

Plantilla para documento de especificación de requisitos de software

Nombre del proyecto: *Sistema de Trivias - Temática: Software - Tecnología*

Fecha: 18/09/2023

Versión:

Creado por: *Salomone Genaro - Herrera Cristian*

Historial de revisiones

| Versión | Autor | Descripción de la versión | Fecha de realización |
|---------|-------|---------------------------|----------------------|
| | | | |

Historial de revisiones

| Aprobador | Versión aprobada | Firma | Fecha |
|-----------|------------------|-------|-------|
| | | | |
| | | | |
| | | | |
| | | | |

Historial de aprobaciones

| Revisor | Versión revisada | Firma | Fecha |
|---------|------------------|-------|-------|
| | | | |
| | | | |
| | | | |
| | | | |

Índice

1

Introducción

- 1.1 Alcance del producto
- 1.2 Valor del producto
- 1.3 Público objetivo
- 1.4 Uso previsto
- 1.5 Descripción general

2

Requisitos funcionales

3

Requisitos de la interfaz externa

- 3.1 Requisitos de la interfaz de usuarios
- 3.2 Requisitos de la interfaz de hardware
- 3.3 Requisitos de la interfaz de software
- 3.4 Requisitos de la interfaz de comunicación

4

Requisitos no funcionales

- 4.1 Seguridad
- 4.2 Capacidad
- 4.3 Compatibilidad
- 4.4 Confiabilidad
- 4.5 Escalabilidad
- 4.6 Mantenibilidad
- 4.7 Facilidad de uso
- 4.8 Otros requisitos no funcionales

5

Definiciones y acrónimos

1 Introducción

Describe el propósito del documento.

Describir de manera clara y concisa los beneficios, objetivos, metas, valor, público objetivo, uso previsto y características del software en desarrollo. Proporciona una visión general del producto, sus funcionalidades y requisitos.

1.1 Alcance del producto

Enumera los beneficios, los objetivos y las metas del producto.

Se propone desarrollar un sistema de trivias basado en software y tecnología. Los beneficios incluyen entretener a los usuarios, fomentar el conocimiento y mejorar la jugabilidad con un sistema de rankings. Los objetivos son crear trivias con preguntas relacionadas a la temática mencionada. Las metas son permitir el registro de usuarios, seleccionar la dificultad de las trivias, responder preguntas, ver resultados y tener un ranking de usuarios.

1.2 Valor del producto

Describe qué valor encontrará el público en tu producto.

El valor que esperamos encontrar en nuestros usuarios es: Entretenimiento, aprendizaje a través de las preguntas y respuestas, sana competencia mediante el ranking de usuarios.

1.3 Público objetivo

Describe a quién está destinado el producto.

El producto se realizó en base a la materia Análisis y Diseño de Sistemas de la carrera Analista en Computación de la Universidad Nacional de Río Cuarto.

1.4 Uso previsto

Describe de qué manera tu público objetivo usará el producto.

El producto final no se utiliza para la comercialización, por lo que no existe un público objetivo en general. Pero se puede tener una constancia del propio producto para un futuro destino.

1.5 Descripción general

Redacta un resumen de las funciones y características del software.

Los usuarios pueden registrarse, elegir la dificultad de la trivía, responder preguntas multiple choices, true-falses y autocompletadas con un tiempo límite de respuesta. Existe un sistema de ayuda para la trivía con dificultad principiante. Al finalizar la trivía se muestra un resultado y se otorgan puntos, se visualiza un sistema de rankings de usuarios con los puntos obtenidos en cada trivía.

2 Requisitos funcionales

Enumera los requisitos de diseño, los requisitos gráficos, los requisitos del sistema operativo y las limitaciones del producto.

- 1)** Gestión de usuario (Username/ Email/ Password).
 - 2)** Selección de dificultad (Principiante - Experto).
 - 3)** Diferentes tipos de preguntas (Choice - Autocompletado - True/False).
 - 4)** Después de una trivia, proporcionar información sobre si la respuesta fue correcta (o no), y además, mostrar la respuesta correcta.
 - 5)** Obtención de puntos al responder correctamente cada pregunta.
 - 6)** Si se elige dificultad principiante, se permitirá un sistema de ayudas/ pistas con información adicional de la pregunta.
 - 7)** Ranking de usuarios.
 - 8)** Timer para preguntas: cada pregunta tendrá un timer con un determinado tiempo para responder según su dificultad.
 - 9)** Sección de reclamos y valoración de la App. El usuario puede enviar un mensaje al desarrollador
 - 10)** Actualización a gestión de usuario. Ahora es posible autenticarse al sistema mediante Google Sign In.
 - 11)** Traducción de preguntas. El usuario tiene la posibilidad de elegir entre más de 100 lenguajes para usar el sistema.
-

3.1 Requisitos de la interfaz de usuario

Describe la lógica detrás de las interacciones entre los usuarios y el software (diseños de pantalla, guías de estilo, etc.).

Al ingresar a la aplicación, se muestran botones para iniciar sesión o registrarse, dando la posibilidad de autenticarse mediante Google. Una vez iniciada la sesión o el registro, se conduce al usuario a una página principal para elegir el modo principiante o experto de la trivia, también hay una sección de reclamos y otra para elegir un lenguaje para triviar. Una vez elegido alguno de los modos mencionados, estamos listos para responder la primer pregunta, donde una vez respondida, existe un botón "siguiente pregunta" para pasar a la pregunta siguiente. Si se elige modo principiante, además, existe un botón "¿Necesita una ayudita?" donde al darle clic, se despliega un mensaje de ayuda en relación a la pregunta.

3.2 Requisitos de la interfaz de hardware

Especifica los dispositivos compatibles con el software, los requisitos de red y los protocolos de comunicación que se utilizarán.

El producto realizado, es decir la aplicación, se usa de manera remota. Es decir, aquel que tiene el programa completo en su computadora, podrá levantar la app y visualizarla desde un host local. La propia app no se pensó para comercializarla, sino más bien para un proyecto académico.

3.3 Requisitos de la interfaz de software

Incluye las conexiones entre tu producto y otros componentes de software, como el marco de front-end y back-end, las bibliotecas, etc.

Front-end: Se utilizaron lenguajes como HTML, JavaScript y CSS.

Back-end: Se utilizó el lenguaje Ruby con un framework Sinatra.

Base de datos: Se utilizó SQLite.

3.4 Requisitos de la interfaz de comunicación

Enumera los requerimientos de los programas de comunicación que usará tu producto, como los emails o los formularios integrados.

Cuando un usuario se registra, debe ingresar su correo electrónico (gmail), pero no se envía ninguna información. El ranking de usuarios permite visualizar el puntaje de los usuarios que han realizado trivias, por lo tanto, se tiene un registro de aquellos que han jugado.

Se integra a la aplicación el sistema de Google Sign In mediante OAuth para validar el token de google si el usuario elige autenticación mediante google. También se hace uso de la API text-translator (RAPID-API) para petitionar las traducciones de las preguntas y sus respectivas respuestas si el usuario desea jugar en otro idioma.

4 Requisitos no funcionales

4.1 Seguridad

Incluye las normas de privacidad y protección de datos que deben cumplirse.

Se utiliza un método para la encriptación de contraseñas de usuarios. Esto impide que la persona que maneja la base de datos vea la contraseña.

4.2 Capacidad

Describe la demanda de almacenamiento actual y futura de tu software.

Como el producto de software fue realizado con fines académicos, no es necesario preocuparse por el almacenamiento que ocupa el producto, la escalabilidad, etc. Si se puede pensar en una refactorización a nivel de código del producto.

4.3 Compatibilidad

Enumera los requisitos mínimos de hardware para el software.

Para el procesador, cualquier tipo puede funcionar, con una frecuencia de 1.8 GHz o superior. La memoria RAM de al menos 2 GB. Para el espacio en disco se requiere un almacenamiento libre de 200 MB o superior.

4.4 Confiabilidad

Calcula cuál es el tiempo de fallo crítico esperado de tu producto con relación a un uso normal.

Para calcular el tiempo de fallo crítico esperado, se utilizan métricas y técnicas de análisis de confiabilidad, como el análisis de fallos, las pruebas de estrés y la estimación estadística. Estos métodos permiten identificar y medir los posibles puntos de falla en el sistema y evaluar la confiabilidad general del producto

4.5 Escalabilidad

Calcula la cantidad máxima de trabajo con la que tu software puede funcionar según lo esperado.

No existe una escalabilidad real, ya que los usuarios son los propios creadores, y aquellos que también prueben el producto para ver su funcionamiento. Se podría pensar en mejorar el producto de software para ser lanzado al mercado y que tenga usuarios reales.

4.6 Mantenibilidad

Describe cómo se debe usar la integración continua para implementar funciones o reparar errores rápidamente.

Utilización de pruebas unitarias (testing) para algunos modelos. Utilización de Git Hub para el trabajo en grupo y la evolución del mismo. Utilización de la herramienta Docker.

4.7 Facilidad de uso

Describe qué tan fácil resultará el uso del software a los usuarios finales.

El producto a nivel visual es sencillo y fácil de entenderlo. Las trivias con cualquier nivel de dificultad, son muy desafiantes y entretenidas para responder. Se debe contar con un mínimo conocimiento de software y tecnología en general y sino se va aprendiendo a medida que se realizan muchas trivias.

4.8 Otro

Enumera cualquier otro requisito no funcional.

5