Contenido

[1 Autores del trabajo, planificación y entrega 2](#_Toc481430928)

[1.1 Autores 2](#_Toc481430929)

[1.2 Planificación 2](#_Toc481430930)

[1.3 Entrega 2](#_Toc481430931)

[2 Requisitos del prototipo a implementar 3](#_Toc481430932)

[2.1 Requisitos funcionales 3](#_Toc481430933)

[2.2 Otros requisitos 4](#_Toc481430934)

[3 Criterios de comparación en la implementación 4](#_Toc481430935)

[3.1 Criterio 1: Facilidad de instalación 4](#_Toc481430936)

[3.2 Criterio 2: Idioma 4](#_Toc481430937)

[3.3 Criterio 3: Asistente de uso 4](#_Toc481430938)

[3.4 Criterio 4: Diseño de interfaz 4](#_Toc481430939)

[3.5 Criterio 5: Intuitividad 5](#_Toc481430940)

[3.6 Criterio 6: Utilización de recursos 5](#_Toc481430941)

[3.7 Criterio 7: Parámetros a analizar 5](#_Toc481430942)

[3.8 Criterio 8: Generación de informes 5](#_Toc481430943)

[3.9 Criterio 9: Requisitos previos a la instalación 5](#_Toc481430944)

[3.10 Criterio 10: Ayudas y tutoriales 5](#_Toc481430945)

[3.11 Criterio 11: Tiempo invertido en realizar una prueba desde cero 6](#_Toc481430946)

[4 Informe de la prueba con WebLoad 7](#_Toc481430947)

[4.1 Diseño de la prueba 7](#_Toc481430948)

[4.2 Documentación de construcción de las pruebas 7](#_Toc481430949)

[4.3 Documentación de la configuración de la consola de testeo 7](#_Toc481430950)

[4.4 Documentación del análisis de resultados 7](#_Toc481430951)

[5 Informe de la prueba con LoadRunner 8](#_Toc481430952)

[5.1 Diseño de la prueba 8](#_Toc481430953)

[5.2 Documentación de construcción de las pruebas 8](#_Toc481430954)

[5.3 Documentación de la configuración de la consola de testeo 8](#_Toc481430955)

[5.4 Documentación del análisis de resultados 8](#_Toc481430956)

[6 Comparación de las dos implementaciones 9](#_Toc481430957)

[6.1 Evaluación de los criterios en la implementación usando WebLoad 9](#_Toc481430958)

[6.2 Evaluación de los criterios en la implementación usando LoadRunner 9](#_Toc481430959)

[7 Comparación de la implementación de las tecnologías 10](#_Toc481430960)

[8 Conclusiones 12](#_Toc481430961)

# Autores del trabajo, planificación y entrega

## Autores

El grupo 8 del turno de tarde (T8) está formado por:

* Pedro Ignacio Santiago (Coordinador).
* BogdanIonutCiobanu
* Sandra Antoral

La comparación se realizará entre WebLOAD y LoadRunner, dos herramientas empresariales de testeo de rendimiento, carga y estrés, en aplicaciones y servicios web.

## Planificación

En el siguiente enlace se puede visualizar la planificación de tareas por puntos del trabajo, organizado y adjudicado a cada uno de los miembros del grupo.

[*https://www.tomsplanner.es/public/tg2-t8*](https://www.tomsplanner.es/public/tg2-t8)

El peso de este trabajo en la calificación total de la asignatura es de un 10%, por tanto, requiere de una dedicación de 15 horas del total de 150 horas de la asignatura.

Existen 6 grupos de tareas que se corresponden a los 6 apartados del trabajo con sus correspondientes subtareas. También, están adjudicadas al trabajo realizado por cada componente del grupo, así como de las horas realizadas por cada uno de nosotros.

Se aprecia que la suma de los trabajos realizados en los apartados da un total de 15 horas por persona, siendo el total de horas invertidas en este trabajo de 45.

## Entrega

En el siguiente link, se redirigirá al repositorio de GitHub del grupo T8 donde se podrán encontrar los archivos a calificar con la siguiente nomenclatura:

[*https://github.com/PedroISP/TG2*](https://github.com/PedroISP/TG2)

* Informe del trabajo: con el nombre TG3\_final.docx
* Presentación del trabajo: TG3\_final.pptx
* Prototipos obtenidos de la implementación de cada una de las tecnologías (Incluye todos los archivos necesarios para la instalación y uso de cada prototipo):
  + PrototipoWebLoad.zip
  + PrototipoLoadRunner.zip

# Requisitos del prototipo a implementar

Partimos de la premisa de que nuestra tecnología no parte de ningún desarrollo ni necesita de ningún diseño para ponerla en funcionamiento.

Las dos suits de aplicaciones para testear servicios, aplicaciones y sistemas web cumplen con un montón de funcionalidades que nos permiten en ambos casos realizar casi cualquier tarea relacionada con la investigación.

En conclusión los requisitos funcionales y los criterios que propondremos a lo largo del trabajo, se centran más en aspectos no tan técnicos como pueden ser la facilidad de uso, instalación y prerrequisitos, tiempos de desarrollo de las pruebas a realizar, tiempos de ejecución de las pruebas y algunos más que detallaremos a continuación.

*El objetivo del proyecto es comparar la implementación de un mismo prototipo de sistema utilizando dos tecnologías diferentes (A y B).*

*Es importante cumplimentar este apartado antes de empezar a implementar el prototipo de cada tecnología, porque ambos prototipos deben cumplir los requisitos que se establezcan en este apartado. Si se van a crear dos equipos de trabajo, uno para cada prototipo, el contenido de este apartado es lo que han de compartir ambos equipos como punto de partida.*

*Cuanto más detallados sean los requisitos, mayor será la precisión en la comparación que se realizará al final del trabajo. Se trata de conseguir dos prototipos con igual funcionalidad, pero utilizando diferentes tecnologías.*

*Se puede dar libertad a los equipos de desarrollo en cuanto al diseño, pero la funcionalidad debe ser lo más parecida posible. Por ejemplo, no es necesario que los colores utilizados en las pantallas sean exactamente los mismos en ambos prototipos, a no ser que los miembros del grupo lo hayan decidido así, en cuyo caso, esos detalles de colores deben incluirse en el catálogo de requisitos, para que ambos equipos los cumplan.*

## Requisitos funcionales

En la siguiente tabla se indicará el catálogo de requisitos funcionales del sistema.

| **REQ.** | **DESCRIPCIÓN** |
| --- | --- |
| RF01 | Facilidad de instalación |
| RF02 | Idioma |
| RF03 | Asistente de uso |
| RF04 | Diseño de interfaz |
| RF05 | Intuitividad |
| RF06 | Utilización de recursos |
| RF07 | Parámetros a analizar |
| RF08 | Generación de informes |

## Otros requisitos

En la siguiente tabla se indicará el catálogo de requisitos no funcionales del sistema.

| **REQ.** | **DESCRIPCIÓN** |
| --- | --- |
| R01 | Requisitos previos a la instalación |
| R02 | Ayudas y tutoriales |
| R03 | Tiempo invertido en realizar una prueba desde cero |

# Criterios de comparación en la implementación

En el trabajo TG2 se definieron criterios de comparación de las dos tecnologías a nivel teórico.

En este trabajo se definen criterios para la comparación de la implementación de las tecnologías en la construcción del prototipo de sistema de ejemplo, cuyos requisitos son los establecidos en el apartado 2.

## Criterio 1: Facilidad de instalación

* Nombre del criterio: Facilidad de instalación de todos los componentes del sistema.
* Descripción: Facilidad que ofrece el instalador de la aplicación a la hora de realizar una instalación predeterminada y básica para realizar una prueba normal.
* Tipo de valor:Escala de 1 a 10.

## Criterio 2: Idioma

* Nombre del criterio: Idioma
* Descripción: Lengua en la que está desarrollada la aplicación
* Tipo de valor: Cadena de caracteres

## Criterio 3: Asistente de uso

* Nombre del criterio: Asistente de uso como guía para utilizar la aplicación
* Descripción: Magnitud que nos permite evaluar la relación facilidad-uso de los asistentes para la generación de pruebas
* Tipo de valor: Escala de 1 a 10

## Criterio 4: Diseño de interfaz

* Nombre del criterio: Diseño de la interfaz
* Descripción: Magnitud que nos permite evaluar el diseño de la interfaz para modificar, insertar y borrar datos y parámetros para la ejecución de pruebas
* Tipo de valor: Escala de 1 a 10

## Criterio 5: Intuitividad

* Nombre del criterio: Intuitividad
* Descripción: Magnitud que nos permite evaluar el grado de diseño que ofrece la aplicación para guiarnos fácilmente por la interfaz
* Tipo de valor: Escala de 1 a 10

## Criterio 6: Utilización de recursos

* Nombre del criterio: Utilización de recursos del sistema
* Descripción: Nos permitirá diferenciar las dos herramientas en función del impacto que tienen sobre el sistema sobre el que se ejecutan las pruebas.
* Tipo de valor: Enumerado [Alto-Bajo]

## Criterio 7: Parámetros a analizar

* Nombre del criterio: Parámetros a analizar
* Descripción: Facilidad de elección sobre una serie de parámetros a analizar referentes a la prueba a testear.
* Tipo de valor: Escala de 1 a 10

## Criterio 8: Generación de informes

* Nombre del criterio: Generación de informes
* Descripción: Magnitud que nos permite evaluar la capacidad de generar informes y el estilo y morfología de estos así como, el número de formatos posibles.
* Tipo de valor: Escala de 1 a 10

## Criterio 9: Requisitos previos a la instalación

* Nombre del criterio: Requisitos previos a la instalación
* Descripción: Programas y características necesarias del sistema previas a la instalación de las aplicaciones
* Tipo de valor: Cadena de caracteres

## Criterio 10: Ayudas y tutoriales

* Nombre del criterio: Ayudas y tutoriales
* Descripción: Magnitud que evalua la relación calidad-cantidad de los tutoriales accesibles on-line
* Tipo de valor: Magnitud de 1 a 10

## Criterio 11: Tiempo invertido en realizar una prueba desde cero

* Nombre del criterio: Tiempo invertido
* Descripción: Tiempo invertido en realizar una prueba desde que se abre el programa hasta que se extraen las analíticas en un archivo de lectura.
* Tipo de valor: Minutos

# Informe de la prueba conWebLoad

# Informe de la prueba con LoadRunner

Se trata de incluir en este apartado la documentación del desarrollo del proyecto de implementación, utilizando la tecnología B, del sistema cuyos requisitos funcionales se enumeraron en el apartado 2.

Para realizar las pruebas con LoadRunner, a diferencia de WebLoad, no hay que hacer ningún paso adicional después de instalar el programa con el instalador que nos provee HP.

LoadRunner, trae como opcional la posibilidad de instalar una base de datos y ejecutar una web de ejemplo en la que se pueden reservar vuelos. Esto es para realizar las pruebas y practicar sobre una web de prueba. En nuestro caso, no lo instalaremos ya que haremos las pruebas sobre una web totalmente online. Por lo tanto, como hemos mencionado al principio, después de instalar LoadRunner con su instalador, no hay que hacer ningún paso adicional.

## Diseño de la prueba

La prueba a realizar se realizará sobre la página:

* <http://www.webloadmpstore.com/index.php>

En la cual realizaremos el siguiente recorrido:

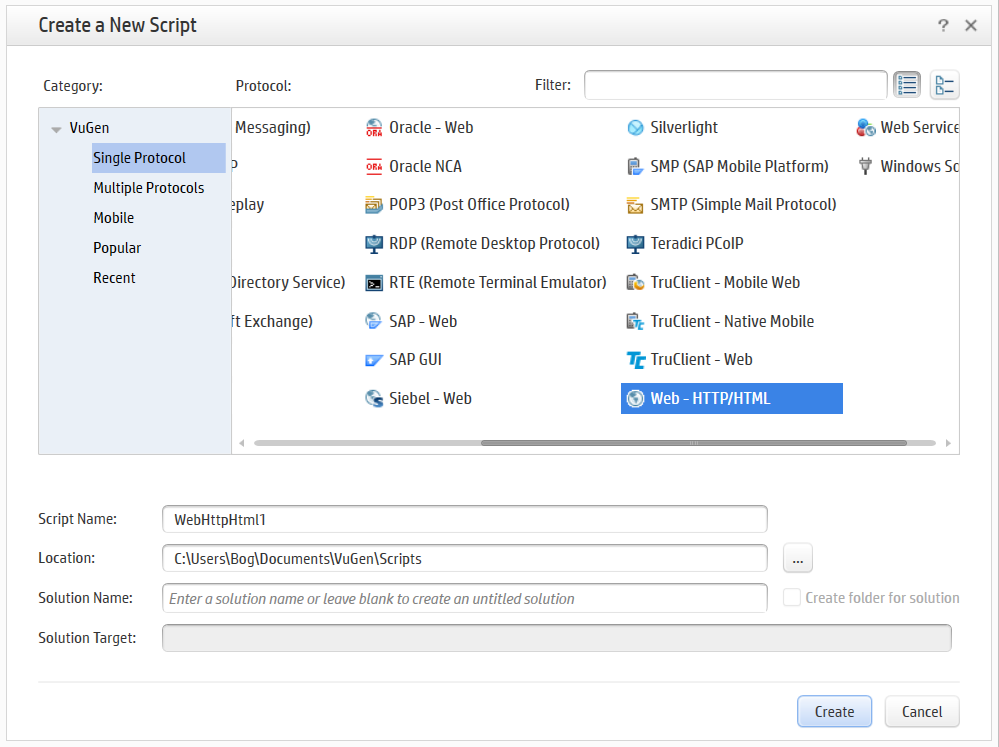
1. Ingresar en <http://webloadmpstore.com/>
2. Hacer login
   1. User: demo
   2. Pass: demo
3. Hacer búsqueda, arriba a la derecha, “CONSOLE”
4. Click en “WebLOADConsole Demo”
5. Click en “CART” (Carro de la compra)
6. Borrar producto

## Documentación de construcción de las pruebas

Para comenzar, hay que saber que LoadRunner tiene 3 aplicaciones separadas, la primera llamada "Virtual User Generator", la segunda llamada "Controller" y la tercera llamada "Analysis".

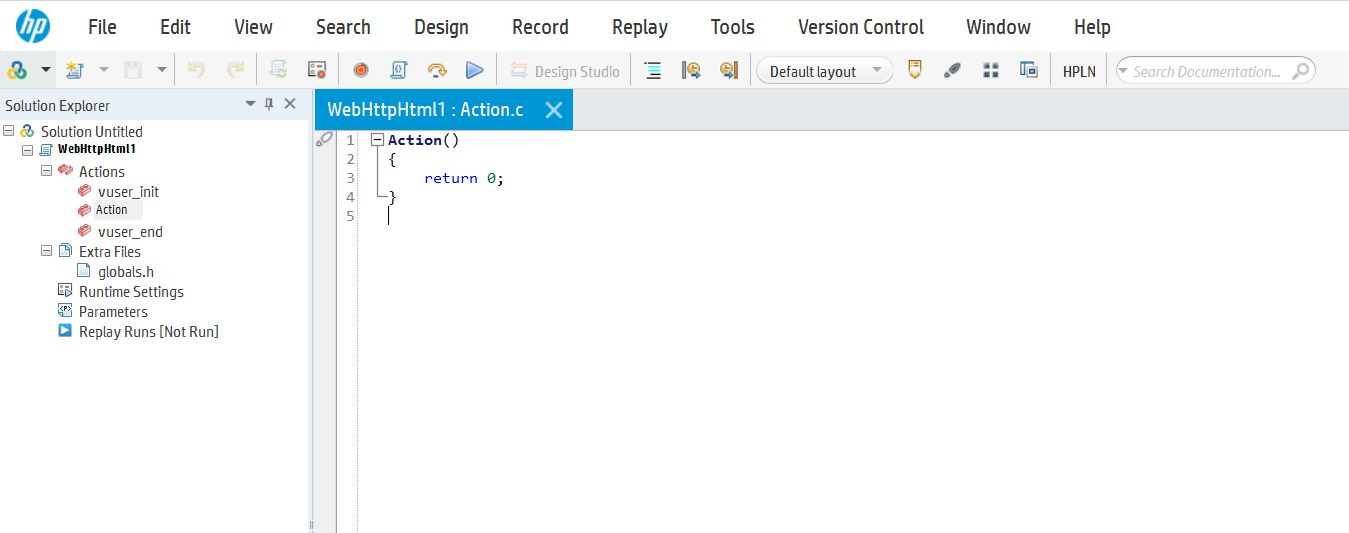
La construcción de la prueba comienza con la primera aplicación "Virtual User Generator", en la que diseñaremos el recorrido que realizara por la web el usuario virtual.

Abrimos la aplicación y una vez dentro nos dirigimos a la esquina izquierda superior, y pinchamos en "File>New Script and Solution".



En la pantalla superior, tenemos que seleccionar el protocolo utilizado, en nuestro caso se trata de una web simple, así que elegimos "Web - HTTP/HTML". Podemos darle también un nombre especifico al script o dejarlo tal cual esta. La ruta en "Location" es mejor dejarlo como aparece por defecto ya que el segundo programa, como veremos más adelante, buscara estos scripts en esa ruta.

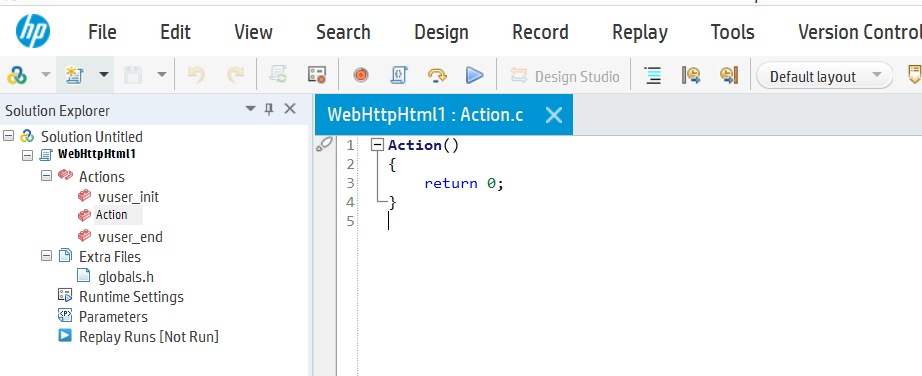
Una vez le demos a Create tendremos lo siguiente:



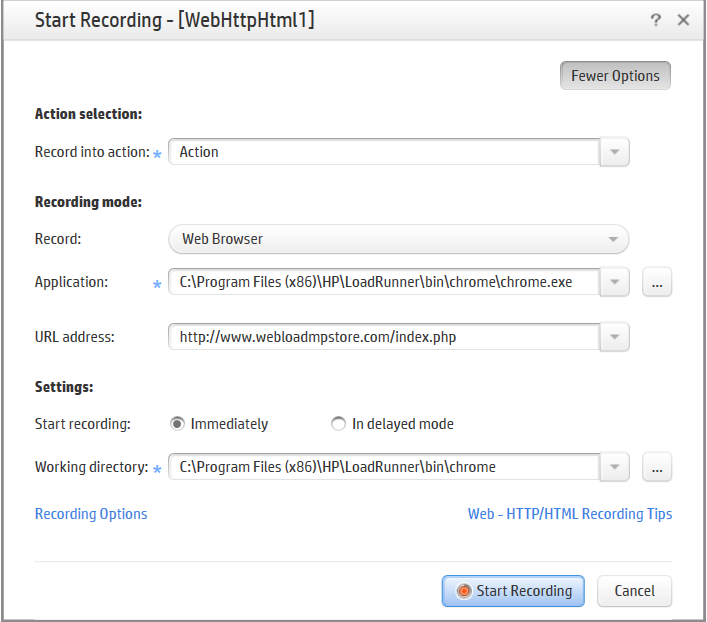
Este apartado corresponde al script que indicara al usuario virtual que hacer. Como vemos al principio estará totalmente vacío.

Como podemos ver en el desplegable de la izquierda hay bastantes archivos, aunque el más importante es "Action" que corresponde a que hará el usuario virtual. Otra parte importante del desplegable es "Runtime Settings", en este apartado podremos indicar cuantas iteraciones queremos que realice el usuario de este script aunque en nuestro caso no lo tocaremos y lo dejaremos por defecto en 1.

El siguiente paso, es grabar el script, si no estamos ya en el apartado "Action", le damos doble click en el desplegable de la izquierda y una vez abierto podemos seleccionar del menú superior la opción "Record > Record" o bien darle al siguiente botón rojo (se señala con una flecha para que se vea claramente):



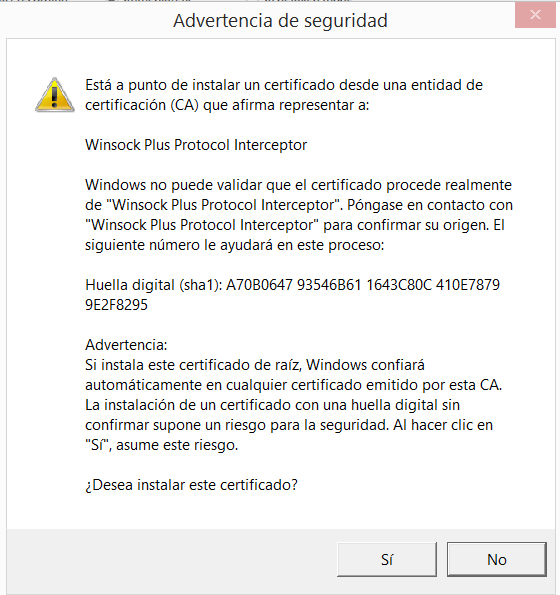
Una vez le demos a dicho botón, se nos abrirá una pestaña de configuración que deberemos configurar como la siguiente (para este ejemplo claro):



Como podemos ver, en "Application" aparece una ruta hacia chrome.exe, esto es debido a que por defecto el navegador predeterminado que se nos abrirá para grabar el recorrido del usuario es Internet Explorer pero, si damos a los tres puntitos que están al lado "..." podemos buscar dentro de la carpeta de LoadRunner otros navegadores como son por ejemplo Firefox o en este caso Chrome.

URL Address es la dirección a la web a la que se realizara el testeo, cuando le demos al botón "Start Recording" dicha web se nos abrirá en el navegador que hemos elegido en el paso anterior.

El apartado de "Settings" lo dejaremos tal y como viene por defecto y pulsamos el botón "Start Recording".



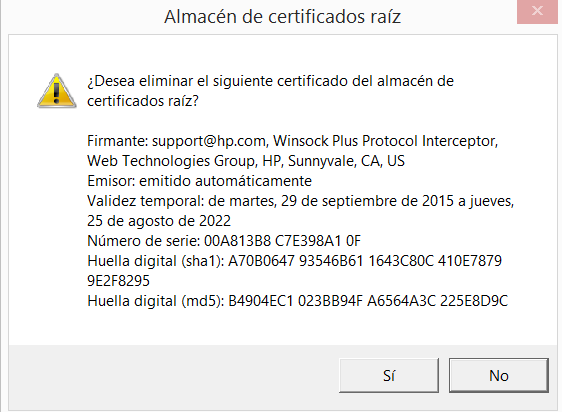
Se nos preguntara si queremos instalar un certificado al cual tenemos que decirle que si, cuando acabemos la prueba, se nos preguntara si queremos borrar el certificado así que no habrá ningún problema, no se quedara en el ordenador.



Como podemos ver en la captura superior, cuando arranca el programa, tendremos la web a la que queremos hacerle la prueba y a la derecha nos aparecerá una barra flotante que podemos mover a nuestro antojo. Una vez se abra la página como en la captura superior, tendremos que realizar el recorrido por el cual pasara nuestro usuario virtual. Todo se quedara grabado de forma automática, el tiempo entre acciones, también es tenido en cuenta, por ejemplo, si tardamos varios segundos en entrar a un enlace el usuario virtual también lo hara.

Cuando acabemos con el recorrido del usuario, en la barra flotante tendremos que darle al botón de stop (segundo botón empezando por la izquierda de la barra flotante).

Una vez parado, se nos preguntara si queremos eliminar el certificado que hemos instalado antes, le indicaremos que si:

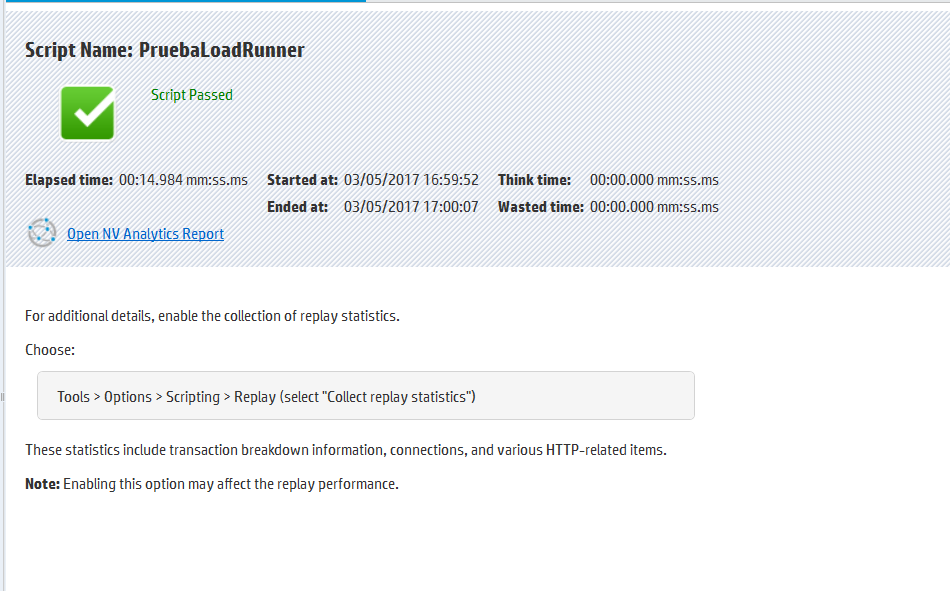


Una vez realizado lo anterior, se nos habrá generado el script del usuario virtual:



En este paso, tendremos que comprobar que el script generado por el programa funciona correctamente, por lo que en la barra superior le damos al botón "Replay" o en el menú superior "Replay>Run", también nos valdría pulsar "F5" en nuestro teclado.

Esto lo que hará será ejecutar el usuario virtual para ver que funciona correctamente, si todo está bien, tendrá que salirnos la siguiente pantalla:

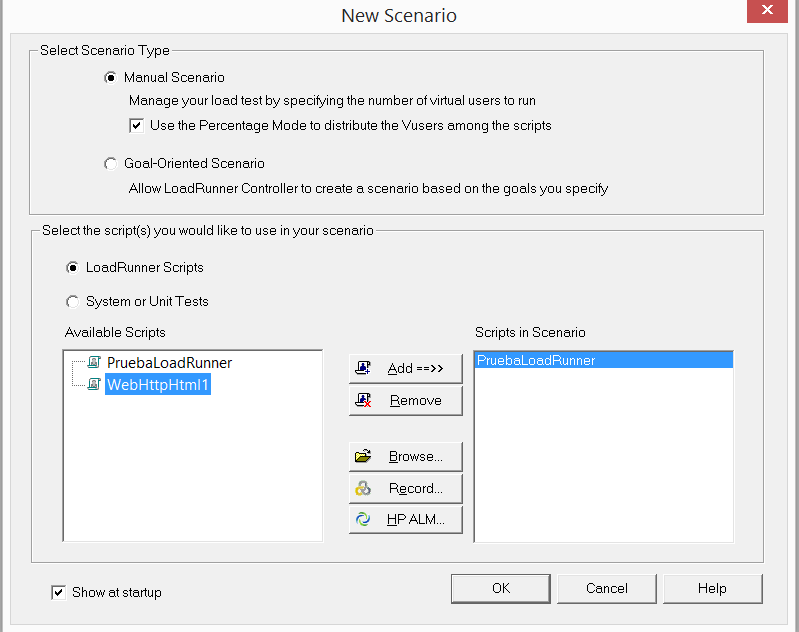


Si todo está bien, y el script no ha fallado, guardamos y ya hemos acabado con esta parte del programa. Si por el contrario el script ha fallado, podremos editar el script manualmente en busca del fallo o bien volver a grabar el script desde el principio como hemos indicado antes.

## Documentación de la configuración de la consola de testeo

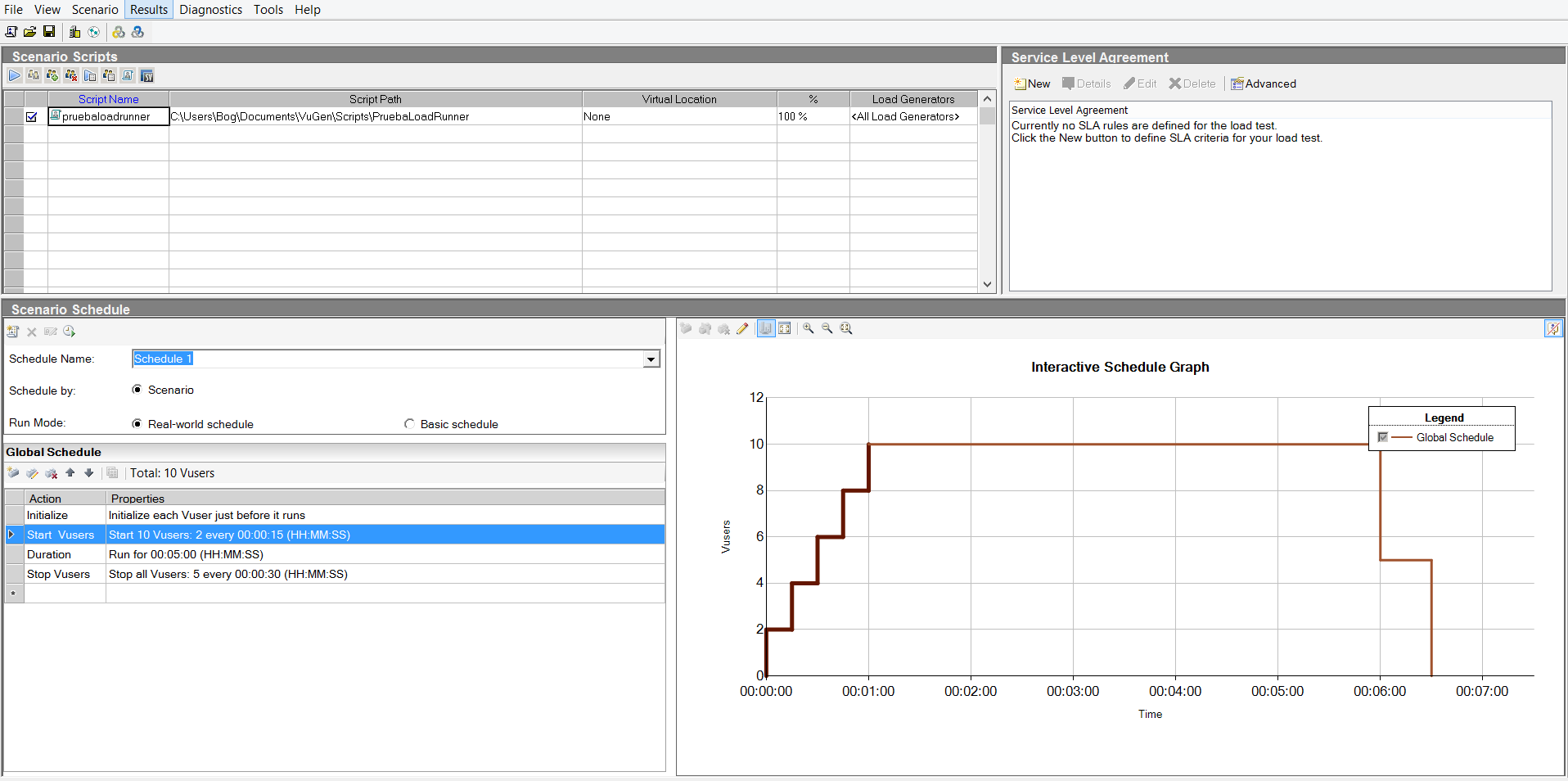
Para comenzar con este apartado, tenemos que abrir la segunda aplicación que nos provee el programa LoadRunner. Se trata de la aplicación "Controller".

Una vez abierto, tendremos que crear un nuevo escenario, eligiendo el script que hemos creado anteriormente para los usuarios virtuales:

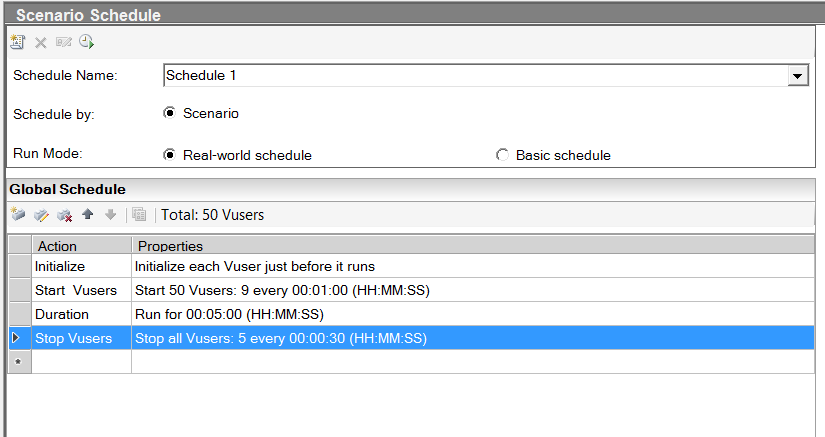


Y lo añadimos en "Scripts in Scenario". Pulsamos OK.

Se nos abrirá la siguiente pantalla:



En esta pantalla, que es básicamente el controlador que se encargara de generar carga sobre la web con usuarios virtuales, en la parte inferior izquierda tendremos que configurarlo de la siguiente manera:



Como podemos ver, el "Run Mode" se configura en la opción "Real-world schedule", esto significa que los usuarios, accederán al sistema de forma progresiva, y se irán también de forma progresiva, una prueba muy realista. En cambio, la opción "Basic Schedule" lo que hace es introducir a todos los usuarios virtuales de golpe y después sacarlos de golpe, algo que no podríamos considerar como una prueba de lo que en la realidad ocurrirá aunque sí que nos serviría para hacer pruebas de carga y ver cuántos usuarios simultáneos aguantaría el sistema.

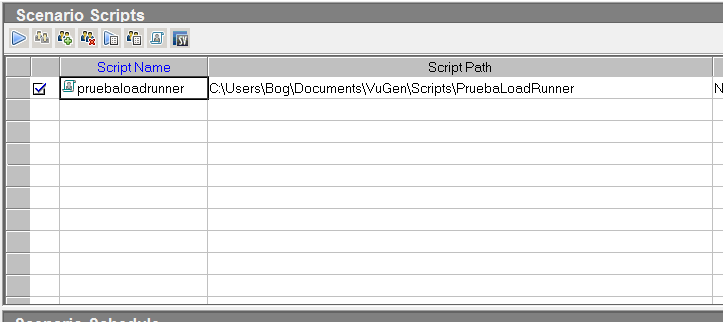
Para cambiar cualquier opción que está en la columna "Properties", solo tendremos que hacer doble click sobre ella y se nos abrirá un cuadro de dialogo donde podremos editarlo.

Como podemos observar, se inicializaran 50 usuarios progresivamente, 9 usuarios cada minuto. La prueba durara 5 minutos y se pararan 5 usuarios cada 30 segundos.

Estos parámetros han sido establecidos así, para hacer la prueba lo más parecida posible a la realizada con WebLoad.

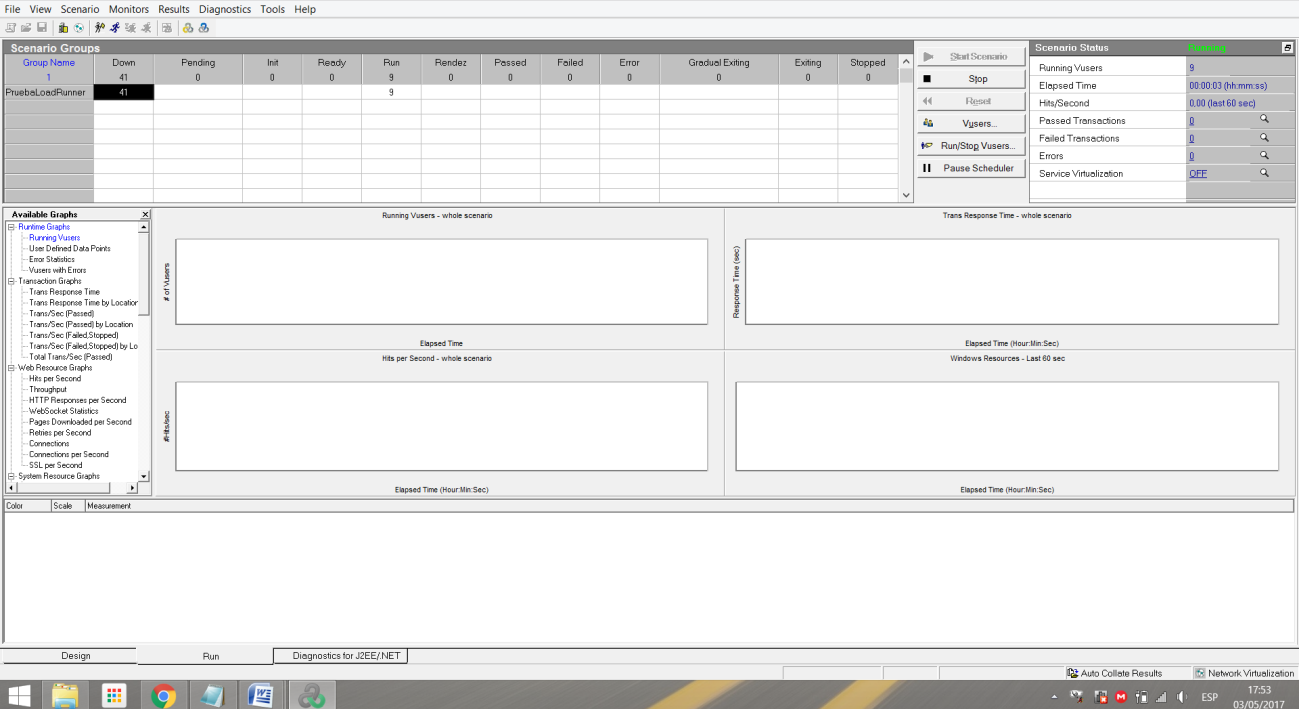
Una vez que tengamos configurado lo anterior, podemos proceder a realizar la prueba.

Lo único que tendremos que hacer es darle al botón del "Play" que tenemos en la esquina izquierda superior:



Una vez realizado lo anterior, tendremos que esperar a que los usuarios virtuales acaben.

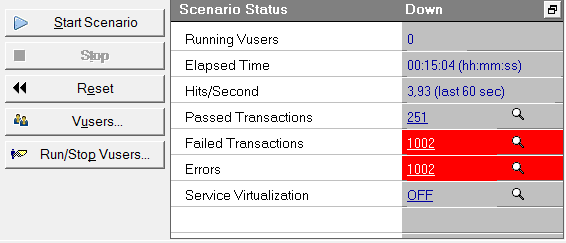
Como veremos, se nos abrirá otra vista en la que veremos cómo se van ejecutando poco a poco los usuarios virtuales y si hay errores o no.



En esta pestaña, también podemos ver gráficos de las operaciones que están sucediente en tiempo real con los usuarios virtuales. Además de las 4 gráficas que vienen por defecto, en el desplegable de la izquierda hay muchos más que podremos añadir.

Como podemos ver, la prueba en total ha tardado unos 15 minutos, 5 minutos para introducir los usuarios progresivamente, 5 minutos dando toda su potencia con los 50 usuarios simultáneos y por ultimo otros 5 minutos finalizando progresivamente cada usuario.

Este sería el resultado de la prueba sin entrar en el aspecto del análisis aun:



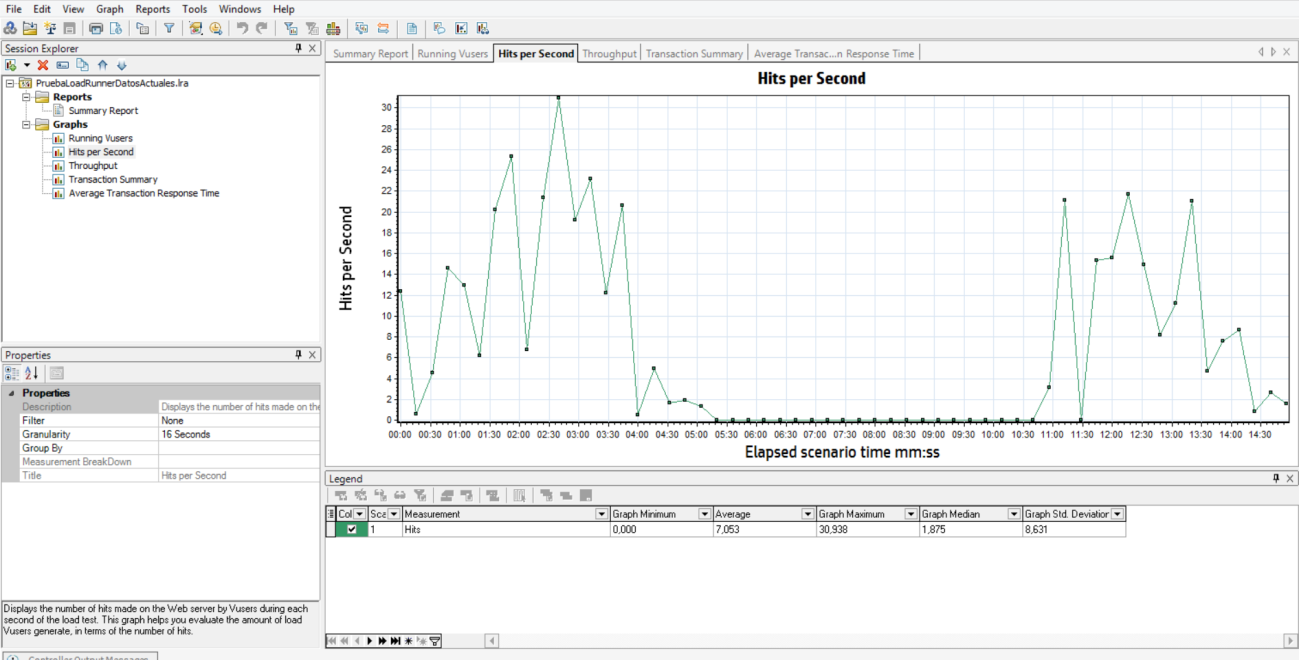
Como podemos ver, han fallado 1002 transacciones, esto es debido a que a partir de los 30 usuarios, el sistema no ha aguantado mas, y no se han podido realizar más transacciones de los usuarios virtuales. Con 50 usuarios virtuales activos, todo han sido peticiones fallidas. Cuando se han ido sacando los usuarios, en los últimos 5 minutos, el sistema pudo recibir peticiones otra vez y se realizaron algunas mas de forma satisfactoria.

En este caso, podemos afirmar a groso modo, que el sistema, no soporta 50 usuarios simultáneos ni de lejos.

Por último, para analizar los datos generados en ese apartado, en la barra de tareas tendremos que seleccionar "Results>Analize results". Esto, nos abrirá el tercer programa llamado Analysis que veremos a continuación.

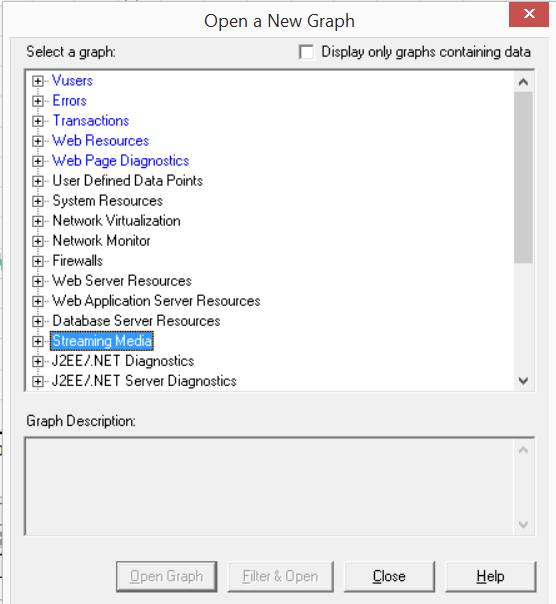
## Documentación del análisis de resultados

En este apartado, solo queda visualizar los datos obtenidos del testeo realizado:



Por defecto, solo nos aparecerá un resumen general y 5 gráficas, como se puede observar en la imagen superior, aunque, podemos añadir de forma adicional mas gráficas:

En la barra de herramientas superior, pulsaremos en "Graph>Add New Graph..." y nos mostrara lo siguiente:



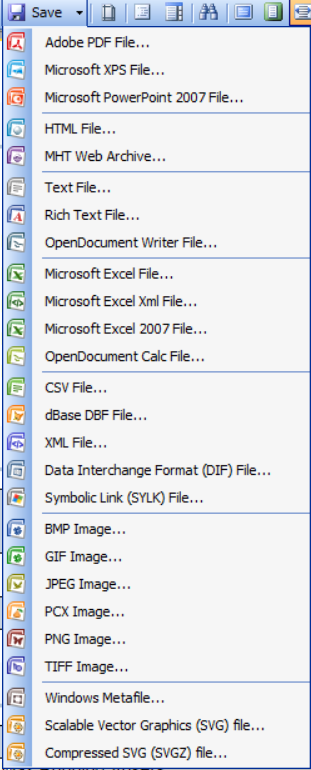
Cada una de las opciones mostradas arriba, es un desplegable con muchísimos gráficos dentro de cada una, es probable que este marcada la opción "Display only graphs containing data" con lo cual, no aparecerán todos los gráficos. Recordemos que al principio, cuando diseñamos la prueba para los usuarios virtuales, seleccionamos el protocolo que queríamos testear, en nuestro caso se trataba de HTTP/HTML simple, con lo cual, la mayoría de las graficas que podemos ver en la imagen superior, no tendrían datos para este protocolo. Solo los desplegables marcados en azul tienen datos resultantes de nuestro testeo.

Aun mencionado lo anterior, tenemos muchas gráficas disponibles, entre las que se encuentran las siguientes:

* Vusers ejecutandose
* Resumen de los Vusers
* Estadísticas de errores
* Errores por segundo
* Errores totales por segundo
* Promedio del tiempo de respuesta de la operación.
* Operaciones por segundo
* Operaciones totales por segundo
* Resumen de las operaciones
* Resumen del rendimiento de la operación
* Tiempo de respuesta de la operación bajo carga
* Tiempo de respuesta de la operación por ubicación.
* Peticiones por segundo
* Rendimiento
* Conexiones
* Conexiones por segundo
* Diagnostico de la página Web

Este reporte puede ser exportado en formato HTML o en formato de reporte normal. El formato HTML es básicamente como una web que podemos ver de forma local con todos los gráficos que hayamos añadido.

El reporte normal puede tener cualquiera de las siguientes extensiones:



En el anexo de este documento, se añade también el reporte exportado en formato PDF para su posterior revisión.

# Comparación de las dos implementaciones

Se trata de dar valores a los criterios de comparación definidos en el apartado 3 sobre la implementación de cada uno de los prototipos.

## Evaluación de los criterios en la implementación usando WebLoad

Debe incluir al menos una tabla con la siguiente estructura.

| **CRITERIO** | **EVALUACIÓN** |
| --- | --- |
| Criterio 1 |  |
| Criterio 2 |  |
| Criterio 3 |  |
| Criterio 4 |  |
| Criterio 5 |  |
| Criterio 6 |  |
| Criterio 7 |  |
| Criterio 8 |  |
| Criterio 9 |  |
| Criterio 10 |  |
| Criterio 11 |  |

Y algunos comentarios aclaratorios sobre aquellos criterios cuyo valor indicado en la tabla no sea suficiente para entenderlo.

## Evaluación de los criterios en la implementación usando LoadRunner

Debe incluir al menos una tabla con la siguiente estructura.

| **CRITERIO** | **EVALUACIÓN** |
| --- | --- |
| Criterio 1 | 10 |
| Criterio 2 | Ingles ( aunque cuenta con paquete de idiomas para traducirlo) |
| Criterio 3 | 8 |
| Criterio 4 | 7 |
| Criterio 5 | 8 |
| Criterio 6 | Alto |
| Criterio 7 | 5 |
| Criterio 8 | 10 |
| Criterio 9 | Ningún requisito previo. |
| Criterio 10 | 10 |
| Criterio 11 | 1 hora |

Y algunos comentarios aclaratorios sobre aquellos criterios cuyo valor indicado en la tabla no sea suficiente para entenderlo.

# Comparación de la implementación de las tecnologías

Debe incluir al menos una tabla resumen, en sección de página horizontal, cruzando los criterios y los valores de cada tecnología. Con una columna de comentarios sobre la comparación

| **CRITERIOS** | **WebLoad** | **LoadRunner** | **COMENTARIOS** |
| --- | --- | --- | --- |
| Criterio 1 |  |  |  |
| Criterio 2 |  |  |  |
| Criterio 3 |  |  |  |
| Criterio 4 |  |  |  |
| Criterio 5 |  |  |  |
| Criterio 6 |  |  |  |
| Criterio 7 |  |  |  |
| Criterio 8 |  |  |  |
| Criterio 9 |  |  |  |
| Criterio 10 |  |  |  |
| Criterio 11 |  |  |  |

# Conclusiones

A partir de la información incluida en el apartado 7 y de la experiencia al realizar el trabajo, el grupo debe estar en condiciones de manifestar su opinión sobre la implementación del sistema utilizando ambas tecnologías, y debe plasmarla en este apartado, indicando las ventajas e inconvenientes más relevantes de utilizar una u otra tecnología para implementar el sistema.

---------------------------

(Hay que cumplir la estructura básica indicada de secciones. Pero si se desea se pueden añadir otras secciones como anexos. Por ejemplo, alguna encuesta de opinión realizada sobre las tecnologías, etc.)