

Zephyr & Ranorex



Jaime Llorente Pedrajas (Coordinador)

Pablo Javier Ramírez Martínez

Álvaro Sanz Alfaro

Alejandro Eugenio Tena Escolar

José Ignacio Vadillo Carrasco

TG1

Contenido

[1. Autores del trabajo, planificación y entrega 3](#_Toc5652477)

[1.1 Autores 3](#_Toc5652478)

[1.2 Planificación 3](#_Toc5652479)

[1.3 Entrega 3](#_Toc5652480)

[2. Descripción de las tecnologías 3](#_Toc5652481)

[2.1 Descripción Zephyr 3](#_Toc5652482)

[2.2 Descripción Ranorex 4](#_Toc5652483)

[3. Criterios de comparación 5](#_Toc5652484)

[3.1 Categoría A: Información general 5](#_Toc5652485)

[3.1.1 Criterio A.1: Autor de la herramienta 5](#_Toc5652486)

[3.1.2 Criterio A.2: Sede de la empresa 5](#_Toc5652487)

[3.1.3 Criterio A.3: Fecha de creación de la herramienta 5](#_Toc5652488)

[3.1.4 Criterio A.4 Versión gratuita del software 5](#_Toc5652489)

[3.1.5 Criterio A.5 Implementación de la herramienta. 6](#_Toc5652490)

[3.1.6 Criterio A.6 Lenguaje de la herramienta 6](#_Toc5652491)

[3.1.7 Criterio A.8: Flexibilidad de idiomas. 6](#_Toc5652492)

[3.1.8 Criterio A.8: Tiempo de instalación. 6](#_Toc5652493)

[3.2 Categoría B: Características principales de las herramientas 6](#_Toc5652494)

[3.2.1 Criterio B.1: Test realizados por la herramienta 6](#_Toc5652495)

[3.2.2 Criterio B.2: Interfaz gráfica de pruebas 6](#_Toc5652496)

[3.2.3 Criterio B.3: Tipo de código 6](#_Toc5652497)

[3.2.4 Criterio B.4: Soporte técnico 7](#_Toc5652498)

[3.3 Categoría C: Mantenimiento y Formación 7](#_Toc5652499)

[3.3.1 Criterio C.1: Cursos de aprendizaje 7](#_Toc5652500)

[3.3.2 Criterio C.2: Asistencia a los cursos 7](#_Toc5652501)

[3.3.3 Criterio C.3: Mantenimiento 7](#_Toc5652502)

[3.3.4 Criterio C.4: Cantidad de información disponible 7](#_Toc5652503)

[3.4 Categoría D: Hardware 8](#_Toc5652504)

[3.4.1 Criterio D.1: Memoria RAM 8](#_Toc5652505)

[3.4.2 Criterio D.2: Almacenamiento externo 8](#_Toc5652506)

[3.5 Categoría E: Software 8](#_Toc5652507)

[3.5.1 Criterio E.1: Sistema Operativo 8](#_Toc5652508)

[3.5.2 Criterio E.2: Lenguajes de programación disponibles 8](#_Toc5652509)

[4. Evaluación de los criterios por tecnología 8](#_Toc5652510)

[4.1 Evaluación de los criterios para Zephyr 8](#_Toc5652511)

[4.2 Evaluación de los criterios para Ranorex 9](#_Toc5652512)

[5. Comparación de tecnologías 10](#_Toc5652513)

[6. Recomendaciones 14](#_Toc5652514)

[6.1 Situación 1 14](#_Toc5652515)

[6.1.1 Descripción de la situación 14](#_Toc5652516)

[6.1.2 Recomendación de tecnología a utilizar 14](#_Toc5652517)

[6.2 Situación 2 16](#_Toc5652518)

[6.2.1 Descripción de la situación 16](#_Toc5652519)

[6.2.2 Recomendación de tecnología a utilizar 17](#_Toc5652520)

[7. Conclusiones 17](#_Toc5652521)

# 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 10% el peso de la asignatura se requiere un total de 15h de cada alumno, lo que eso hace 75 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.

[TG2 - Diagrama de Gantt](https://app.teamweek.com/#pp/1ok4UB5xBP2GR2RVMuBtqMVfRsqhjZhJ)

## 1.3 Entrega

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

[TG2 - GitHub](https://github.com/jaimellorente/TG2)

En dicho repositorio se encuentra la carpeta TG1 dónde se encuentran los siguientes archivos:

* Trabajo terminado: del trabajo terminado con el nombre **TG2\_final.docx**
* Presentación del trabajo: **TG2\_final.pptx**

Hemos creado la carpeta TG2 mirando al futuro ya que luego tendremos que crear TG3 en el mismo repositorio.

# 2. Descripción de las tecnologías

Cuando hablamos de Automation Testing Tools decimos que nos encontramos ante una prueba manual que es realizado por un humano sentado frente a un ordenador, ejecutando así cuidadosamente los pasos de la prueba. Y por ello las pruebas de automatización significa utilizar una herramienta de automatización para ejecutar su conjunto de casos de prueba.

## 2.1 Descripción Zephyr

**Zephyr** es la herramienta de gestión de pruebas número 1, que proporciona soluciones de extremo a extremo para equipos ágiles de todos los tamaños. Con esta herramienta se obtiene la flexibilidad, visibilidad y la información que necesita para lanzar un mejor software más rápido.

Por lo tanto, aquella empresa que cuente con JiRA para administrar sus requerimientos y otro tipo de documentación, y que ha estado usando a esta herramienta para seguir el Testing a través de defectos como issues, ahora puede probar usar al Zephyr como una herramienta propia para Testing, y q ha sudo desarrollada para responder a las exigencias propias de todo proyecto de Testing.

Zephyr ofrece:

* Crear, ver, editar y clonar pruebas.
* Enlazar a stories, tareas, requerimientos, etc.
* Integración Agile a los boards de testing.
* Planificar los ciclos de ejecución de pruebas.
* Ejecutar pruebas.
* Enlazar defectos.
* Navegación focalizada en el proyecto que se está trabajando.
* Crear reportes con matrices de trazabilidad.
* Seguir métricas de calidad.
* Un poderoso buscador de pruebas / defectos.

La presentación de las opciones de menú propias para la gestión de cualquier proyecto de Testing son muy fáciles de comprender, son intuitivas y este aspecto es muy importante a la hora de evaluar la curva de aprendizaje por la que debe pasar el área de Testing.

## 2.2 Descripción Ranorex

**Ranorex** la usan más de 14000 usuarios en todo el mundo y aceleran las pruebas con esta herramienta. Es un todo en uno para el tema de automatización de pruebas. Es una herramienta fácil para los que nunca la han usado ya que cuenta con una interfaz.

**Definición:** Ranorex es una poderosa herramienta para la automatización de pruebas. Es un marco de automatización de prueba GUI utilizado para la prueba de aplicaciones web, de escritorio y móviles. Ranorex no tiene su propio lenguaje de scripting para automatizar la aplicación. Utiliza lenguajes de programación estándar como VB.NET y C #.

**Descripción**: Ranorex puede automatizar cualquier escritorio, aplicación basada en web o software móvil. Es compatible con muchas tecnologías como Silverlight, .NET, Winforms, Java, SAP, WPF, HTML5, Flash, Flex, Windows Apps (Native / Hybrid), y iOS, Android.

Ranorex tiene un buen registro, reproducción y edición de acciones realizadas por el usuario con el editor de captura / reproducción basado en objetos, lo que significa que el grabador Ranorex ofrece un enfoque simple para crear pasos de prueba automatizados para aplicaciones web, móviles (nativas y WAP) y de escritorio.

Los pasos bien diseñados se pueden reutilizar en múltiples casos de prueba utilizando el registrador de pruebas Ranorex. Ranorex tiene un buen informe de ejecución de pruebas de interfaz de usuario, que se puede utilizar para reproducir errores y mantener pruebas. Los archivos de informes de prueba de Ranorex están basados ​​en XML.

Los informes de ejecución se generan durante la ejecución de la prueba. Reduce el tiempo necesario para determinar qué y dónde salieron mal las cosas. El informe también incluye capturas de pantalla de los casos de prueba fallidos.

La herramienta Ranorex admite los siguientes tipos de pruebas de automatización:

* Pruebas en varios navegadores: Ranorex puede probar y validar aplicaciones web en navegadores populares como Chrome, IE, Safari y Firefox.
* Pruebas de regresión: Ranorex puede usarse para pruebas de regresión en entornos de compilación continua para encontrar nuevos errores de software mucho más rápido.

# 3. Criterios de comparación

## 3.1 Categoría A: Información general

### 3.1.1 Criterio A.1: Autor de la herramienta

*Nombre del criterio: Autor.*

*Descripción: Nombre de la persona, institución o empresa que ha creado la herramienta.*

*Tipo de valor: Texto libre*

### 3.1.2 Criterio A.2: Sede de la empresa

*Nombre del criterio: Sede*

*Descripción: nombre del país, universidad o ciudad en la que tiene la sede principal la empresa.*

*Tipo de valor: Texto libre*

### 3.1.3 Criterio A.3: Fecha de creación de la herramienta

*Nombre del criterio: fecha creación*

*Descripción: año en el cuál se creó la herramienta.*

*Tipo de valor: Fecha*

### 3.1.4 Criterio A.4 Versión gratuita del software

*Nombre del criterio: versión gratuita*

*Descripción: dictaminar si existe una prueba gratuita y si es así bajo que requisitos*

*Tipo de valor: BOOLEANO(SI/NO)*

### 3.1.5 Criterio A.5 Implementación de la herramienta.

*Nombre del criterio: implementación*

*Descripción: constatar en que SO o en que superficie se puede implementar la herramienta, incluso si puede implementarse en la nube directamente.*

*Tipo de valor: Texto libre*

### 3.1.6 Criterio A.6 Lenguaje de la herramienta

*Nombre del criterio: Lenguaje usado*

*Descripción: lenguaje de programación en el que está implementada la herramienta.*

*Tipo de valor: Texto libre*

### 3.1.7 Criterio A.8: Flexibilidad de idiomas.

*Nombre del criterio: Flexibilidad de idiomas*

*Descripción: Idiomas en los que esta disponible esta herramienta.*

*Tipo de valor: Texto libre*

### 3.1.8 Criterio A.8: Tiempo de instalación.

*Nombre del criterio: Tiempo de instalación*

*Descripción: Tiempo que tarda en instalarse la tecnología hasta poder empezar a trabajar.*

*Tipo de valor: Numérico*

## 

## 3.2 Categoría B: Características principales de las herramientas

### 3.2.1 Criterio B.1: Test realizados por la herramienta

*Nombre del criterio: test*

*Descripción: Explicación de que tests realiza cada herramienta*

*Tipo de valor: Texto libre*

### 3.2.2 Criterio B.2: Interfaz gráfica de pruebas

*Nombre del criterio: Interfaz*

*Descripción: en este apartado dictaminamos que interfaz es más sencilla o más intuitiva*

*Tipo de valor: Texto libre*

### 3.2.3 Criterio B.3: Tipo de código

*Nombre del criterio: Tipo de código*

*Descripción: En este requisito comentamos la política de creación de actualizaciones de cada empresa si es un código cerrado o si es open source en el que hay una comunidad que realiza mejoras*

*Tipo de valor: Texto libre*

### 3.2.4 Criterio B.4: Soporte técnico

*Nombre: soporte técnico*

*Descripción: en este apartado comentamos la calidad del soporte técnico para ayudas sobre ambas herramientas*

*Tipo de valor: Texto libre*

## 3.3 Categoría C: Mantenimiento y Formación

### 3.3.1 Criterio C.1: Cursos de aprendizaje

*Nombre del criterio: Cursos de formación*

*Descripción: en este apartado comentamos que tipo de cursos tiene disponibles cada herramienta para enseñar a nuevos usuarios.*

*Tipo de valor: Texto libre*

### 3.3.2 Criterio C.2: Asistencia a los cursos

*Nombre del criterio: Asistencia*

*Descripción: es simplemente la disponibilidad de asistencia a los cursos.*

*Tipo de valor: Texto libre*

### 3.3.3 Criterio C.3: Mantenimiento

*Nombre del criterio: Mantenimiento del producto*

*Descripción: Si la empresa ofrece un mantenimiento del producto al adquirir la tecnología.*

*Tipo de valor: Booleano (SI/NO)*

### 3.3.4 Criterio C.4: Cantidad de información disponible

*Nombre del criterio: Cantidad de información*

*Descripción: Indicación sobre la cantidad de información cuantificable disponemos.*

*Tipo de valor: Texto libre*

## 

## 3.4 Categoría D: Hardware

### 3.4.1 Criterio D.1: Memoria RAM

*Nombre del criterio: Memoria RAM*

*Descripción: Cantidad de memoria necesaria para implementar la solución.*

*Tipo de valor: Numérico (GB)*

### 3.4.2 Criterio D.2: Almacenamiento externo

*Nombre del criterio: Espacio de almacenamiento externo*

*Descripción: Espacio mínimo necesario para implementar la tecnología.*

*Tipo de valor: Numérico (GB)*

## 3.5 Categoría E: Software

### 3.5.1 Criterio E.1: Sistema Operativo

*Nombre del criterio: Sistema operativo*

*Descripción: Sistemas operativos validos para utilizar la tecnología.*

*Tipo de valor: Texto libre*

### 3.5.2 Criterio E.2: Lenguajes de programación disponibles

*Nombre del criterio: Lenguaje de programación*

*Descripción: Necesidad de programación para instalar la tecnología y en caso de ser necesario, hay que indicar que lenguaje de programación debemos usar.*

*Tipo de valor: Texto libre*

# 4. Evaluación de los criterios por tecnología

## 4.1 Evaluación de los criterios para Zephyr

Debe incluir al menos una tabla con la siguiente estructura.

|  |  |
| --- | --- |
| CRITERIOS | EVALUACIÓN |
| Criterio A.1: Autor de la herramienta | Wind River System, pero un año después paso a ser un proyecto de Linux. |
| Criterio A.2: Sede principal | San José, USA |
| Criterio A.3: Fecha | 2016 |
| Criterio A.4 : Versión gratuita | NO, pero prueba 30 días |
| Criterio A.5: Implementación | Opción de implementación en la nube como un software como servicio (Saass) y de instalarlo en el ordenador físico en Windows. |
| Criterio A.6: Lenguaje usado | Lenguaje C y assembly |
| Criterio A.7: Flexibilidad de idiomas | Por defecto inglés. |
| Criterio A.8: Tiempo de instalación |  |
| Criterio B.1: Test | especializada en test de seguridad y basados en requisitos, pero no realiza test parametrizados |
| Criterio B.2: Interfaz gráfica | Intuitiva |
| Criterio B.3: Tipo código | Zephyr tiene una gran comunidad de usuarios con un código abierto |
| Criterio B.4: Soporte técnico | Es código abierto y pese a que tiene ayudas de Linux del tipo man, no tiene soporte como tal. En la actualidad están trabajando en la creación de un buen soporte |
| Criterio C.1: Cursos aprendizaje | Cursos en persona (más específicos), en directo en línea, seminarios web y documentación o tutoriales. |
| Criterio C.2: Asistencia a cursos | Da asistencia en línea para cualquier problema con los cursos monitorizados |
| Criterio C.3: Mantenimiento | Si |
| Criterio C.4: Cantidad de información disponible | Menor cantidad de la deseada para poder usar y entender esta tecnología. |
| Criterio D.1: Memoria RAM | Mínimo 8GB |
| Criterio D.2: Almacenamiento externo | 100GB |
| Criterio E.1: Sistema operativo | Linux, Windows y MAC. |
| Criterio E.2: Lenguajes de programación disponibles | C#. |

\*\* Criterio B.3: Tipo de código: Zephyr de hecho tiene en su página el número de commits y autores que trabajan en la herramienta mediante github. <https://www.zephyrproject.org/>



## 4.2 Evaluación de los criterios para Ranorex

|  |  |
| --- | --- |
| CRITERIOS | EVALUACIÓN |
| Criterio A.1: Autor de la herramienta | Jenö Herget y su hijo Gabo Herget |
| Criterio A.2: Sede principal | Una ciudad de Austria que se llama Gratz. |
| Criterio A.3: Fecha | Inicio en 2004, implementación en 2006 |
| Criterio A.4 : Versión gratuita | NO, pero prueba 30 días |
| Criterio A.5: Implementación | No hay opción en la nube, se implementa en Windows. |
| Criterio A.6: Lenguaje usado | Lenguaje C y VB.NET |
| Criterio A.7: Flexibilidad de idiomas | Francés, inglés y alemán |
| Criterio A.8: Tiempo de instalación |  |
| Criterio B.1: Test | Tipo de valor: Ranorex está especializada en test basados en requisitos y test parametrizados, pero no implementa tests de seguridad |
| Criterio B.2: Interfaz gráfica | Intuitiva y muy amigable |
| Criterio B.3: Tipo código | Es código cerrado, aunque puede cambiar algunos métodos en la librería central |
| Criterio B.4: Soporte técnico | Ranorex tiene un soporte técnico bastante bueno según los usuarios |
| Criterio C.1: Cursos aprendizaje | Cursos en persona (más específicos), en directo en línea, seminarios web y documentación o tutoriales. |
| Criterio C.2: Asistencia a cursos | Da asistencia en línea para cualquier problema con los cursos monitorizados, además, sirve algunos cursos más específicos que pueden ser en persona con un trato mucho más directo |
| Criterio C.3: Mantenimiento | SI |
| Criterio C.4: Cantidad de información disponible | Existe gran cantidad de información para poder usar y entender esta tecnología. |
| Criterio D.1: Memoria RAM | No aplica |
| Criterio D.2: Almacenamiento externo | No aplica |
| Criterio E.1: Sistema operativo | Windows, MAC y Linux. |
| Criterio E.2: Lenguajes de programación disponibles | C#, Java o Delphi. |

# 5. Comparación de tecnologías



* Gran cantidad de competencia en el mercado.
* Zephyr participa en eventos de pruebas de software conocidos mundialmente como el de EuroStar o Las Vegas.
* Código abierto.
* Se encuentra dentro de Atlassian Platinum Top Vendor.
* Dispone de dos packs de prueba.
* Fácil implementación y uso.
* Una de las versiones (alojamiento en servidor) permanece para siempre, aunque dejes de pagar licencia.
* Atención al cliente de clase mundial.
* Es considerado uno de los mejores programas de prueba de softtware

**OPORTUNIDADES**

**DEBILIDADES**

**AMENAZAS**

* No es multilenguaje (solo inglés).
* Es difícil encontrar su precio.

**FORTALEZAS**



Análisis DAFO Zephyr



* Gran competencia software en el mercado.
* Disponibilidad de varios paquetes con distintos precios competitivos.
* Herramienta usada por multinacionales conocidas como Siemens o TomTom.
* Posibilidad de ser asesorado por un socio de Ranorex
* No es multilenguaje (solo inglés).
* Dispone de pack de prueba sin tener que añadir tarjeta.
* Posibilidad de uso en distintos entornos (escritorio, web, móvil).
* Fácil para principiantes y potente para expertos.
* Soporta las principales tecnologías.
* Soporte técnico y comunidad de usuarios cualificada.

**OPORTUNIDADES**

**DEBILIDADES**

**AMENAZAS**

**FORTALEZAS**



Análisis DAFO Ranorex

|  |  |  |  |
| --- | --- | --- | --- |
| CRITERIOS | Zephyr | Ranorex | COMENTARIOS |
| A.1: Madurez y Versiones | Gran cantidad de versiones | Gran cantidad de versiones | En ambos casos existe una gran variedad de versiones y se van actualizando regularmente. |
| A.2: Coste | Sin especificar | Varios costes | Ranorex ofrece un precio competitivo mientras que Zephyr no muestra el precio del producto en su web. |
| A.3: Flexibilidad de idiomas | Inglés | Inglés | Ambos están desarrollados en inglés como idioma por defecto. |
| A.4: Instalación | Sencillo | Sencillo | *Fácil instalación en cualquiera de los dos casos* |
| A.5: Popularidad | Elevada | Muy elevada | Los dos son programas con mucho uso. |
| B.1: Garantía | Alta | Alta | En este caso en las dos herramientas se preocupan de actualizar regularmente las versiones para garantizar el buen uso y funcionamiento de estas. |
| B.2: Soporte | Sí | Sí | Las dos herramientas disponen de soporte de incidencias. |
| C.1: Interfaz Intuitiva | Sí | Sí | Las dos herramientas tienen un diseño de la interfaz bastante intuitiva. |
| C.2: Dificultad de uso | Media | Baja | En el caso de Zephyr encontramos algo de dificultad al usarla, pero en el caso de Ranorex encontramos un software intuitivo con una guía de usuario para aquellas personas que se inicien en ese software. |
| C.3: Integración | Alta | Alta | Las dos herramientas disponen de una gran capacidad de integración con otras herramientas. |
| C.4: Calidad del informe | Alta | Alta | Las dos calidades del informe son destacadas en el mercado. |

# 6. Recomendaciones

Deben platearse posibles situaciones de uso, y recomendar justificadamente una u otra tecnología en función de la situación. Al menos 2 situaciones diferentes.

## 6.1 Situación 1

### 6.1.1 Descripción de la situación

Tenemos una empresa de tamaño grande de 500 personas y se necesita una herramienta de pruebas de automatización. Necesitamos una herramienta que no sea básica como la mayoría del mercado porque no tienen todas las funcionalidades que necesitamos y el precio no es un inconveniente si la herramienta hace su función. Se necesitan varios requisitos por parte de esta herramienta.

* Que sea compatible con varios lenguajes de programación como C o Java
* Admitir ejecución en paralelo
* Tests basados en los requisitos y que estén parametrizados
* Licencias de distintos usos dependiendo del usuario de la empresa
* Soporte alto por parte de la empresa de la herramienta

### 6.1.2 Recomendación de tecnología a utilizar

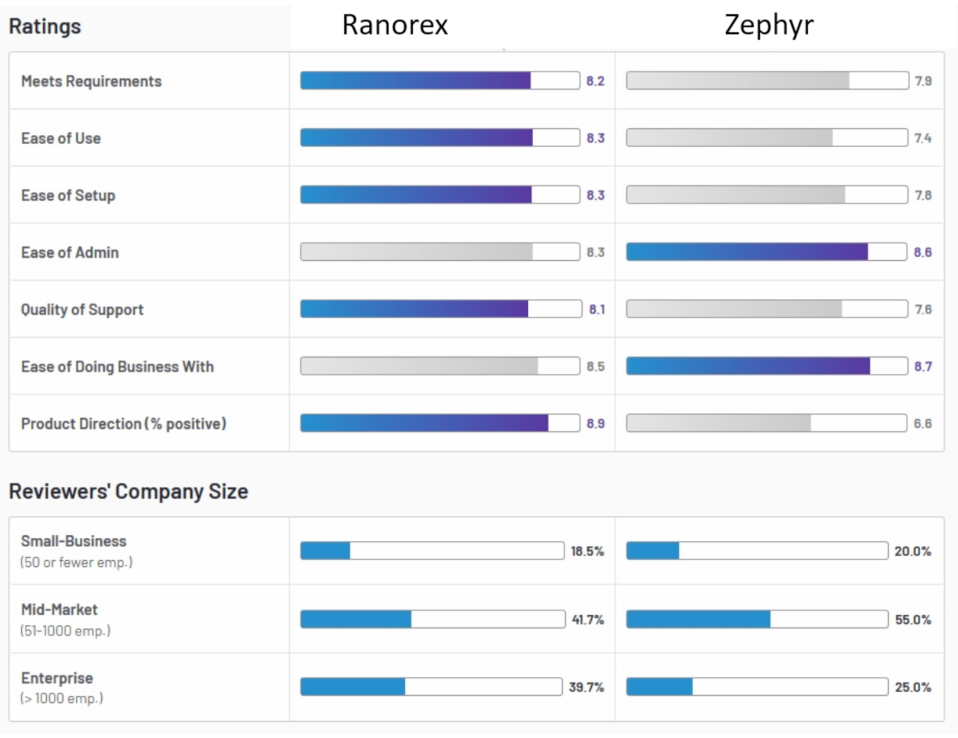
Debe indicarse la tecnología propuesta para esa situación.

Debe incluirse una tabla como la siguiente, mostrando las ventajas, respecto a los criterios, que ofrece cada tecnología en esa situación concreta.

Incluir sólo los criterios sobre los que se aprecien ventajas de una de las tecnologías frente a otra. No incluir criterios que no sean relevantes para la decisión

|  |  |
| --- | --- |
| Criterios relevantes para la decisión | Ventajas **Ranorex** |
| Criterio B.3: Tipo código | Nos interesa que sea código cerrado para una mayor compatibilidad con el hardware y para poder desarrollar más recursos en su desarrollo y mayor estabilidad |
| Criterio B.1: Test | Ranorex esta especializado en el tipo de tests basados en requisitos y parametrizados |
| Criterio E.2: Lenguajes de programación disponibles | Compatible con Java y C |
| Criterio C.4: Cantidad de información disponible | Soporte técnico bastante bueno y amplio según la comunidad |

Ranorex es la elegida para esta empresa



Alrededor del 80% de las medianas y grandes empresas que han utilizado estas herramientas han valorado su uso y en 5 de los 7 parámetros de la votación gana Ranorex por lo que se puede ver para que público va enfocada esta herramienta.

## 6.2 Situación 2

### 6.2.1 Descripción de la situación

Tenemos una empresa de tamaño pequeño que tiene 20 empleados y que la principal herramienta como tracker de issues es en este caso JIRA que permite que muchos otros complementos se integren a ella para potenciar su funcionalidad. No tenemos un gran capital por lo que antes de comprar una herramienta tendríamos que tener pruebas gratis y luego no un precio muy alto por parte de la herramienta Necesitamos un complemento que sea capaz de:

- Permitir gestionar el Testing de manera integral

- Dar lo opción de instalar parte de las cosas en la nube ya que el sistema informático de la empresa es bastante escaso

- Registrar los primeros y principales datos del proyecto de Testing

- Opción de tests de seguridad basados en requisitos

-Registrar las acciones de nuestra ejecución y de las incidencias que identifiquemos.

- Generar diversos tipos de reportes con toda la información recopilada

### 6.2.2 Recomendación de tecnología a utilizar

|  |  |
| --- | --- |
| Criterios relevantes para la decisión | Ventajas **Zephyr** |
| Criterio B.3: Tipo código | Código abierto con una gran comunidad de usuarios para ayudarte a integrarlo a tu empresa |
| Criterio A.4: Versión gratuita | No hay versión gratuita pero antes de comprar la versión definitiva se puede probar durante 30 días y luego su precio es bastante asequible comparado con Ranorex |
| Criterio B.1: Test | Zephyr está especializada en test de seguridad y basados en requisitos |
| Criterio A.5: Implementación | Zephyr nos da la opción de implementarlo en la nube |

# 7. Conclusiones

Ambas plataformas son geniales para la realización de diferentes proyectos, son asequibles y cada una tiene sus puntos más fuertes y sus puntos más débiles, lo cual no significa que una sea mejor que otra a la hora de hacer algo con ellas. Y si se está buscando aprender sobre Automation testing tools, cada una enseña algo diferente.

Para llevar a cabo un informe muy completo y exhaustivo podemos emplear las dos herramientas ya que podríamos decir que son complementarias.

Son dos herramientas muy útiles en su campo que se pueden ajustar muy bien a nuestras necesidades en cuanto a testear automáticamente información, estas herramientas nos ahorraran mucho tiempo de trabajo.

Cada una aporta un conocimiento diferente. Una vez sepamos que queremos hacer decidiremos que herramienta es más adecuada para el caso.