

Zephyr & Ranorex



Jaime Llorente Pedrajas (Coordinador)

Pablo Javier Ramírez Martínez

Álvaro Sanz Alfaro

Alejandro Eugenio Tena Escolar

José Ignacio Vadillo Carrasco

TG3

Contenido

[1. Autores del trabajo, planificación y entrega 3](#_Toc8078157)

[1.1 Autores 3](#_Toc8078158)

[1.2 Planificación 3](#_Toc8078159)

[1.3 Entrega 3](#_Toc8078160)

[2. Requisitos del prototipo a implementar 3](#_Toc8078161)

[2.1 Requisitos funcionales 3](#_Toc8078162)

[2.2 Otros requisitos 4](#_Toc8078163)

[3. Criterios de comparación en la implementación 4](#_Toc8078164)

[3.1 Criterio 1: Tiempos 4](#_Toc8078165)

[3.2 Criterio 2: Recursos 5](#_Toc8078166)

[3.3 Criterio 3: Funcionalidad 6](#_Toc8078167)

[3.4 Criterio 4: Diseño 6](#_Toc8078168)

[4. Proyecto de implementación de un prototipo del sistema utilizando la herramienta Ranorex 7](#_Toc8078169)

[4.1 Documentación de diseño 7](#_Toc8078170)

[4.2 Documentación de construcción 7](#_Toc8078171)

[4.3 Documentación de pruebas 8](#_Toc8078172)

[4.4 Documentación de instalación 10](#_Toc8078173)

[4.5 Manual de usuario 10](#_Toc8078174)

[5. Proyecto de implementación de un prototipo del sistema utilizando la herramienta Zephyr 12](#_Toc8078175)

[5.1 Documentación de diseño 12](#_Toc8078176)

[5.2 Documentación de construcción 13](#_Toc8078177)

[5.3 Documentación de pruebas 14](#_Toc8078178)

[5.4 Documentación de instalación 14](#_Toc8078179)

[5.5 Manual de usuario 16](#_Toc8078180)

[6. Comparación de las dos implementaciones 20](#_Toc8078181)

[6.1 Evaluación de los criterios en la implementación usando la tecnología Ranorex 20](#_Toc8078182)

[6.2 Evaluación de los criterios en la implementación usando la tecnología Zephyr 21](#_Toc8078183)

[7. Comparación de la implementación de las tecnologías 22](#_Toc8078184)

[8. Conclusiones 1](#_Toc8078185)

# 1. Autores del trabajo, planificación y entrega

## 1.1 Autores

El grupo T1 está formado por:

* Jaime Llorente Pedrajas (Coordinador)
* Pablo Javier Ramírez Martínez
* Álvaro Sanz Alfaro
* Alejandro Eugenio Tena Escolar
* José Ignacio Vadillo Carrasco

Nos encargaremos de comparar las tecnologías Zephyr y Ranorex.

## 1.2 Planificación

Al estar el grupo formado por 5 personas y al ser un 30% el peso de la asignatura se requiere un total de 45h de cada alumno, lo que eso hace 225 horas en total.

El reparto de las horas ha sido de manera equitativa, en la que se ha diferenciado por una parte el trabajo del coordinador del grupo como el del resto de los integrantes del grupo. Por ello cada uno de los integrantes se ha especializado en una tecnología para así la información estuviese orientada al mismo entorno.

En este diagrama de Gantt se puede ver cómo hemos organizado el trabajo.

[TG3 - Diagrama de Gantt](https://app.teamweek.com/#pp/jhYUp7saUnjteotVjooYVQKO6AzPFhR0)

## 1.3 Entrega

Incluimos un enlace en el que aparece nuestro repositorio en GitHub dónde incorporamos nuestro trabajo y sus archivos.

[TG3 - GitHub](https://github.com/jaimellorente/TG3)

En dicho repositorio se encuentra la carpeta TG3 dónde se encuentran los siguientes archivos:

* Trabajo terminado: del trabajo terminado con el nombre **TG3\_final.docx**
* Presentación del trabajo: **TG3\_final.pptx**
* Prototipos obtenidos implementando cada una de las tecnologías (deben incluir el código fuente y todos los archivos necesarios para la instalación y uso de cada prototipo):
  + **PrototipoTecnologiaA\_final.zip (o .rar)**
  + **PrototipoTecnologiaB\_final.zip (o .rar).**

# 2. Requisitos del prototipo a implementar

## 2.1 Requisitos funcionales

| **REQ.** | **DESCRIPCIÓN** |
| --- | --- |
| RF01 | Registro de usuarios mediante correo electrónico y contraseña. |
| RF02 | Confirmación de contraseña. |
| RF03 | Uso de un Captcha. |
| RF04 | Acceso al sistema mediante un usuario registrado. |
| RF05 | Cierre de sesión. |
| RF06 | Acceso al perfil personal. |
| RF07 | Modificación del perfil personal. |
| RF08 | Acceso a un apartado de ayuda. |
| RF09 | Acceso a página de contacto. |
| RF10 | Chat de soporte. |
| RF11 | Selección de idioma de la web |
| RF12 | Soló uso en Windows |
| RF13 | Errores de diseño y desarrollo |
| RF14 | Facilidad de inclusión de archivos remotos como Jira |
| RF15 | Recursos ocultos. |

## 2.2 Otros requisitos

| **REQ.** | **DESCRIPCIÓN** |
| --- | --- |
| R01 | Manejo sencillo. El camino que siga el usuario dentro del sitio debe ser rápido e intuitivo |
| R02 | Orden visual. se necesita orden visual para que el usuario lo pueda comprender |
| R03 | Requisitos de datos. Solo datos visibles |

# 3. Criterios de comparación en la implementación

## 3.1 Criterio 1: Tiempos

**3.1.1: Tempo de instalación de la tecnología.**

*Nombre del criterio:* Instalación de tecnología

*Descripción:* Tiempo transcurrido en la instalación del software

*Tipo de valor:* Numérico (minutos)

**3.1.2: Tiempo de planificación**

*Nombre del criterio:* Tiempo de planificación

*Descripción:* Horas empleadas en la planificación inicial antes de implementar el software.

*Tipo de valor:* Numérico (horas)

**3.1.3: Obtención de conocimientos**

*Nombre del criterio:* Obtención de conocimientos

*Descripción:* Horas empleadas en la búsqueda de información para así poder entender el funcionamiento de la tecnología antes de usarla.

*Tipo de valor: Numérico (horas)*

**3.1.4: Tiempo arranque**

*Nombre del criterio:* Tiempo de arranque de tecnologías*.*

*Descripción:* Tiempo que tarda la tecnología en ponerse en funcionamiento desde que se ejecuta*.*

*Tipo de valor:* Numérico (segundos).

## 3.2 Criterio 2: Recursos

**3.2.1: Recursos de CPU**

*Nombre del criterio:* Recursos necesarios de CPU*.*

*Descripción:* Consumo de recursos de CPU por parte de la tecnología.

*Tipo de valor:* Texto libre.

**3.2.2: Recursos de Memoria RAM**

*Nombre del criterio:* Recursos necesarios de Memoria RAM.

*Descripción*: Consumo de recursos de Memoria RAM por parte de la tecnología.

*Tipo de valor:* Numérico (KiloBytes).

**3.2.3: Almacenamiento**

*Nombre del criterio:* Almacenamiento necesario

*Descripción:* Cantidad que ocupa en el disco dichas tecnologías.

*Tipo de valor:* Numérico (MegaBytes).

## 3.3 Criterio 3: Funcionalidad

**3.3.1: Facilidad de uso**

*Nombre del criterio: Facilidad de uso*

*Descripción:*

*Tipo de valor: Texto libre*

**3.3.2: Calidad del soporte**

*Nombre del criterio: Calidad del soporte*

*Descripción: Facilidad para contactar con el soporte y buena respuesta por su parte.*

*Tipo de valor: Numérico (del 1 al 10)*

**3.3.3: Documentación**

*Nombre del criterio:* Documentación de soporte

*Descripción:* Calidad de la documentación que nos ofrece la tecnología para el uso del programa.

*Tipo de valor:* Numérico (del 1 al 10)

**3.2.4: Seguridad**

*Nombre del criterio:* Seguridad

*Descripción:* Ingreso al programa mediante usuario y contraseña

*Tipo de valor:* booleano (sí / no)

## 3.4 Criterio 4: Diseño

**3.3.1: Control de usuario**

*Nombre del criterio:* Control de usuario

*Descripción:* El usuario puede moverse de ventana a ventana y hacer cualquier cosa que desee.

*Tipo de valor:* booleano (sí / no)

**3.3.2: Control de Claridad**

*Nombre del criterio*: Claridad

*Descripción:* La información presentada debe ser fácil de comprender y su debe ser visualmente sencillo.

*Tipo de valor:* booleano (sí / no)

**3.3.3 Criterio de control de Personalización**

*Nombre del criterio: Personalización*

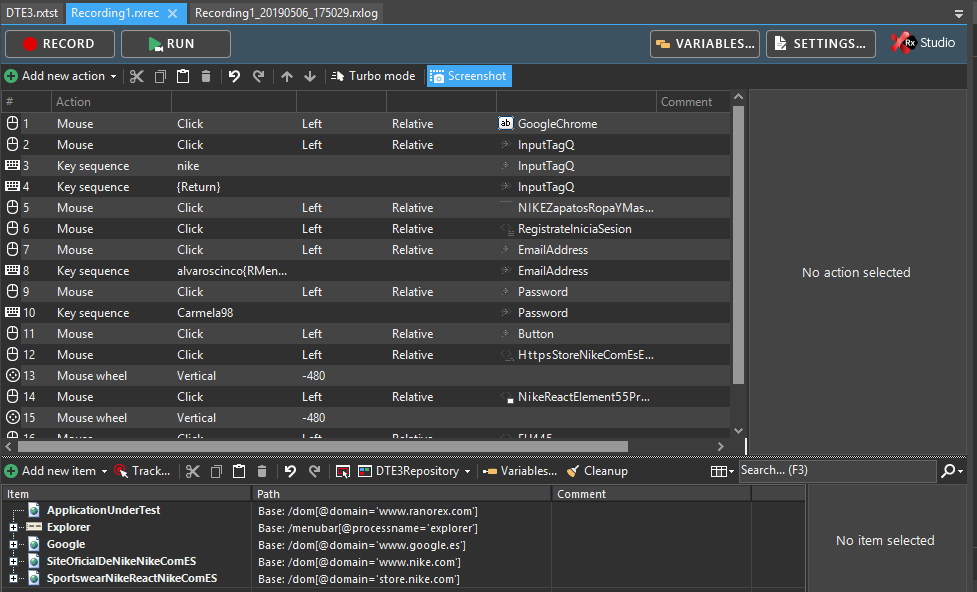
*Descripción:* Valorar si es posible personalizar las ventanas de nuestra tecnología a gusto del usuario.

*Tipo de valor: booleano (sí / no)*

# 4. Proyecto de implementación de un prototipo del sistema utilizando la herramienta Ranorex

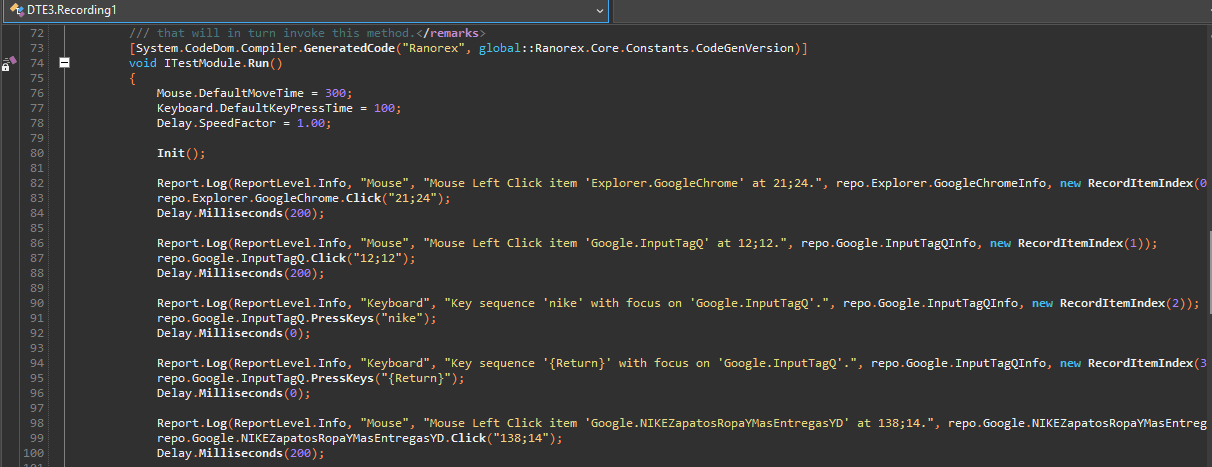
## 4.1 Documentación de diseño

Hemos realizado el diseño del prototipo añadiendo acciones y elementos al proyecto.

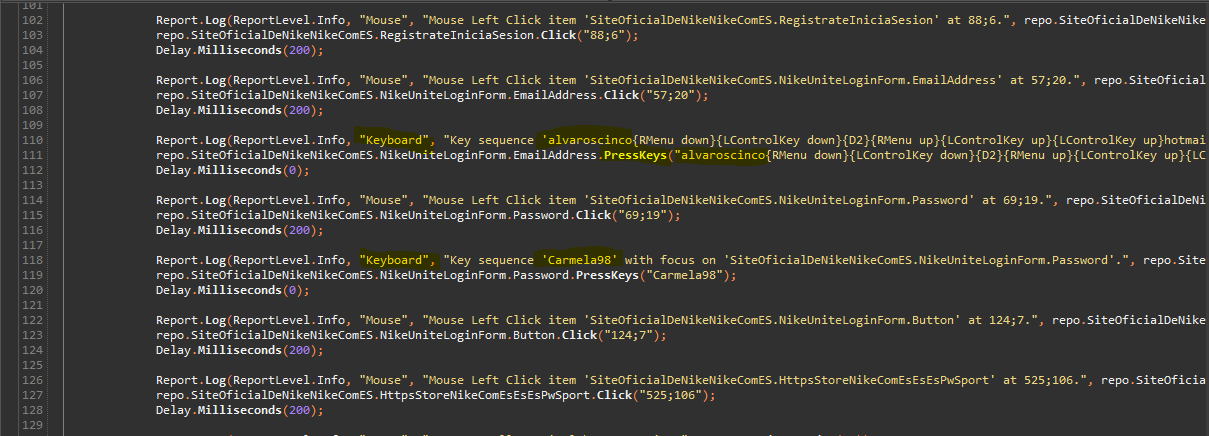


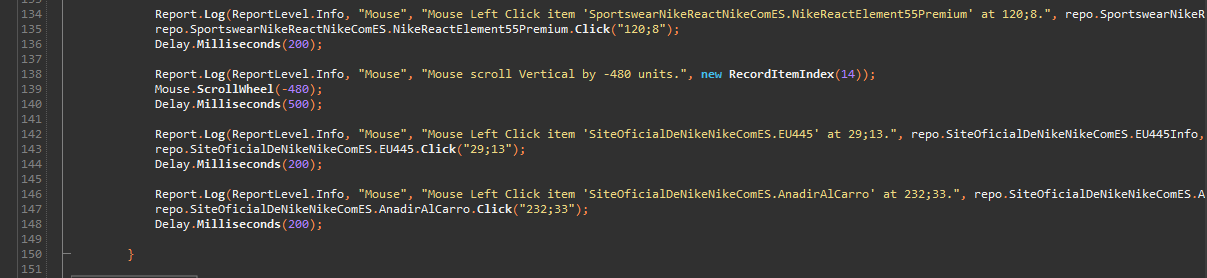
## 4.2 Documentación de construcción

Hemos creado un prototipo en el que se hace automáticamente una entrada a Nike y una compra de unas zapatillas entrando con un nombre de usuario y una password. En todos los pasos del recording anterior, si se hace clic se puede ver la captura del paso por el que va el prototipo automático. En las capturas de a continuación encontramos los datos del ratón tiempo de la ruleta y coordenadas y otras acciones como meter el usuario (key sequence).



Ranorex además capta las contraseñas, para poder realizar el test entero, como vemos en la siguiente captura.

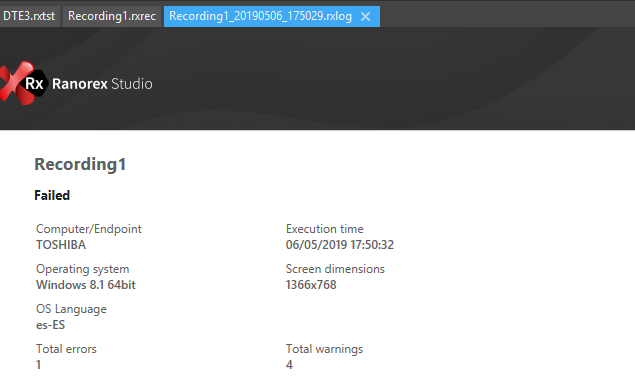


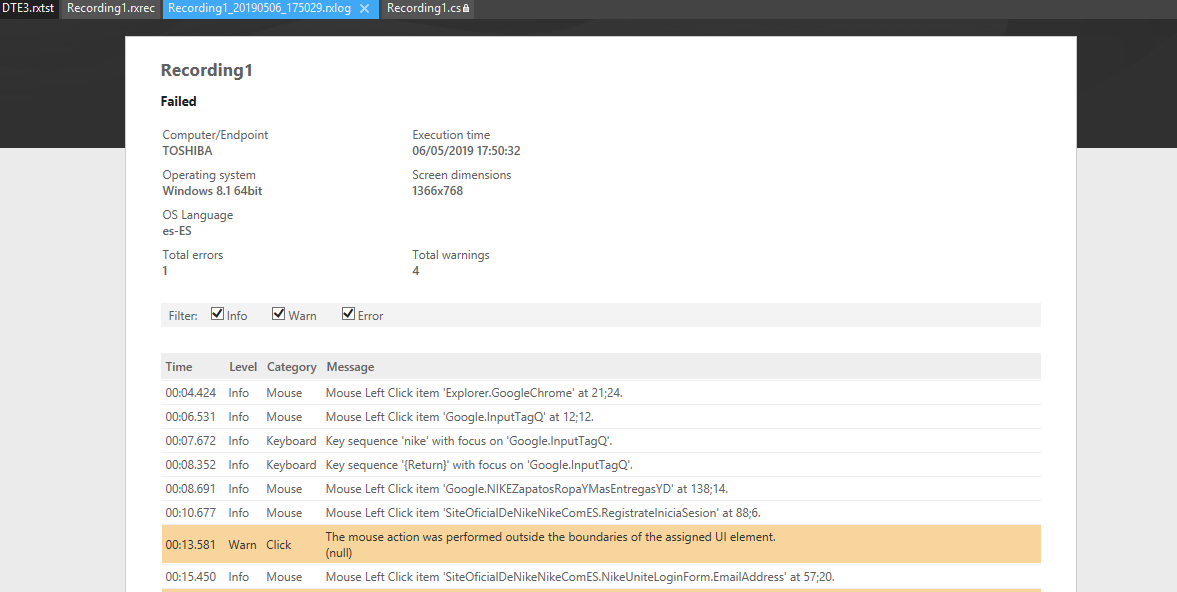


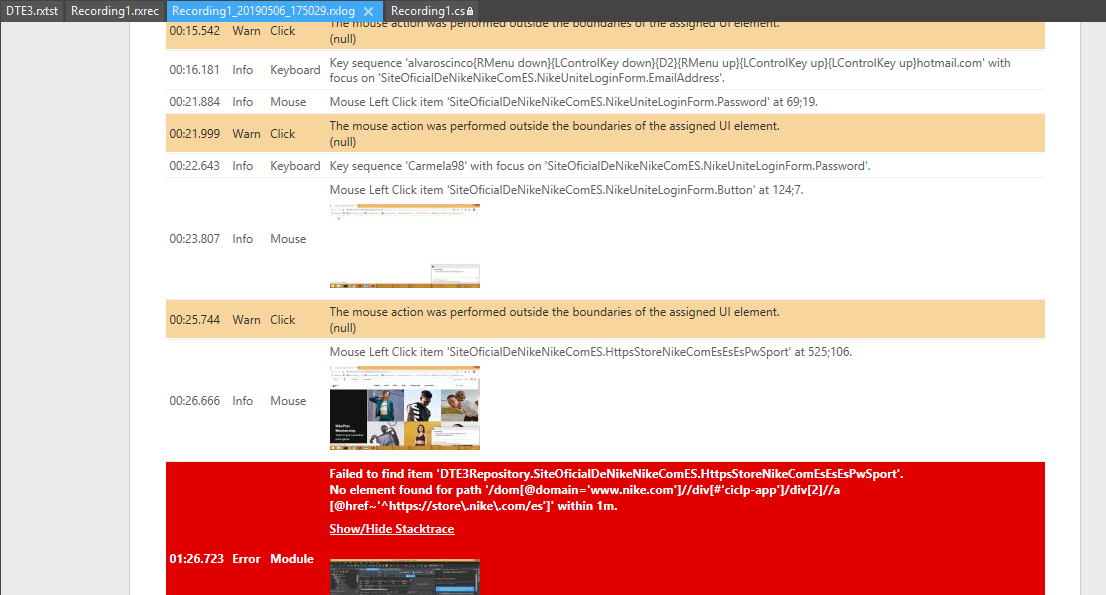
## 4.3 Documentación de pruebas

La prueba que hemos realizado es comprar unas zapatillas en Nike, hemos podido observar que Ranorex no tiene bien calibrado el ratón. A veces cuando teóricamente tiene que llegar a un enlace para hacer click, no llega porque no está bien calibrado (podemos comprobar lo que tiene que hacer en las capturas de cada acción del recording), y salta un error por el cuál no continua la prueba.

Quizás sea necesario tener Chrome en la barra de tareas para iniciar la prueba

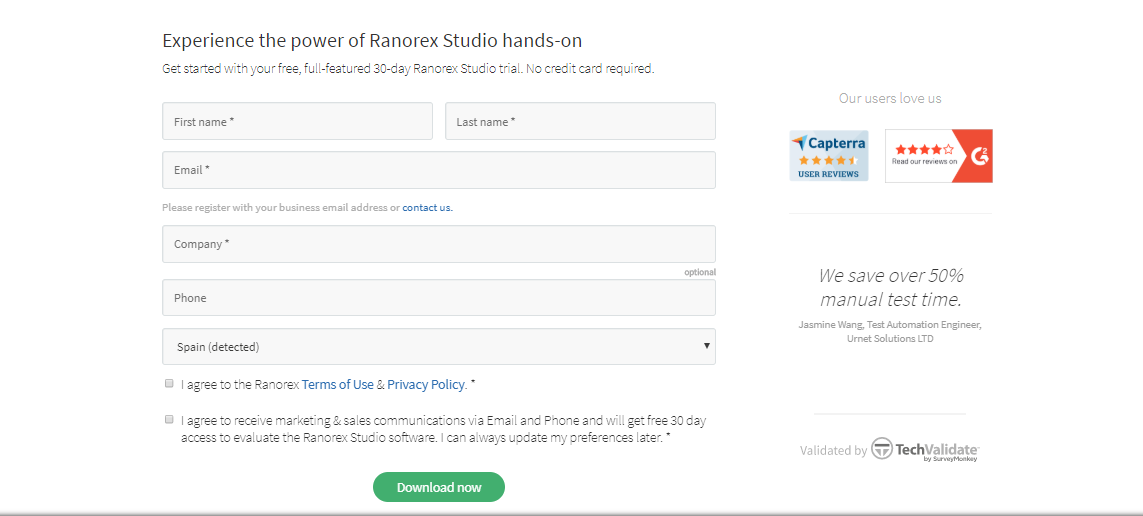


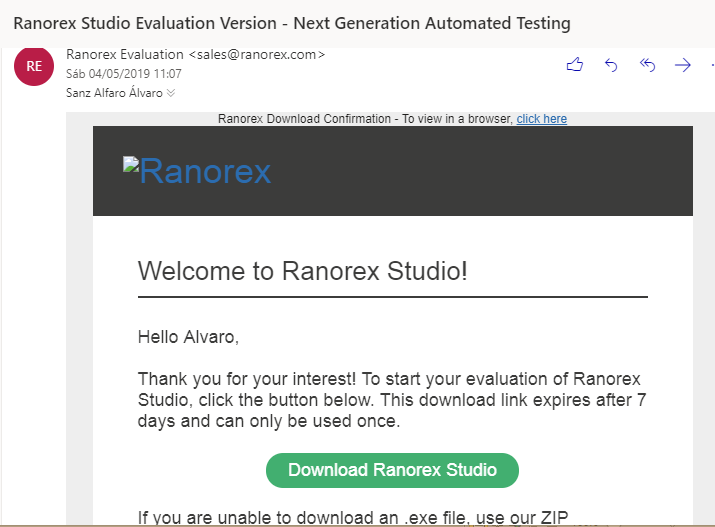




## 4.4 Documentación de instalación

Cuando entramos en la página oficial de ranorex y pulsamos en download free trial, nos aparece esta capturan en la que solo hay que rellenar los campos. Cuando la rellenamos nos mandarán al correo un mensaje al correo, donde nos proporcionan el programa.

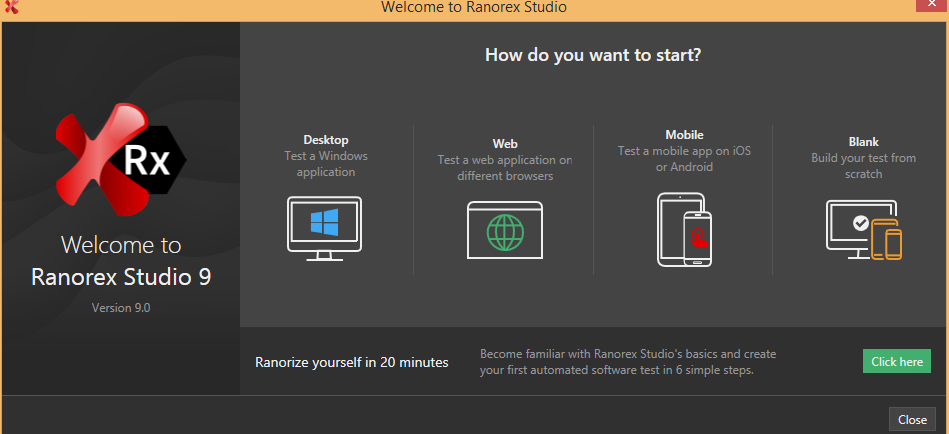




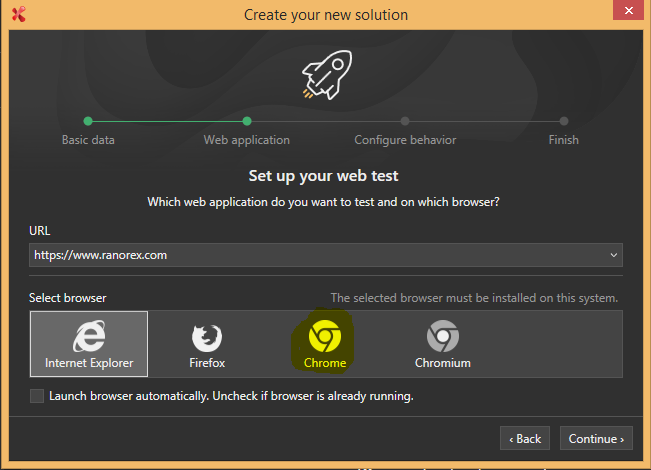
## 4.5 Manual de usuario

Para comenzar, debemos seleccionar new solution via wizard si no hemos comenzado el proyecto u open solution, si ya lo hemos realizado.

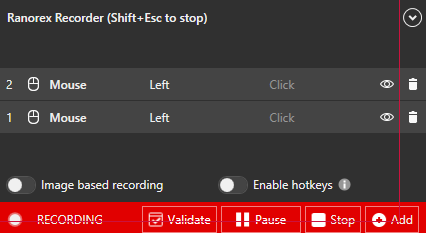
Después debemos seleccionar a que queremos realizar la prueba. En nuestro caso es una página web.



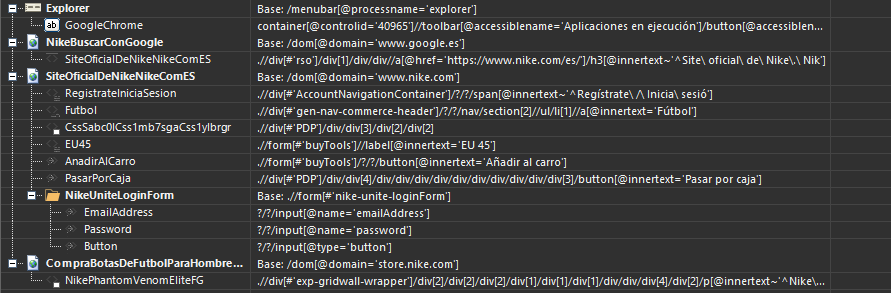
Más tarde tienes la opción de seleccionar el navegador que desees, el nuestro caso Chrome.



Una vez que se nos ha creado el proyecto podemos comenzar a diseñar las pruebas automáticas. Simplemente tenemos que pulsar record y comenzar a pulsar, escribir o deslizar el ratón o su rueda para ver cómo se desplaza. En la siguiente captura podemos ver cómo funciona el record. En cuanto pulsamos stop, pasa directamente al Ranorex, a recording action.



Ranorex pone en las partes de las acciones cada movimiento que se realiza con el record activado, incluyendo capturas y elementos de ruta donde tiene que acceder. Si no accede a esa ruta y accede a otra también saltará un error. En la siguiente captura podemos ver las rutas



El usuario puede grabar con el record tanto como quiera que dure la prueba automática y tantas funcionalidades como desee. Como ya hemos comentado previamente es probable que al darle a run salte algún error de que no calibre bien el ratón y por tanto no encuentre la ruta a la que tiene que ir.

# 5. Proyecto de implementación de un prototipo del sistema utilizando la herramienta Zephyr

## 5.1 Documentación de diseño

Para el uso de Zephyr es necesario instalar el complemento desde el propio Jira por lo tanto esta es la página principal de Jira:

Imagen que contiene captura de pantalla, monitor

Descripción generada automáticamente

## 5.2 Documentación de construcción

La construcción se basa en la creación de test o pruebas como también se llaman. Al crear pruebas elegimos el tipo de prueba ya sea una nueva implementación como un soporte o incluso un error encontrado que queremos resolver. Estas pruebas se clasifican a su vez por un nivel de prioridad para saber cuales hay que solucionar, realizar o dar soporte primero.

Se adjunta un ejemplo de pruebas creadas por nosotros que serán detalladas posteriormente:

Imagen que contiene captura de pantalla

Descripción generada automáticamente

## 5.3 Documentación de pruebas

Durante la creación de los test o pruebas, hemos tenido bastantes problemas ya que, al ser Zephyr un determinado complemento de Jira, había que especificar que esas pruebas eran del tipo Zephyr para luego poder ser sometidas a las herramientas de testing que incluía Zephyr.

## 5.4 Documentación de instalación

Lo primero que debemos hacer es entrar en la pagina oficial de Jira y obtener la versión de Jira paquete de software para poder trabajar con la interfaz.

Imagen que contiene captura de pantalla

Descripción generada automáticamente

Una vez hemos accedido a la interfaz, procedemos a dirigirnos a la pagina oficial de Zephyr donde al intentar descargar una versión de prueba detectará nuestra cuenta en Jira y nos redirigirá a la pagina web de Jira donde podremos instalar el complemento.

Imagen que contiene captura de pantalla

Descripción generada automáticamente

## 5.5 Manual de usuario

Lo primero que debemos hacer es crear un proyecto en la página principal de Jira.

Imagen que contiene captura de pantalla

Descripción generada automáticamente

Entramos dentro de nuestro proyecto y creamos una nueva prueba.

Imagen que contiene captura de pantalla

Descripción generada automáticamente

En esta prueba indicaremos primero el tipo de incidencia y luego un resumen de que es lo que sucede o lo que se quiere implementar. A parte, podremos elegir el nivel de importancia de ella.

Imagen que contiene captura de pantalla

Descripción generada automáticamenteImagen que contiene captura de pantalla

Descripción generada automáticamente

Cuando ya hemos creado la incidencia podemos observar en el apartado de incidencias un listado con cada una de ellas y si pinchamos en una de ellas podremos modificar si ha sido empezada si esta pendiente o si ya esta realizada. En el caso de que aun no este asignada a ningún miembro podremos auto asignárnosla.

Imagen que contiene captura de pantalla

Descripción generada automáticamente

Una vez tengamos todas las incidencias creadas podremos hacer un testing a través de Zephyr del conjunto de incidencias de todo un proyecto. De esta forma, podemos ver comparativas de las incidencias creadas y realizadas, de las que son errores y las que son nuevas funcionalidades, etc.

Imagen que contiene captura de pantalla

Descripción generada automáticamente

A parte, encontraremos muchísimas funcionalidades para ver el porcentaje de tareas de cada tipo o resueltas para poder hacernos una idea de la cantidad de trabajo que queda por realizar o categorizar por el tipo o la prioridad.

Imagen que contiene captura de pantalla

Descripción generada automáticamente

# 6. 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.

## 6.1 Evaluación de los criterios en la implementación usando la tecnología Ranorex

Debe incluir al menos una tabla con la siguiente estructura.

| **CRITERIO** | **EVALUACIÓN** |
| --- | --- |
| Tiempo de instalación de la tecnología | Una instalación bastante sencilla que lleva unos 10 minutos entre que te registras y te llega la confirmación al correo |
| Tiempo de planificación | Creación de los requisitos y planificar las pruebas a realizar. 3h |
| Obtención de conocimientos | Video Tutoriales de Youtube y tutoriales de los cursos gratis que pusimos en el TG1. 5h |
| Tiempo de arranque | Ranorex es bastante rápido, en 5 segundos ya esta ejecutándose |
| Recursos de CPU | https://i.gyazo.com/26aefbbcb9b75b7ff7bf4c8810c45c58.png |
| Recursos de Memoria RAM | https://i.gyazo.com/cd625a550b46e26556f1b17f051f9ad7.png |
| Almacenamiento | https://i.gyazo.com/684fde883a55159cf21040461417df91.png |
| Facilidad de uso | Ranorex es bastante intuitivo una vez sabes cómo funciona |
| Calidad del soporte | Nos intentamos poner en contacto con Ranorex por el chat pero no recibimos atención por parte de ellos |
| Documentación de soporte | Bastante buena para saber como manejar el programa e implementar tests |
| Seguridad | Solo a la hora de descargarlo, una vez lo descargas ya no tienes que registrarte |
| Control de Claridad | Información fácil de comprender con varios tipos de diseño de interfaz adecuándose a tus gustos |
| Personalización | Sí, es posible mover las ventanas de las funcionalidades de Ranorex |

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

## 6.2 Evaluación de los criterios en la implementación usando la tecnología Zephyr

| **CRITERIO** | **EVALUACIÓN** |
| --- | --- |
| Tiempo de instalación de la tecnología | Una extensión que se añade al Jira y es instantáneo |
| Tiempo de planificación | Creación de los requisitos y planificar las pruebas a realizar. 3h |
| Obtención de conocimientos | Video Tutoriales de Youtube y tutoriales de los cursos gratis que pusimos en el TG1. 5h |
| Tiempo de arranque | Como hemos optado por una solución en la nube cuando inicias sesión en Jira ya estás trabajando con Zephyr |
| Recursos de CPU | https://i.gyazo.com/8baaf40cbcb4c7a4f8d94079bf887d65.png |
| Recursos de Memoria RAM | https://i.gyazo.com/143e418c82b69c19b039c9fcf71f27f4.png |
| Almacenamiento | Hemos optado por una solución en la nube mas intuitiva por lo que no ocupa espacio de almacenamiento físico |
| Facilidad de uso | El Zephyr es algo más complejo a la hora de realizar las pruebas |
| Calidad del soporte | Nada más entrar en la página de Zephyr se puso en contacto con nosotros un consultor por si teníamos dudas con Zephyr y le preguntamos varias dudas y nos las soluciono muy rápido |
| Documentación de soporte | En su página web hay varios vídeos para realizar las pruebas |
| Seguridad | Cada vez que se entra hay que iniciar sesión |
| Control de Claridad | Es bastante claro y dinámico |
| Personalización | No es muy personalizable porque no podemos mover las herramientas dentro de la aplicación |

# 7. 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** | **RANOREX** | **ZEPHYR** | **COMENTARIOS** |
| --- | --- | --- | --- |
| Tiempo de instalación de la tecnología | Te registras con varios datos personales y para empezar a descargarlo hay que confirmar desde el correo | Instantánea porque es una extensión donde metes datos personales básicos y ya lo tienes listo | Es más práctico la instalación de Zephyr para el usuario pero es más seguro Ranorex para que metas un correo falso porque hay que confirmar el registro en el correo para poder descargarlo |
| Tiempo de planificación | Creación de los requisitos y planificar las pruebas a realizar. 3h | Creación de los requisitos y planificar las pruebas a realizar. 3h | El tiempo de planificación antes de instalar las tecnologías ha sido similar |
| Obtención de conocimientos | Video Tutoriales de Youtube y tutoriales de los cursos gratis que pusimos en el TG1. 5h | Video Tutoriales de Youtube y tutoriales de los cursos gratis que pusimos en el TG1. 5h | Para saber como implementar los tests y pruebas de cada tecnología hemos tenido que invertir unas horas informándonos de cómo funciona la tecnología |
| Tiempo de arranque | Al ser una aplicación tarda más | Solución en la nube que al registrarte en Jira ya tienes acceso | Zephyr es más rápido y en este apartado es mejor opción |
| Recursos de CPU | CPU : 15% | CPU: 24% | Medido con el Monitor de Recursos Ranorex utiliza menos recursos de CPU |
| Recursos de Memoria RAM | Memoria RAM usada: 43% | Memoria RAM usada: 56% | Medido con el Monitor de Recursos Ranorex utiliza menos memoria RAM |
| Almacenamiento | Al ser una aplicación ocupa 384Mb en el disco | Al ser solución en la nube no ocupa espacio | Realmente que Zephyr sea en la nube no es una ventaja muy destacable en cuanto a almacenamiento porque Ranorex solo ocupa 384Mb |
| Facilidad de uso | Ranorex tiene muchas herramientas pero una vez sabes como funciona tiene muchas más funcionalidades y es más intuitivo | Zephyr visualmente es más sencillo pero nos ha costado más realizar la pruebas y nos ha surgido algún problema | Preferimos el Ranorex porque aunque visualmente sea más complejo luego es más intuitivo y tiene más funcionalidades |
| Calidad del soporte | Por parte de atención al cliente no recibimos respuesta | Respuesta inmediata por parte de atención al cliente | Zephyr muy buena calidad del soporte ya que nos resolvieron las dudas que teníamos sobre como implementar algunas funcionalidades |
| Documentación de soporte | En la página web buenas indicaciones escritas y con imágenes para resolver posibles dudas | Básicamente muchos vídeos en los que va a costar más encontrar la información que deseas | Ranorex nos da una mejor documentación de soporte con imágenes muy aclarativas y con índices para saber donde tienes que ir directamente según tu duda |
| Seguridad | Como es una aplicación te la descargas y ya puedes acceder siempre a ella | Cada vez que entras tienes que registrarte en Jira para acceder a Zephyr | Nos parecen buenas opciones las dos maneras ya que una es aplicación y la otra esta dentro de Jira en Internet |
| Control de Claridad | Información fácil de comprender con varios tipos de diseño de interfaz adecuándose a tus gustos | Es bastante claro y dinámico | La claridad va según gustos por eso preferimos Ranorex ya que puedes poner cada accesorio del programa en cualquier lado de la pantalla para que esté a tu gusto |
| Personalización | Es posible mover las ventanas de las funcionalidades de Ranorex | No es muy personalizable porque no podemos mover las herramientas dentro de la aplicación | Ranorex es mucho más personalizable y se adecua más al usuario |

# 8. Conclusiones

Al usar las dos tecnologías podemos concluir diciendo que son dos tecnologías que, aunque sean usadas para testing son complementarias. Ranorex, es una herramienta software con la cual podemos observar fallos en distintos programas dinámicamente mientras los ejecutamos. Zephyr con Jira, es una forma de poder tener al tanto a todos los personales que se dedica a la realización de un proyecto, de las incidencias que van surgiendo al igual que de su estado y de la persona al cargo de ella.

Hemos tenido algunos problemas a la hora de guardar incidencias en la herramienta de Zephyr, aunque, podemos asegurar que es una excelente herramienta que está preparada para equipos con un gran número de trabajadores. Brinda una gran oportunidad de tener al alcance de tu mano en cada momento el progreso de un determinado proyecto.

Es por ello, que consideramos que Ranorex es una herramienta que nos puede ser más útil debido a las prestaciones que nos brinda en comparación a Zephyr.