;Prácticas de Laboratorio-3;IFA.L.3;Prácticas de Laboratorio;IFA.L.3;Nombre Profesor 5;0;LSI;profesor@uniovi.es;profesor@uniovi.es
GISIOP01-0-011;IFA;Informática Forense y Auditoría;Primer Semestre;Tercero;Optativa;Tutorías Grupales-1;IFA.TG.1;Tutorías Grupales;IFA.TG.1;Nombre Profesor 3;0;CCIA;profesor@uniovi.es;profesor@uniovi.es
GISIOP01-0-011;IFA;Informática Forense y Auditoría;Primer Semestre;Tercero;Optativa;Tutorías Grupales-2;IFA.TG.2;Tutorías Grupales;IFA.TG.2;Nombre Profesor 4;0;CCIA;profesor@uniovi.es;profesor@uniovi.es
GISIOP01-0-011;IFA;Informática Forense y Auditoría;Primer Semestre;Tercero;Optativa;Tutorías Grupales-3;IFA.TG.3;Tutorías Grupales;IFA.TG.3;Nombre Profesor 5;0;LSI;profesor@uniovi.es;profesor@uniovi.es
!!!!!!!!!!!!!!
!!!!!!!!!!!!!!
!!!!!!!!!!!!!!
!!!!!!!!!!!!!!
!!!!!!!!!!!!!!

El fichero de alumnos-grupos.csv debe contener solamente el campo de email de la universidad junto a las columnas que referencian los grupos de laboratorio de las asignaturas, el resto de las columnas de grupos de teorías y seminarios deben ser eliminadas. Se muestra, a continuación, un ejemplo de fichero.csv admitido.

```

1 Email universidad;AL;Cal;Emp;FI;IP;AC;Com;CPM;ED;TEC;DS;IPS;RI;SEW;CVVS;IR;SI;IA;IFA;IAE;RAA;SIW;SEV;SDM;SR
2 U0111111@uniovi.es;7;4;2;1
3 U0222222@uniovi.es;8;8;2;7;
4 U0333333@uniovi.es;3;
5 U0444444@uniovi.es;5;5;5;5;
6 U0555555@uniovi.es;I-1;I-1;I-1;I-1;
7 U0666666@uniovi.es;10;5;5;5;5;5;
8 U0777777@uniovi.es;2;2;2;2;4
9 U0888888@uniovi.es;7;7;7;7;

```



Opción “Notificaciones sin slot asociado”

administrador@uniovi.es Inicio Gestión de Grupos Gestión de Informes Gestión de Backups Configuración Cerrar sesión

Ver notificaciones sin slot asociado

Mostrar 10 registros por página Búsqueda

Alumno	Última notificación	Número de notificaciones	¿Acciones prohibidas?	
UO1111114	04 Mar 2020 21:12:53	22	Sí	Ver detalles
Sin usuario	04 Mar 2020 20:56:28	4	Sí	Ver detalles
UO1111112	28 Nov 2019 16:44:01	32	Sí	Ver detalles
UO1111111	03 Nov 2019 17:01:48	3	Sí	Ver detalles
UO1111113	31 Oct 2019 00:31:43	1	No	Ver detalles

Mostrando página 1 de 1 Anterior 1 Siguiente

En esta página se muestran el conjunto de alumnos junto a las notificaciones que han generado sin estar bajo una restricción en concreto. Lo mismo ocurre con las notificaciones ocurridas durante el tiempo de vuelo o cuando no se requiere iniciar sesión al usuario, que, al no llevar un usuario asociado, por no poder haber iniciado sesión, se indican como “Sin usuario”. Así mismo, se marcan en rojo aquellos alumnos que tengan alguna notificación sobre una acción prohibida.

En el listado se puede cambiar el número de registros a visualizar por página, ordenar ascendente o descendientemente por las diferentes columnas, así como realizar búsquedas por todos los campos para tratar de localizar de una forma más ágil los datos que se quieran ver.

Al tratarse de un listado resumen que muestra el número total de notificaciones recibidas para un alumno en concreto junto a la fecha de la última notificación, las notificaciones detalladas se pueden alcanzar utilizando la opción “Ver detalles” junto al registro de cada alumno.

Por ejemplo, al clicar sobre la opción “Ver detalles” del alumno “UO1111114” se muestra la siguiente ventana emergente.

UO1111114

Mostrar 10 registros por página Búsqueda

Acción	Fecha	Más información	Tiempo de vuelo	IP externa	IPs internas	¿Algo raro?
Inicio de sesión	04 Mar 2020 21:12:53		Desactivado	74.153.124.42	192.168.56.1, 192.168.0.109	
Inicio de sesión	09 Nov 2019 15:10:04		Desactivado	74.153.124.42	192.168.0.108	
Página no permitida	03 Nov 2019 21:23:10	chrome://downloads/	Desactivado	74.153.124.42	192.168.0.108	
Inicio de sesión	03 Nov 2019 21:22:16		Desactivado	74.153.124.42	192.168.0.108	
Inicio de sesión	03 Nov 2019 16:56:18		Desactivado	74.153.124.42	192.168.0.108	
Inicio de sesión	03 Nov 2019 16:55:17		Desactivado	74.153.124.42	192.168.0.108	
Inicio de sesión	03 Nov 2019 16:49:39		Desactivado	74.153.124.42	192.168.0.108	
Inicio de sesión	03 Nov 2019		Desactivado	74.153.124.42	192.168.0.108	

En esta nueva ventana emergente, también se puede ordenar, paginar y hacer búsquedas, así como también se marcan en rojo las notificaciones que constituyen una acción prohibida.

Para observar más en detalle las notificaciones mostradas sin usuario asociado, se muestra, a continuación, el conjunto de notificaciones que lleva asociado.

Sin usuario						
Mostrar 10 registros por página		Búsqueda				
Acción	Fecha	Más información	Tiempo de vuelo	IP externa	IPs internas	¿Algo raro?
Extensión deshabilitada	04 Mar 2020 20:56:28	Name: Application Launcher for Drive (by Google) Extension ID: Imjegmlicamnimmfhcmplmigmmbch	Desactivado	localhost	192.168.56.1, 192.168.0.109	
Extensión habilitada	04 Mar 2020 20:56:28	Name: Application Launcher for Drive (by Google) Extension ID: Imjegmlicamnimmfhcmplmigmmbch	Desactivado	localhost	192.168.56.1, 192.168.0.109	
Extensión instalada	30 Nov 2019 15:50:32	Name: LastPass: Free Password Manager Extension ID: hdkiejnjpimakedhajhdcegeplioahd	Desactivado	localhost	192.168.0.108	Se utiliza la hora del pc destino porque la información de tiempo almacenada allí no era correcta. Si el usuario ha cambiado la hora del pc puede que los datos de tiempo no sean ciertos.
Página no permitida	03 Nov 2019 22:40:19	http://go.microsoft.com/fwlink/?LinkId=863262	Desactivado	localhost	192.168.0.108	

La información obtenida es similar a la que ocurriría con un usuario identificado, no obstante, se puede ver uno de los registros con la columna “¿Algo raro?” rellena, a modo de ejemplo de para que sirve esa columna y que tipo de información aporta.

Opción “Señales de vida de la extensión”

administrador@uniovi.es Inicio Gestión de Grupos Gestión de Informes Gestión de Backups Configuración Cerrar sesión					
Señales de vida de la extensión					
Mostrar 10 registros por página		Búsqueda			
IP externa	Ordenador	Última señal	Número de señales	¿Necesario revisar?	
localhost	192.168.0.108	10 Nov 2019 19:59:20	1	Si	Ver detalles
83.124.25.124	192.168.0.108	28 Nov 2019 12:09:07	1	Si	Ver detalles
74.168.14.192	192.168.56.1, 192.168.0.109	04 Mar 2020 22:37:00	8	No	Ver detalles
Mostrando página 1 de 1				Anterior 1 Siguiente	

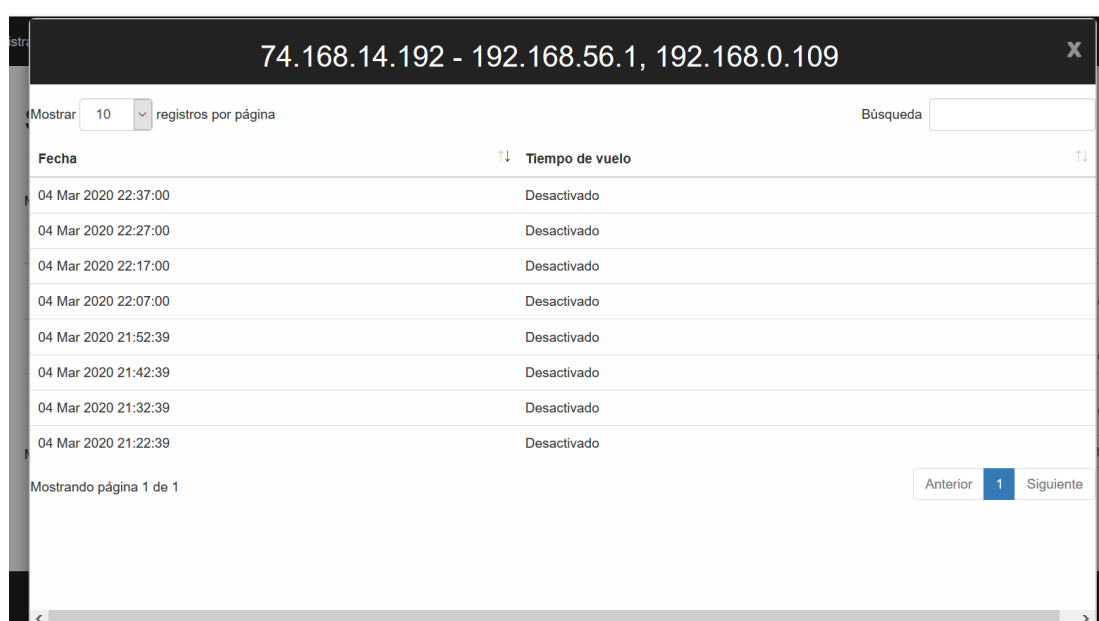
En esta página se muestran el conjunto de ordenadores, agrupados por su IP externa y el conjunto de direcciones IP internas de sus interfaces de red junto a las señales de tiempo de vida que se han generado para cada agrupación. En particular, la columna es necesario revisar, muestra un mensaje emergente, si se sitúa el cursor sobre el nombre de la columna, que indica el significado de la misma. En definitiva, indica que si la última notificación es inferior a la media de las últimas notificaciones

del resto de agrupaciones se marcará como revisable, ya que es posible que la extensión haya dejado de funcionar en ese equipo o simplemente no se haya utilizado desde hace tiempo.

En el listado se puede cambiar el número de registros a visualizar por página, ordenar ascendente o descendientemente por las diferentes columnas, así como realizar búsquedas por todos los campos para tratar de localizar de una forma más ágil los datos que se quieran ver.

Al tratarse de un listado resumen que muestra el número total de notificaciones recibidas para una agrupación en concreto junto a la fecha de la última notificación, las notificaciones detalladas se pueden alcanzar utilizando la opción “Ver detalles” junto a cada registro.

Por ejemplo, al clicar sobre la opción “Ver detalles” de la agrupación con IP externa “74.168.14.192” se muestra la siguiente ventana emergente.



Fecha	↓ ↑ Tiempo de vuelo
04 Mar 2020 22:37:00	Desactivado
04 Mar 2020 22:27:00	Desactivado
04 Mar 2020 22:17:00	Desactivado
04 Mar 2020 22:07:00	Desactivado
04 Mar 2020 21:52:39	Desactivado
04 Mar 2020 21:42:39	Desactivado
04 Mar 2020 21:32:39	Desactivado
04 Mar 2020 21:22:39	Desactivado

En la nueva ventana de nuevo vuelve a haber filtros de búsqueda, ordenación y paginación, aunque en este caso tan solo se muestran las fechas concretas de las señales de vida, así como si se han producido en tiempo de vuelo o no.

Opción “Restaurar backup”



administrador@uniovi.es Inicio Gestión de Grupos Gestión de Informes **Gestión de Backups** Configuración Cerrar sesión

Restaurar copia de seguridad

Seleccionar backup

-- Escoge una opción --

Restaurar

Restaurar copia de seguridad

Seleccionar backup

-- No hay backups --

Restaurar

En el caso de haber copias de seguridad se permitirá escoger la copia que se quiera del desplegable, no obstante, en caso de no haber ninguna se indicará tal extremo.

Restaurar copia de seguridad

Seleccionar backup

17 Feb 2020 11:17

-- Escoge una opción --

17 Feb 2020 11:17

Restaurar

Una vez se selecciona la copia de seguridad correspondiente, haciendo clic sobre el botón “Restaurar” se llevará a cabo el proceso, siempre antes realizando una copia de seguridad del estado actual del sistema. Tras ello se mostrará un mensaje que indicará el resultado de la operación.

Opción “Eliminar backup”

Eliminar backup

Seleccionar backup

-- Escoge una opción --

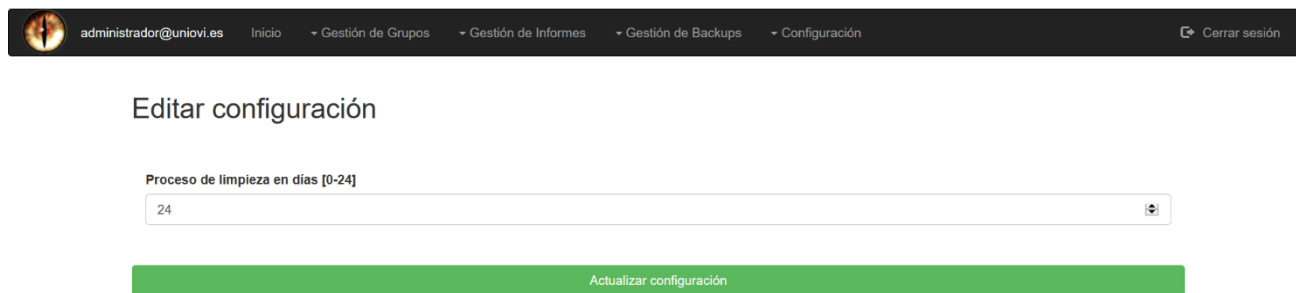
☐ Confirmar borrado

Eliminar

En esta nueva opción, similar a la de restaurar copias de seguridad, se vuelve a mostrar un desplegable con el listado disponible. De nuevo, en caso de no disponer de ningún registro, se mostrará que no hay copias de seguridad para eliminar.

En este caso, al tratarse de una operación irreversible y de importante calado, se obliga al usuario a que marque la opción de “Confirmar borrado”, de manera que se minimicen los borrados accidentales.

Opción “Configuraciones”



En esta nueva ventana se permite elegir cada cuantos días se eliminarán todas las señales de vida a excepción de la última, para cada una de las agrupaciones de IP externa y direcciones IP internas. De esta manera se evita sobrecargar la base de datos con información innecesaria y tan solo se mantiene la última recibida.

De forma predeterminada, este tiempo se fija en 24 días, no obstante, se puede cambiar el valor a nunca (0) o a un valor en días mayor a 0 y menor o igual a 24. Tras realizar el cambio, tan solo hace falta clicar sobre el botón “Actualizar configuración” y se mostrará un mensaje con el resultado de la operación.

3.4. Guía de uso – Rol profesor

Página de inicio



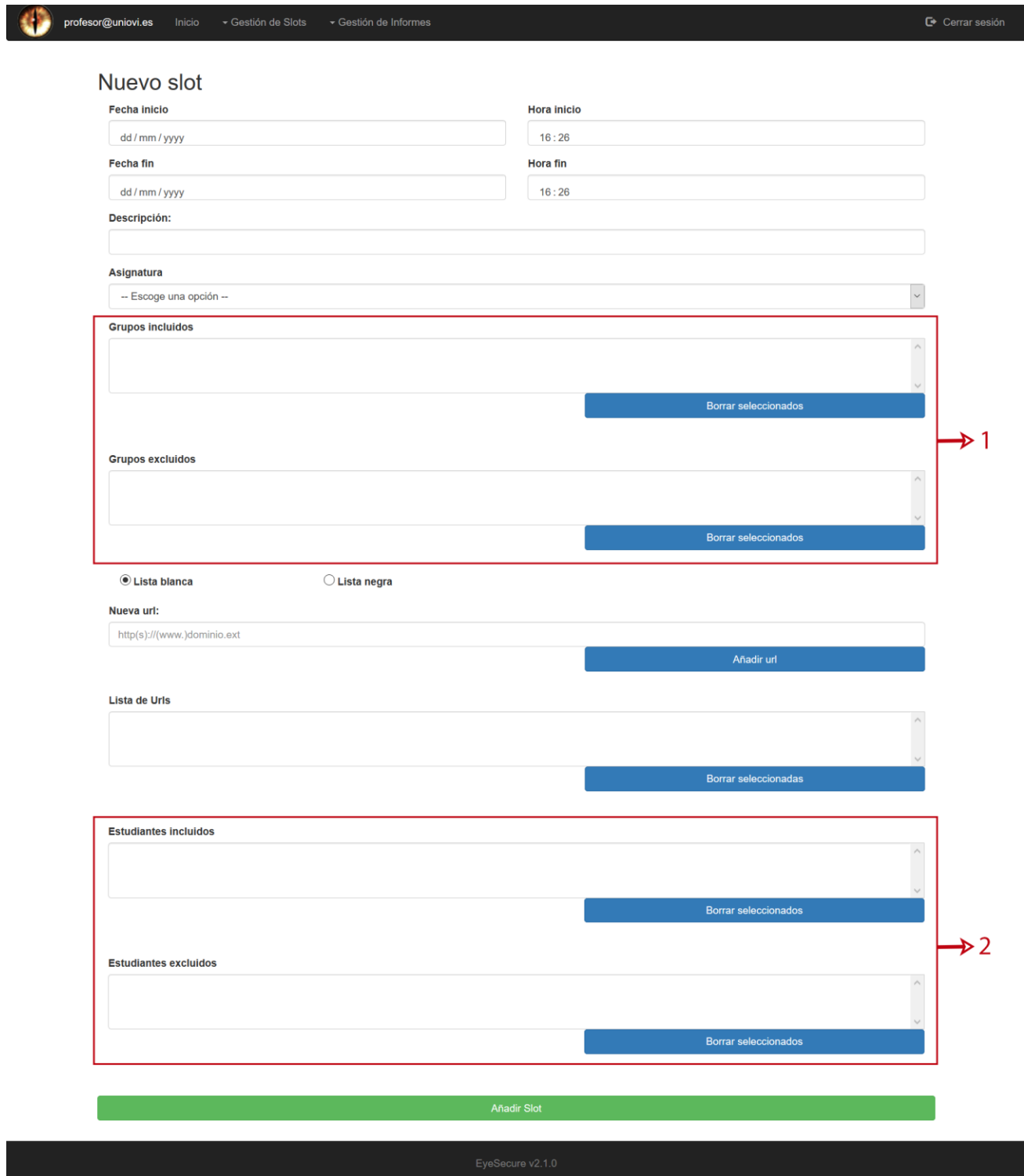
Al clicar sobre la opción 1 se mostrará un menú con las opciones:

- “Nuevo slot” -> Permite crear una restricción.
- “Ver slots” -> Permite ver el conjunto de slots sobre los que tiene permisos el profesor actual.

La opción 2 mostrará un menú con las opciones:

- “Ver listado” -> Muestra un listado con las notificaciones agrupadas por slots.

Opción “Nuevo slot”



Nuevo slot

Fecha inicio: dd / mm / yyyy Hora inicio: 16 : 26

Fecha fin: dd / mm / yyyy Hora fin: 16 : 26

Descripción:

Asignatura: -- Escoge una opción --

Grupos incluidos

Borrar seleccionados

Grupos excluidos

Borrar seleccionados

☒ Lista blanca ☐ Lista negra

Nueva url: http(s)://(www.)dominio.ext

Añadir url

Lista de Urls

Borrar seleccionadas

Estudiantes incluidos

Borrar seleccionados

Estudiantes excluidos

Borrar seleccionados

Añadir Slot

A la hora de añadir una nueva restricción, el profesor debe figurar en el sistema relacionado con uno o más grupos pertenecientes a una o varias asignaturas. De esa forma, el listado de asignaturas

disponibles saldrá en el desplegable “Asignatura”, en caso contrario, se indicará que no tiene asignaturas y no podrá crear la nueva restricción.

Una vez seleccionada una asignatura, saldrán los grupos disponibles de dicha asignatura en el cuadro de selección “Grupos incluidos” dentro de la caja marcada como “1”. A partir de aquí, también se rellenará el cuadro de selección “Estudiantes incluidos” y se podrán ir seleccionando, de forma individual o múltiple, y borrarlos de la restricción.


Al borrar grupos o estudiantes, en el caso de los primeros aparecerán en el cuadro de selección “Grupos excluidos” y en el caso de los segundos en el cuadro de selección “Estudiantes excluidos”. Nuevamente, de estos nuevos cuadros selectivos, los alumnos y grupos podrán ser eliminados, tras lo que aparecerán en el correspondiente cuadro selectivo de incluidos. De esta forma se pueden incluir y excluir alumnos y/o grupos de forma sencilla, ya que además se muestran ordenados alfabéticamente.

Si se eliminan todos los grupos, obviamente los estudiantes desaparecerán, ya que los que se muestran son los asociados a los grupos incluidos. Para poder crear una restricción debe haber estudiantes incluidos, y, por tanto, grupos.

Otro requisito para poder crear una restricción es que haya URLs incluidas siguiendo el formato “http(s)://(www.)dominio.ext”, siendo opcional lo incluido entre paréntesis. Obviamente se admite que haya subdominios en la URL también, pero no se admiten direcciones IP. En caso de querer borrar una URL incluida debe procederse de una forma similar a la que se haría para excluir grupos o alumnos.

Finalmente, una vez elegido el tipo de lista a elegir, blanca o negra, se podrá crear la restricción. Tras lo cual, se mostrará un mensaje indicando el resultado de la operación. En caso de que haya algún alumno incluido en la nueva restricción que ya tuviera otra para el horario indicado, ese alumno será excluido automáticamente y se mostrará que alumno es junto a la asignatura con la que tiene la restricción. Sin embargo, si todos los alumnos resultan excluidos, la restricción no se llegará a registrar y se informará del conjunto de colisiones.

Opción “Ver slots”

 profesor@uniovi.es Inicio Gestión de Slots Gestión de Informes Cerrar sesión

Listado Slots

Mostrar 10 registros por página Búsqueda

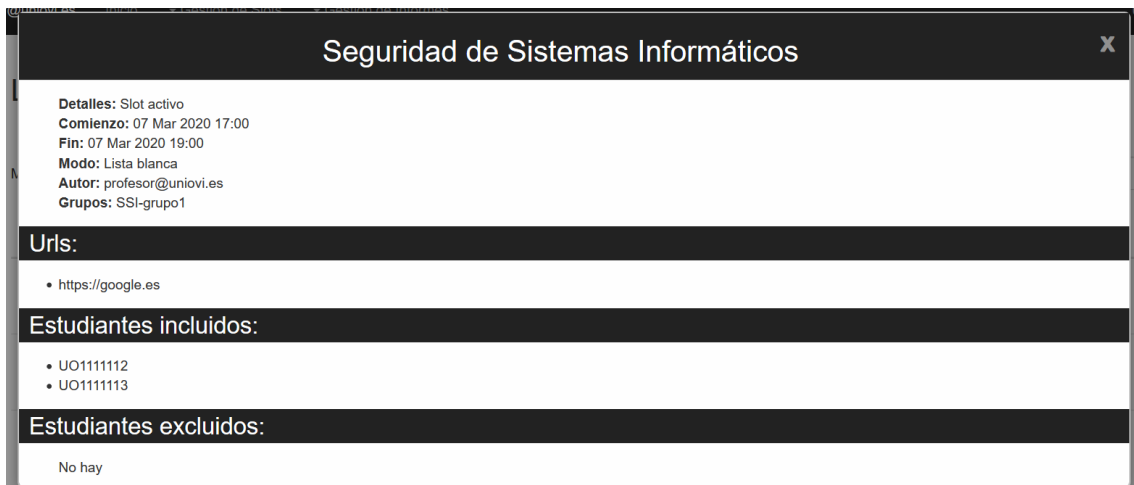
Slot	Asignatura	Comienzo	Fin	Modo de restricción	Autor	
Slot activo	Seguridad de Sistemas Informáticos	07 Mar 2020 17:00	07 Mar 2020 19:00	Lista blanca	profesor@uniovi.es	Ver detalles Modificar Eliminar
Slot prueba 2	Seguridad de Sistemas Informáticos	04 Mar 2020 21:06	04 Mar 2020 21:09	Lista blanca	profesor@uniovi.es	Ver detalles
Slot prueba 1	Diseño de Lenguajes de Programación	04 Mar 2020 20:57	04 Mar 2020 23:57	Lista negra	profesor2@uniovi.es	Ver detalles

Mostrando página 1 de 1 Anterior 1 Siguiente

En esta opción se muestra el listado de restricciones sobre las que el profesor que ha iniciado sesión tiene privilegios, es decir, todas las restricciones de asignaturas sobre las que imparte docencia. El listado admite diversos filtros que facilitan la visualización: paginación, búsqueda y ordenación.

Debido a que se trata de una información resumida de cada slot, se muestra junto a cada uno la opción “Ver detalles” que abre una ventana emergente con toda la información relacionada con la restricción. Así mismo, en caso de que la restricción aún no haya finalizado, se podrá eliminar y/o modificar, lo que resulta útil para excluir a alumnos que no vayan a realizar un examen y que necesiten ser incluidos en la restricción de otra asignatura.

Se muestra, a continuación, la opción de “Ver detalles”.



Seguridad de Sistemas Informáticos

Detalles: Slot activo
Comienzo: 07 Mar 2020 17:00
Fin: 07 Mar 2020 19:00
Modo: Lista blanca
Autor: profesor@uniovi.es
Grupos: SSI-grupo1

Urls:

- https://google.es

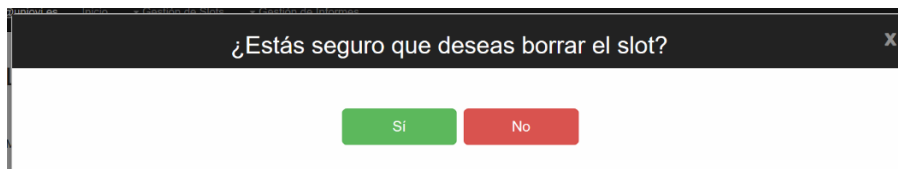
Estudiantes incluidos:

- UO111112
- UO111113

Estudiantes excluidos:

No hay

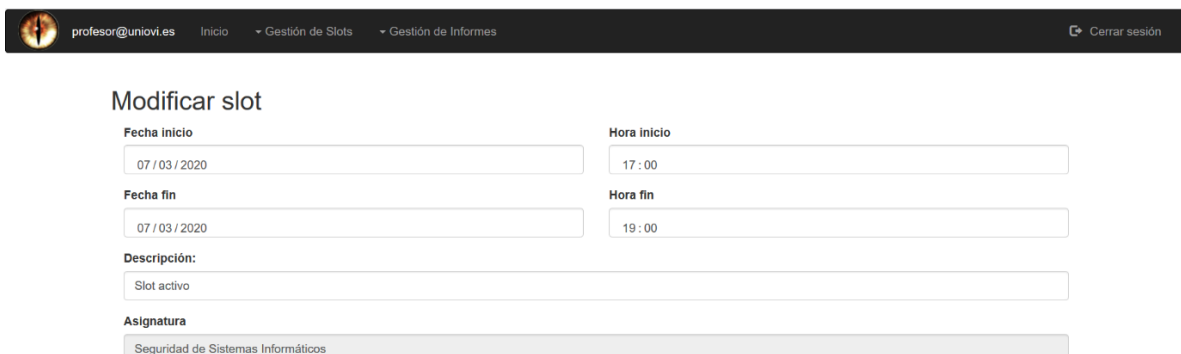
En el caso de que se desee eliminar el slot, se mostrará antes un cuadro de confirmación emergente para confirmar el borrado. De esta manera se reduce la posibilidad de que se borre una restricción de forma accidental.



¿Estás seguro que deseas borrar el slot?

Si No

Finalmente, en caso de que se decida modificar el slot, al clicar sobre la opción se mostrará una página similar al formulario en el que se añaden, a excepción del selector de asignatura, que en este caso estará restringido a la asignatura correspondiente al slot a modificar.



profesor@uniovi.es Inicio Gestión de Slots Gestión de Informes Cerrar sesión

Modificar slot

Fecha inicio
07 / 03 / 2020

Hora inicio
17 : 00

Fecha fin
07 / 03 / 2020

Hora fin
19 : 00

Descripción:
Slot activo

Asignatura
Seguridad de Sistemas Informáticos

En la ventana de modificación se deberán cumplir las mismas reglas que para la creación, es decir, deberá haber URLs declaradas, así como grupos y alumnos incluidos. De nuevo, en caso de que haya colisiones, en caso de incluir alumnos, se indicarán una vez solicitada la modificación y se excluirán automáticamente. Si se excluyen alumnos, que resultaran ya tener alguna notificación asociada a dicha restricción, esas notificaciones no serán borradas pero no se mostrarán en la opción correspondiente. No obstante, en caso de volver a incluirlos en la restricción, esas notificaciones se volverán a mostrar.

Opción “Ver listado”



Ver notificaciones

Seleccionar slot

-- Escoge una opción --

Al ir a la opción de “Ver listado” se mostrará un selector que permite seleccionar uno de los slots, de haberlos, y en caso contrario, indicará que no hay slots creados.

Una vez seleccionado el slot, en caso de que no tenga notificaciones asociadas, se mostrará una ventana similar a la siguiente.



Ver notificaciones

Seleccionar slot

Slot activo

☐ Mostrar solo con acciones no permitidas

Mostrar 10 registros por página

Búsqueda

Alumno	Asignatura	Grupo	Última notificación	Número de notificaciones	¿Acciones prohibidas?
--------	------------	-------	---------------------	--------------------------	-----------------------

No hay registros para mostrar

No hay registros disponibles

Anterior

Siguiente

Y en caso de que haya notificaciones asociadas.



Ver notificaciones

Seleccionar slot

Slot prueba 2

☐ Mostrar solo con acciones no permitidas

Mostrar 10 registros por página

Búsqueda

Alumno	Asignatura	Grupo	Última notificación	Número de notificaciones	¿Acciones prohibidas?
--------	------------	-------	---------------------	--------------------------	-----------------------

UO111111	Seguridad de Sistemas Informáticos	SSI-grupo1	04 Mar 2020 21:08:28	3	Si
----------	------------------------------------	------------	----------------------	---	----

[Ver detalles](#)

Mostrando página 1 de 1

Anterior

1

Siguiente

Los alumnos aparecerán en rojo en caso de que tengan notificaciones derivadas de acciones prohibidas, como visitar URLs no permitidas. Por ello, de cara a facilitar la búsqueda de alumnos que hayan cometido infracciones se da la opción de mostrar solo aquellos alumnos que las hayan cometido mediante un cuadro activable a la derecha del selector de slots. Así mismo, en el listado también se podrá paginar, ordenar y buscar de cara a facilitar la localización de información.

Al tratarse de un resumen de notificaciones por alumno y slot, para ver todas las notificaciones asociadas puede hacerse uso de la opción “Ver detalles” junto a cada registro que mostrará en una ventana emergente el listado completo para ese alumno y slot.

UO111111

Mostrar

10

registros por página

Búsqueda

Asignatura	Grupo	Acción	Fecha	Más información	Tiempo de vuelo	IP externa	IPs internas	¿Algo raro?
Seguridad de Sistemas Informáticos	SSI-grupo1	Página no permitida	04 Mar 2020 21:08:28	https://consent.yahoo.com/collectConsent?sessionId=3_cc-session_4e060930-3e8f-4e72-b5a8-4eb1a43077bd&lang=es-ES&inline=false	Desactivado	174.24.123.2	192.168.56.1, 192.168.0.109	
Seguridad de Sistemas Informáticos	SSI-grupo1	Página no permitida	04 Mar 2020 21:08:24	https://consent.yahoo.com/collectConsent?sessionId=3_cc-session_8de56035-a88f-46a9-8fe0-7ab3d3ce964f&lang=es-ES&inline=false	Desactivado	174.24.123.2	192.168.56.1, 192.168.0.109	
Seguridad de Sistemas Informáticos	SSI-grupo1	Inicio de sesión	04 Mar 2020 21:07:30		Desactivado	174.24.123.2	192.168.56.1, 192.168.0.109	

Mostrando página 1 de 1

Anterior

1

Siguiente

En este nuevo listado se podrán aplicar filtros de búsqueda, ordenación y paginación. Así mismo, las acciones que se consideren no permitidas aparecerán en rojo.

En la columna “¿Algo raro?” se indicarán, cuando corresponda, posibles incongruencias en los datos, como puede ser que se sospeche que la hora del ordenador en el que se originó la notificación pudo ser modificada tratando de evitar las restricciones, o simplemente que en ese ordenador hubiera habido algún tipo de error.

Los inicios de sesión mientras esté vigente la restricción pueden ser útiles para controlar la asistencia de los alumnos, ya que se registra la IP externa así como las direcciones IP internas del ordenador, lo que puede permitir saber en qué ordenador exacto se conectó.

Capítulo 8 ANEXOS



MEJORAS A FUTURO

1.1. Mejoras funcionales

Dentro de las mejoras que se podrían aplicar, a futuro, en términos funcionales están:

- Asegurar compatibilidad con otros sistemas operativos, para los clientes, ya que en la actualidad solamente se ha probado el sistema y desarrollado la documentación para un despliegue en clientes con Windows.
- Adaptar la funcionalidad de forma que la extensión sea compatible con más navegadores basados en el proyecto Chromium, Opera, Edge...
- Internacionalizar la extensión y el portal de administración para poder utilizarlo en otros idiomas y países.
- Permitir a los profesores ver información almacenada en copias de seguridad, para que de esta forma puedan ver datos de otros años en caso de ser necesario.
- Aumentar la variedad de restricciones que se pueden aplicar desde el portal de administración, es decir, permitir filtrar el contenido de una web, permitir o no marcadores y/o historial en función de lo que se determine para ese usuario, bloquear solo partes del contenido en caso de que sean datos procedentes de URL maliciosas y/o no permitidas, capturar y enviar pantallazos actuales del estado del navegador en caso de detectar una infracción o una posible, restringir el almacenamiento de datos de formularios...

1.2. Mejoras de seguridad

Las mejoras a nivel de seguridad son:

- Añadir un captcha en la identificación tanto de la extensión como del portal de administración.
- Mantener actualizado, de un listado de URL maliciosas, el sistema para denegar automáticamente el acceso a estas páginas dentro del entorno en el que se despliegue el proyecto.
- Evitar ataques que inutilicen el sistema, mediante el almacenamiento de notificaciones sin recibir ningún tipo de autenticación. Por ejemplo, las que se reciben de señales de vida y lo ocurrido en tiempo de vuelo. Tal y como está ahora el sistema, alguien podría enviar miles de notificaciones falsas al correspondiente punto final de la API y de esta forma saturar la base de datos. No se puede enviar siempre token en estas notificaciones, ya que una extensión sin identificar, porque ese día no haya restricciones, debe mandarlas para comprobar que está funcionando.



- Utilizar alguna herramienta, para aumentar la seguridad del servidor, del estilo de PortsEntry y mejor aún, otra herramienta que incluye la ya mencionada y otras protecciones adicionales, JShielder.

1.3. Nuevos enfoques

Tal y cómo ya se mencionó en el análisis de alternativas al comienzo de este documento, otro enfoque interesante sería el de desarrollar un navegador basado en Chromium de forma que se permitan más restricciones. Como es el poder esconder la barra de direcciones del navegador para que el usuario, en caso de que haga una fotografía y la cuelgue en redes sociales, esta no aparezca. Situación que lamentablemente ocurre más de lo habitual e incluso en entornos en que se debiera tener mucha precaución como es la Agencia Tributaria o el gobierno.

Finalmente, un enfoque mucho más ambicioso, sería el de crear una especificación que permitiera adaptar el proyecto, de la forma más fácil y con el menor esfuerzo posible, a diferentes entornos. De esta forma, se conseguiría un estándar mucho más robusto y fácil de aplicar en entornos críticos, como puede ser un banco, el gobierno...



REFERENCIAS

- afourney. (2014). *StackOverflow*. Obtenido de <https://stackoverflow.com/questions/20194722/can-you-get-a-users-local-lan-ip-address-via-javascript>
- Allen, R. (2017). *Active Directory Pro*. Obtenido de <https://activedirectorypro.com/how-to-use-rsop-to-check-and-troubleshoot-group-policy-settings/>
- Amazon Web Services, Inc. (2020). *Amazon Web Services On Demand*. Obtenido de <https://aws.amazon.com/es/ec2/pricing/on-demand/>
- Anderson, M. (2017). *Digital Ocean*. Obtenido de <https://www.digitalocean.com/community/tutorials/how-to-install-and-secure-mongodb-on-ubuntu-16-04>
- Anicas, M. (2015). *Digital Ocean*. Obtenido de <https://www.digitalocean.com/community/tutorials/ufw-essentials-common-firewall-rules-and-commands>
- Anicas, M. (2016). *Digital Ocean*. Obtenido de <https://www.digitalocean.com/community/tutorials/initial-server-setup-with-ubuntu-16-04>
- Anicas, M. (2017). *Digital Ocean*. Obtenido de <https://www.digitalocean.com/community/tutorials/how-to-secure-nginx-with-let-s-encrypt-on-ubuntu-16-04>
- ARC 42. (2019). Obtenido de <https://arc42.org/overview/>
- Barbosa, D. C. (2020). *Qué es un proxy y para qué sirve*. Obtenido de welivesecurity: <https://www.welivesecurity.com/la-es/2020/01/02/que-es-proxy-para-que-sirve/>
- Bearnes, B. (2016). *Digital Ocean*. Obtenido de <https://www.digitalocean.com/community/tutorials/how-to-set-up-a-node-js-application-for-production-on-ubuntu-16-04>
- Beverloo, P. (2019). *Peter Beverloo*. Obtenido de <https://peter.sh/experiments/chromium-command-line-switches/>
- blueimp, zhouxw-lang, & Connormiha. (2019). *GitHub*. Obtenido de <https://github.com/blueimp/JavaScript-MD5>
- Boucheron, B. (2017). *Digital Ocean*. Obtenido de <https://www.digitalocean.com/community/tutorials/how-to-set-up-time-synchronization-on-ubuntu-16-04>
- Chrome Developers. (2019). Obtenido de <https://developer.chrome.com/extensions/>



- Dan. (2019). *RegEx Testing*. Obtenido de <https://www.regextester.com/94502>
- Developer Guide Chrome Extensions*. (2019). Obtenido de <https://developer.chrome.com/extensions/devguide>
- Ellingwood, J. (2016). *Digital Ocean*. Obtenido de <https://www.digitalocean.com/community/tutorials/how-to-install-nginx-on-ubuntu-16-04>
- Ellingwood, J. (2016). *Digital Ocean*. Obtenido de <https://www.digitalocean.com/community/tutorials/how-to-set-up-nginx-server-blocks-virtual-hosts-on-ubuntu-16-04>
- Ellingwood, J. (2016). *Digital Ocean*. Obtenido de <https://www.digitalocean.com/community/tutorials/how-to-create-a-self-signed-ssl-certificate-for-nginx-in-ubuntu-16-04>
- Foundation, Mozilla. (2020). *Mozilla SSL Configuration Generator*. Obtenido de <https://ssl-config.mozilla.org/>
- GlassDoor*. (2020). Obtenido de https://www.glassdoor.es/Sueldos/spain-sueldo-SRCH_IL0,5_IN219.htm
- GlassDoor*. (2020). Obtenido de https://www.glassdoor.es/Sueldos/jefe-de-proyecto-sueldo-SRCH_KO0,16.htm
- GlassDoor*. (2020). Obtenido de https://www.glassdoor.es/Sueldos/arquitecto-de-software-sueldo-SRCH_KO0,22.htm
- GlassDoor*. (2020). Obtenido de https://www.glassdoor.es/Sueldos/software-tester-sueldo-SRCH_KO0,15.htm
- Google Inc. (2020). *Google Chrome Enterprise Help*. Obtenido de <https://support.google.com/chrome/a/answer/9296680?hl=en>
- Google Inc. (2020). *Google Chrome Enterprise Help*. Obtenido de https://support.google.com/chrome/a/answer/7532015?hl=en&ref_topic=9023098
- Goujon, A. (2012). *welivesecurity*. Obtenido de <https://www.welivesecurity.com/la-es/2012/09/10/vpn-funcionamiento-privacidad-informacion/>
- Inc, Google. (2019). *Google Chrome Enterprise*. Obtenido de <https://cloud.google.com/chrome-enterprise/browser/download/>
- Inc, MongoDB. (2019). *MongoDB documentation*. Obtenido de <https://docs.mongodb.com/manual/>
- Indeed*. (2020). Obtenido de <https://www.indeed.es/salaries/analista-programador-Salaries>
- Indeed*. (2020). Obtenido de <https://www.indeed.es/salaries/administrador-de-sistemas-Salaries>



- Indeed*. (2020). Obtenido de <https://www.indeed.es/salaries/analista-de-seguridad-inform%C3%A1tica-Salaries>
- Jacques, N. (2020). *SitePoint*. Obtenido de <https://www.sitepoint.com/configuring-nginx-ssl-nodejs/>
- LDAPJS documentation*. (2019). Obtenido de <http://ldapjs.org/client.html>
- Limited, Impero Solutions. (2020). *Impero Software*. Obtenido de <https://support.imperosoftware.com/support/solutions/articles/44001042560-deploy-google-chrome-extension-via-group-policy>
- LTD, SpryMedia. (2019). *DataTables*. Obtenido de <https://datatables.net/examples/styling/bootstrap4>
- Minifier*. (2019). Obtenido de <https://www.minifier.org/>
- Mortensen, P., & Woitasen, D. (2018). *StackOverflow*. Obtenido de <https://stackoverflow.com/questions/10175812/how-to-create-a-self-signed-certificate-with-openssl>
- Mozilla Foundation. (2019). *MDN web docs*. Obtenido de <https://developer.mozilla.org/en-US/docs/Web/JavaScript/>
- Nodejs documentation API*. (2019). Obtenido de <https://nodejs.org/docs/latest-v10.x/api/>
- Nunjucks*. (2019). Retrieved from <https://mozilla.github.io/nunjucks/>
- OutSystems. (2020). *OutSystems*. Obtenido de https://success.outsystems.com/Support/Enterprise_Customers/Installation/Install_a_trusted_root_CA_or_self-signed_certificate
- Proyecto Chromium*. (2019). Obtenido de <https://www.chromium.org/>
- Redondo, J. M. (17 de 6 de 2019). *ResearchGate*. Obtenido de https://www.researchgate.net/publication/327882831_Plantilla_de_Proyectos_de_Fin_de_Carrera_de_la_Escuela_de_Informatica_de_Oviedo
- Securly Support. (2020). *Securly*. Obtenido de <https://support.securly.com/hc/en-us/articles/206688537-How-to-push-the-Securly-SSL-certificate-with-Active-Directory-GPO->
- Span, D. (2017). *Dennis Span*. Obtenido de <https://dennisspan.com/google-chrome-on-citrix-deep-dive/#UsingMicrosoftGroupPolicies>
- Tedhudek. (2017). *Microsoft Docs*. Obtenido de <https://docs.microsoft.com/en-us/windows-hardware/drivers/install/trusted-root-certification-authorities-certificate-store>
- Touesnard, B. (2017). *Delicious Brains*. Obtenido de <https://deliciousbrains.com/ssl-certificate-authority-for-local-https-development/>



UMLet Team. (2020). *UMLet*. Obtenido de <http://www.umlet.com/umletino/umletino.html>

Virdó, H. (2016). *Digital Ocean*. Obtenido de
<https://www.digitalocean.com/community/tutorials/how-to-set-up-a-firewall-with-ufw-on-ubuntu-16-04>

Visual Paradigm International Ltd. (2020). *Visual Paradigm Online*. Obtenido de
<https://online.visual-paradigm.com>

Windows OS Hub. (2019). *WindowsOSHUB*. Obtenido de <http://woshub.com/how-to-deploy-certificate-by-using-group-policy/>