

Especificación de requerimientos

IType

Aplicación Android para
pruebas de velocidad de tipeo

Versión 3.0

Presentado por:

Cancinos, José María
Giraudó, Juan Pablo
Oliva, Nahuel David

Fecha de la Versión: 2/05/2019

Índice

-	HISTORIAL DE VERSIONES	3
1	INTRODUCCIÓN	4
	1.1 Propósito del documento	4
	1.2 Convenciones del documento	4
	1.3 Destinatarios del documento	4
	1.4 Alcance del producto	4
	1.5 Referencias	5
2	DESCRIPCIÓN GENERAL	6
	2.1 Funcionamiento	6
	2.2 Perspectiva del producto	6
	2.3 Funciones del producto	7
	2.4 Clase y características del usuario	7
	2.5 Entorno operativo	8
	2.6 Límites de diseño e implementación	8
	2.7 Documentación de usuario	9
	2.8 Dependencias y suposiciones	9
3	REQUERIMIENTOS DE INTERFAZ EXTERNOS	10
	3.1 Interfaces de usuario	10
	3.2 Interfaces de hardware	10
	3.3 Interfaces de software	10
	3.4 Interfaces de comunicación	10
4	REQUERIMIENTOS FUNCIONALES	12
	4.1 REQ 1: Entrada del usuario	12
	4.2 REQ 2: Interfaz	13
	4.3 REQ 3: Sesión	14
	4.4 REQ 4: Funcionamiento general	15
5	CASOS DE USO	17
	5.1 Definición de casos de uso	17
	5.2 Diagrama de casos de uso	18
	5.3 Diagrama de actividades	19
	5.4 Matriz de trazabilidad	20
6	REQUERIMIENTOS NO FUNCIONALES	21
	6.1 Requerimientos de rendimiento	21
	6.2 Requerimientos de eficiencia	21
	6.3 Requerimientos de seguridad	21
	6.4 Atributos de calidad del software	21
7	CASOS DE PRUEBA DE SISTEMA	22
	7.1 Definición de casos de prueba de sistema	22
	7.2 Definición de casos de prueba alternativos	29
	7.3 Pruebas de aceptación	30
	7.4 Pruebas smoke	30
	7.5 Matriz de trazabilidad	31
8	DIAGRAMAS ADICIONALES	33
	8.1 Diagrama de secuencia para registro	33
	8.2 Diagrama de secuencia para juego	34

Historial de versiones

Nombre de la versión	Fecha	Número de versión
IType – Especificación de requerimientos	10/04/2019	1.0
IType – Especificación de requerimientos	21/04/2019	2.0
IType – Especificación de requerimientos	2/05/2019	3.0

1. Introducción

1.1 Propósito del documento

El propósito del siguiente documento es de presentar una descripción detallada sobre el software **IType**.

Se explicará el propósito, las características, las interfaces, la funcionalidad y los límites en que se deberá operar.

1.2 Convenciones del documento

Este documento ha sido creado en base a la planilla IEEE¹ para documentos de especificación de requerimientos.

1.3 Destinatarios del documento

Este documento está destinado a los usuarios y a potenciales desarrolladores.

1.4 Alcance del producto

IType es una aplicación destinada al sistema operativo Android, en la cual se pone a prueba la velocidad de “tipeo” del usuario. El objetivo es escribir la mayor cantidad de caracteres de manera correcta durante el tiempo de duración de la prueba.

Luego, al terminar la prueba, se da a conocer la velocidad de escritura que se obtuvo y si se logró calificar en la tabla de posiciones.

La aplicación, también, ofrece la posibilidad de escoger distintos niveles de dificultad, permitiendo así el uso tanto a usuarios expertos como novatos.

¹ Instituto de Ingeniería Eléctrica y Electrónica - Institute of Electrical and Electronics Engineers, en inglés. Plantilla de Documento de Requerimientos. https://web.cs.dal.ca/~hawkey/3130/srs_template-ieee.doc

1.5 Referencias

ELEMENTO	REFERENCIA O DIRECCIÓN WEB
Repositorio del proyecto en GitLab:	https://gitlab.com/Nadaol/typing_app
Planilla de SRS – Modelo IEEE	https://goo.gl/nsUFwy
Licencia GNU - Versión 3.0	https://opensource.org/licenses/GPL-3.0
Base de datos - MySQL	
Documentación de Java	https://docs.oracle.com/javase/7/docs/api/index-files/index-1.html

2. Descripción General

2.1 Funcionamiento

La aplicación IType constará de una interfaz con cuatro vistas interactivas:

- a) La vista de **inicio de sesión y registro** será cargada al iniciar la aplicación. Ésta permitirá al usuario iniciar y crear una cuenta. La cuenta se guardará en la base de datos para almacenar la información asociada a la misma.
- b) En la vista del **menú principal** se podrán elegir tres modos de dificultad para la prueba:
 - Fácil.
 - Intermedio.
 - Difícil.
- c) La vista **activa** es aquella en la que se corre la prueba en sí, luego de haber elegido la dificultad de la misma; mostrando la palabra a “tipear” y permitiendo la entrada al usuario.
- d) Finalmente, una vez que la prueba finaliza, se podrá cargar la vista de **estadísticas**. Ésta muestra la tabla con las diez mejores posiciones de acuerdo a la velocidad.

2.2 Perspectiva del producto

IType se origina como resultado del trabajo a presentar para la materia “Ingeniería de Software” de la carrera Ingeniería en Computación de la Universidad Nacional de Córdoba.

Fue pensada como una aplicación de pasatiempos cuyo objetivo es obtener la mayor cantidad de palabras correctas ingresadas mediante el teclado de un teléfono del sistema Android.

La aplicación es de tipo código libre (open source) bajo la licencia “GNU General Public License versión 3.0”

Características:

- Soporte de diversas versiones de Android.
- Cuenta de usuario: cada instancia de aplicación está asociada con una cuenta en el sistema para su funcionalidad.
- Número de usuarios: el sistema da soporte a una gran cantidad de usuarios.
- Publicación de la prueba: La información relacionada a la prueba se publicará en la base de datos si el usuario luego de terminarla califica dentro de los primeros 10 puestos.

2.3 Funciones del producto

La función principal de la aplicación es la generación aleatoria de palabras en idioma español. Éstas se generarán de acuerdo al nivel que el usuario haya seleccionado en la vista de menú principal.

El nivel de dificultad tendrá incidencia sobre 5 factores en el mecanismo de generación de palabras:

- Frecuencia de palabras más o menos habituales del idioma.
- Frecuencia de letras más o menos utilizadas del idioma.
- Repetición de letras, en una palabra.
- Repetición de palabras.
- Tamaño de la palabra

La aplicación mostrará una palabra a la vez. Ésta se actualizará a medida que el usuario las escribe individualmente.

La duración de cada prueba es de 30 segundos. Iniciando la cuenta desde el instante en que se elige la dificultad y finalizando al cumplir los 30 segundos.

Una vez terminada la prueba, el sistema calculará la velocidad de “tipeo” que se obtuvo mediante la cantidad total de caracteres escritos correctamente y comparará con la base de datos si es apto para ser calificado en un puesto determinado. Si es así, la aplicación determinará qué posición de tabla de puntajes es el correcto y se guardará la información en la base de datos MySQL para luego poder ser mostrada por la aplicación en la vista de estadísticas.

2.4 Clase y características del usuario

El usuario de la aplicación es aquel que posee conocimientos básicos sobre la operación de dispositivos móviles que cuentan con el sistema Android.

La edad del usuario se sitúa como mayor a 5 años.

Debe poseer habilidad lectora.

Debe tener un buen conocimiento sobre el idioma español, ya que la aplicación estará exclusivamente en este idioma.

No es necesario ningún nivel de educación avanzada para usar esta aplicación ya que la interacción está en escribir correctamente (copiar caracteres).

Los usuarios deben utilizar el teclado con distribución QWERTY en su disposición hispanohablante (uso del carácter Ñ).

2.5 Entorno operativo

La presente aplicación está basada en el sistema Android. Android es un entorno operativo de código abierto destinado a dispositivos móviles. En este caso, la aplicación se diseña para la versión “smartphone”, sólo para teléfonos.

Android corre sobre un kernel de sistema operativo Linux.

Las versiones de Android actualmente soportadas de manera oficial son:

Android 7: “Nougat”

Android 8: “Oreo”

Android 9: “Pie”

Android 10: “Android Q” (en desarrollo)

Los requerimientos mínimos especificados por Android² para el hardware son:

- RAM: 2 [GB]
- Almacenamiento interno: 16 [GB]
- Velocidad del procesador: 1.1 [GHz]
- Arquitectura de procesador: 32 bits.
- Protección con certificación IP: IP64
- Pantalla táctil

2.6 Límites de diseño e implementación

Este es un proyecto de código libre desarrollado exclusivamente para el sistema operativo Android.

- La aplicación debe ser compatible con la versión más reciente hasta la más antigua versión soportada por Android. (Hasta el momento: Android 7 a Android 9).
- Se deben utilizar herramientas de código abierto.
- Se deben seguir los lineamientos Material Design de estilo dados por el fabricante del sistema Android para cuestiones de visualización y estética de la aplicación en conjunción con la estética del sistema global.
- Uso de lenguaje Java para el desarrollo del código.
- Uso de base de datos mediante MySQL.
- La base de datos contendrá cuatro tablas: una para la sesión de usuarios (usuario, contraseña) y las otras tres para los puestos con sus dificultades respectivas (usuario, velocidad).
- Uso de un sistema de manejo de datos para el control de la base de datos.
- El proceso de desarrollo se gestionará por medio de una determinada herramienta web para gestionar el proceso de desarrollo de software.

² Para Android 7 “Nougat”. Android. <https://www.android.com/enterprise/recommended/requirements/>

- Las pruebas de software se deberán gestionar con una herramienta de gestión de software para testing.
- Uso de sistema de manejo de datos apropiado para el control de la base de datos.

2.7 Documentación de usuario

Se debe proveer una guía de uso (user guide – en inglés) de la aplicación en formato “.pdf” disponible al usuario.

Éste debe contener:

- Introducción de la aplicación
- Instalación
- Uso
- Reportar un problema.

2.8 Dependencias y suposiciones

La aplicación depende del correcto funcionamiento del sistema Android. Cualquier falla en dicho sistema puede afectar seriamente el funcionamiento de la aplicación. En primer lugar, se infiere que el usuario tiene conexión a Internet para poder descargar el archivo “.apk”.

Se espera que el espacio libre de almacenamiento sea el requerido para instalar la aplicación. Caso contrario, el sistema proveerá un mensaje de fallo de instalación. También, se espera que el usuario haya activado la opción: “Permitir fuentes desconocidas” desde la aplicación Ajustes/Seguridad/Administración del dispositivo. Esto permite instalar aplicaciones distintas a las existentes en el Google Play Store (tienda de aplicaciones).

Por último, es necesario que el teclado del sistema esté en distribución QWERTY – español.

3. Requerimientos de interfaz externos

3.1 Interfaces de usuario

- Uso del estilo Material Design que rige en el ecosistema Android.
- Orientación de pantalla: vertical.
- Botones:
 - Ingresar (en vista de inicio de sesión)
 - Registrarse (
 - Selección de dificultad de nivel (en vista de menú inicial)
 - Estadísticas (en vista de menú inicial)
 - Fácil (en vista selección de nivel)
 - Intermedio (en vista selección de nivel)
 - Difícil (en vista selección de nivel)
 - Cancelar (en vista activa)
 - Menú principal (en vista estadísticas)
 - Salir (en vista de menú inicial)
- Fuente de texto: Roboto.

3.2 Interfaces de hardware

- Teléfonos con sistema Android.
- Compatibilidad: Android 7 o mayor.
- Uso de pantalla táctil y botones en dicha pantalla.
- Se delega el control de hardware al sistema Android.

3.3 Interfaces de software

- Java Development Kit 8.
- IDE: Android Studio 3.3.2
- Android 9.0 "Pie".
- SQLite 3.28.0
- Gradle 4.1
- Librerías varias de Android para interfaz gráfica y funcionamiento.

3.4 Interfaces de comunicación

- Para la comunicación y administración entre servidor y la base de datos MySQL se utilizará la herramienta phpMyAdmin.

- El servidor será implementado mediante el servicio de hosting de <https://ar.000webhost.com/>.
- El dominio de la página web será <http://itype.ml/> implementado con el servicio de dominios <https://www.freenom.com/es/index.html?lang=es>.

4. Requerimientos funcionales

La siguiente lista contiene todos los requerimientos funcionales que se deberán cumplir en la aplicación. Están identificados unívocamente con un nombre y una breve descripción.

4.1 REQ 1: Entrada del usuario:

4.1.1 Descripción y prioridad

Entrada de datos mediante teclado. Prioridad media. Se asigna dicha prioridad ya que el sistema Android limita directamente esta funcionalidad.

4.1.2 Estímulo y secuencia de respuesta

- 1) Abrir la aplicación
 - Se despliega la vista de inicio de sesión
- 2) Ingresar usuario y contraseña de forma correcta.
 - Se despliega la vista de selección de nivel.
- 3) Seleccionar un nivel cualquiera.
 - Se despliega la vista activa.
- 4) Situar en el cajón de texto.
 - Se debe desplegar el teclado
- 5) Ingresar caracteres mediante teclado en pantalla.
 - Solamente se despliegan caracteres de tipo letra y espacio.
- 6) Realizar la prueba correctamente.
 - Se obtendrá una nueva palabra para competir.
- 7) Reingresar una nueva palabra distinta a la anterior.
 - Se observará que la nueva palabra es tomada como correcta.

4.1.3 Requerimientos funcionales

R1.1: En la vista activa, la aplicación deberá permitir al usuario ingresar solo caracteres del tipo letra y espacios.

R1.2: En la vista activa, la entrada será insensible a las mayúsculas por lo que se comparará siempre en minúscula.

R1.3: La palabra escrita será eliminada automáticamente al momento de ser reconocida.

R1.4: La entrada mediante teclado será únicamente carácter por carácter. La función de texto predictivo deberá estar deshabilitada dentro de la aplicación.

4.2 REQ 2: Interfaz:

4.2.1 Descripción y prioridad

Las vistas del programa especifican los distintos escenarios de funcionamiento, acceso y despliegue de datos.

Prioridad: Alta.

4.2.2 Estímulo y secuencia de respuesta

- 1) Abrir la aplicación.
 - La vista de inicio de sesión se deberá desplegar.
- 2) Ingresar usuario y clave
 - a) Mensaje de: "inicio correcto"
 - Se despliega la vista de selección de nivel
 - b) Mensaje: "No existe usuario"
 - Se despliega la vista Registro
 - b) Mensaje: "Contraseña inválida".
 - Reintentar en vista actual (de inicio de sesión)
- 3) Ingresar usuario y clave existentes y de manera correcta
 - Se despliega la vista de selección de nivel.
- 4) Selección de nivel aleatorio: se deben divisar los botones "fácil", "intermedio" y "difícil".
 - Se selecciona un nivel y se despliega la vista activa.
- 4) En modo vista activa, se realiza la prueba hasta partida correcta.
 - Al finalizar la prueba, se despliega la vista de estadísticas.
- 5) Se divisa la tabla de posiciones y demás datos estadísticos.
 - Botón de regresar. Se despliega la vista de selección de nivel.

4.2.3 Requerimientos funcionales

R2.1: Las vistas deberán ser:

- Inicio sesión.
- Registro
- Selección de nivel.
- Vista activa (Juego en marcha).
- Vista a la tabla de posiciones.

R2.2: Tanto en la vista Inicio de sesión como en la vista de selección de nivel se deberá implementar un botón que permita al usuario salir de la aplicación

R2.3: Tanto en la vista activa como en la de tabla de posiciones se deberá implementar un botón que permita al usuario ir a la vista de selección de nivel.

R2.4: En la vista de tabla de posiciones se deberán mostrar:

- El nombre del usuario
- El número del puesto en la tabla de posiciones
- La velocidad de la prueba en palabras por minuto
- El nivel de dificultad de la prueba

R2.5: En la vista activa se deberá mostrar una sola palabra modelo a tipear.

R2.6: La tabla de estadísticas deberá estar compuesta por los 10 puestos con mayor velocidad, ordenados de mayor a menor; siendo el primer puesto el más rápido.

R2.7: Se deberá permitir la elección de tres niveles de dificultad en la vista “selección de nivel”. Ellos son:

- Fácil.
- Intermedio.
- Difícil.

4.3 REQ 3: Sesión:

4.3.1 Descripción y prioridad

Se requiere garantizar el inicio de sesión de forma segura y confiable, así como la creación de la cuenta ante no estar registrado.

Prioridad: Alta.

4.3.2 Estímulos y secuencia de respuesta

- 1) Abrir la aplicación.
 - La vista de inicio de sesión se deberá desplegar.
- 2) Ingresar usuario y clave no registrados.
 - Se despliega el teclado. Se observan que los caracteres en el cuadro de clave deben estar protegidos.
- 3) Mensaje: “No existe usuario”
 - Se despliega la vista Registro
- 4) Registrar e ingresar un nuevo usuario y clave.
 - Mensaje: “Registro exitoso”. Se despliega la vista de inicio de sesión.
 - Mensaje: “Usuario ya existente. Reintentar”.
- 5) Ingresar usuario y clave creados recientemente.

- Mensaje de: “inicio correcto”. Se despliega la vista de selección de nivel

4.3.3 Requerimientos funcionales

R3.1: Tanto la contraseña como el usuario se deberá codificar con el formato UTF-8.

R3.2: Se deberá poder iniciar y registrar en la base de datos.

R3.3: En caso de que el usuario exista y la contraseña no coincida con los datos almacenados; se debe indicar que ésta es incorrecta y se pedirá de nuevo un inicio de sesión.

4.4 REQ 4: Funcionamiento general

4.4.1 Descripción y prioridad

Características del juego en la vista activa en cuanto al puntaje, duración y forma de validación.

Prioridad: Alta

4.4.2 Estímulos y secuencia de respuesta

- 1) Abrir la aplicación.
 - La vista de inicio de sesión se deberá desplegar.
- 2) Ingresar usuario y clave
 - a) Mensaje de: “inicio correcto”
 - Se despliega la vista de selección de nivel
 - b) Mensaje: “No existe usuario”
 - Se despliega la vista Registro
 - b) Mensaje: “Contraseña inválida”.
 - Reintentar en vista actual (de inicio de sesión)
- 3) Ingresar usuario y clave existentes y de manera correcta
 - Se despliega la vista de selección de nivel.
- 4) Selección de nivel aleatorio: se deben divisar los botones “fácil”, “intermedio” y “difícil”.
 - Se selecciona un nivel y se despliega la vista activa.
- 5) En modo vista activa, se inserta la palabra, pero sin el espacio final.
 - Se debe obtener mensaje de palabra incorrecta.
- 6) No ingresar palabra alguna hasta finalizar el tiempo de prueba
 - Fin de prueba. Se despliega vista de estadísticas.
- 7) No ingresar palabra alguna hasta finalizar el tiempo de prueba

- Corroborar puntaje final igual a 0.
- 8) Reintentar prueba y cancelar inmediatamente.
 - Despliegue de vista de selección de nivel.

4.4.3 Requerimientos funcionales

R4.1: La prueba de velocidad deberá finalizar en un tiempo de 30 segundos,

R4.2: La entrada del usuario será reconocida como palabra para la comparación luego de haber escrito al menos un carácter seguido de un espacio.

R4.3: La velocidad final se calculará como los caracteres totales “tipeados” por sesenta, dividido cinco veces el tiempo total de la prueba.

Esto es:

$$Vf = (60 * Tc) / (5 * Tp)$$

Siendo:

- **Vf:** Velocidad final. Medida en: palabras por minuto.
- **Tc:** Total de caracteres correctos “tipeados”. Medido en: caracteres.
- **Tp:** Tiempo total de la prueba. Medido en: segundos.

R4.4: Al finalizar el juego, se deberá subir los datos a la base de datos de forma automática.

R4.5: Al finalizar el juego, se deberá mostrar la vista Estadísticas de juego y presentar la información de la partida más reciente.

R4.6: La actualización de estadísticas se implementará mediante un sistema de notificación el cual consiste en que el servidor deberá enviar la actualización de las tablas a todos los usuarios actualmente conectados y viendo las estadísticas.

5. Casos de Uso

5.1 Definición de casos de uso

- Usuario:

CU1: Crear usuario

CU2: Iniciar sesión

CU3: Presionar botón en menú principal (ir a selección de dificultad)

CU4: Elección de la dificultad

CU5: Realizar la prueba

CU6: Ver estadísticas de juego

CU7: Salir del programa

- Servidor:

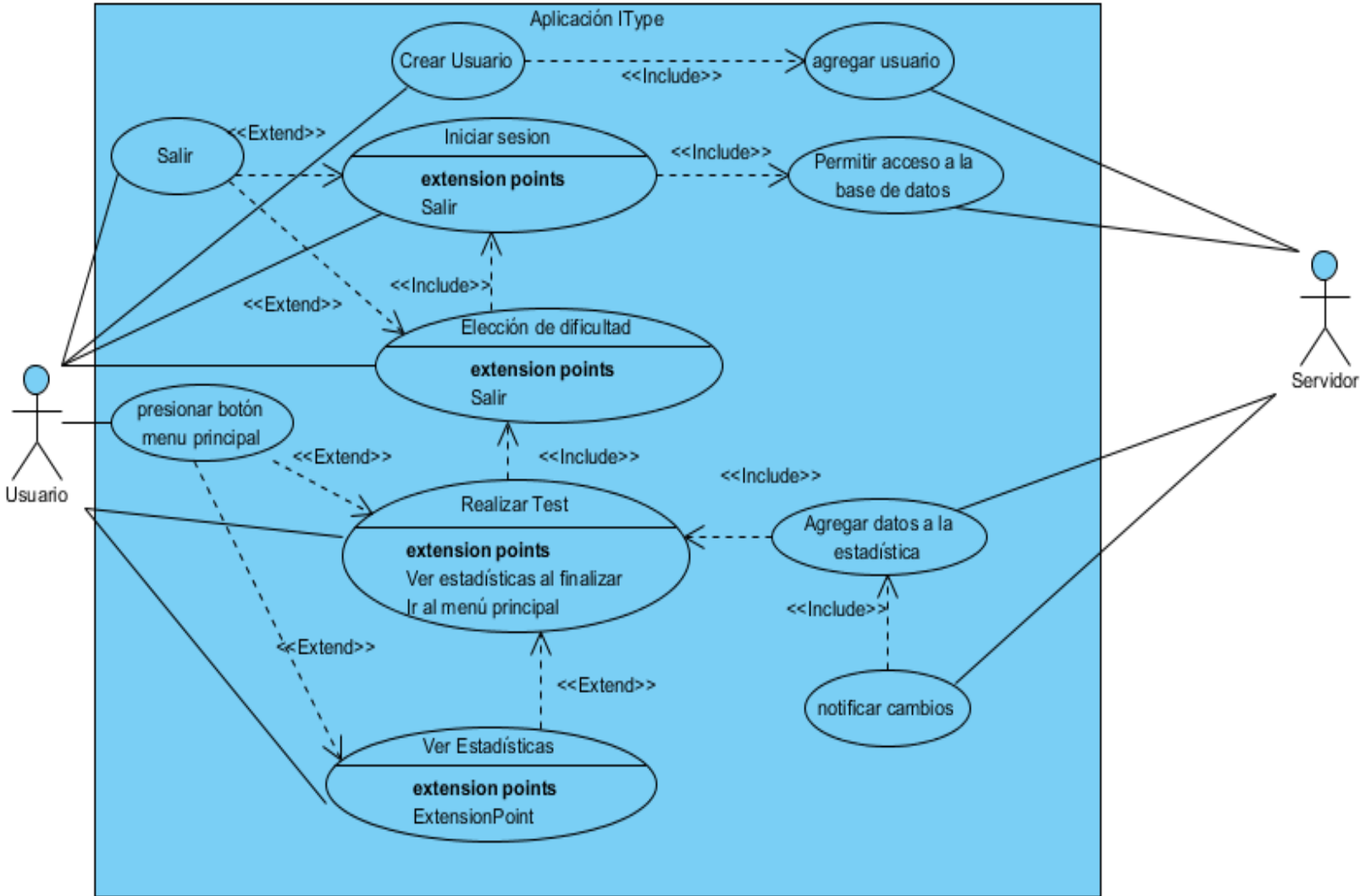
CS1: Agregar usuario

CS2: Permitir acceso a la base de datos

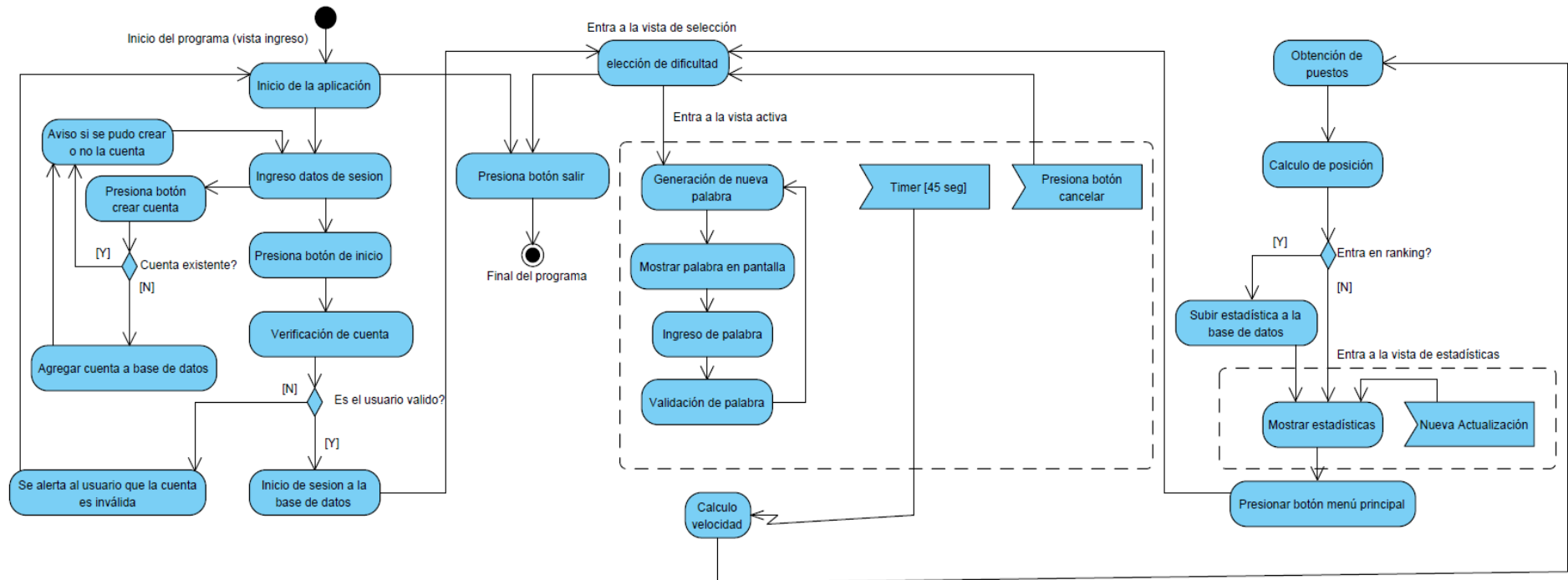
CS3: Agregar datos a la estadística

CS4: Notificar cambios

5.2 Diagrama de casos de uso



5.3 Diagrama de actividades



5.4 Matriz de trazabilidad

Relación entre requerimientos funcionales y casos de uso

Casos de Uso / Requisitos funcionales	CU1	CU2	CU3	CU4	CU5	CU6	CU7	CS1	CS2	CS3	CS4
R1.1					X						
R1.2					X						
R1.3					X						
R1.4					X						
R2.1		X		X	X	X					
R2.2			X	X	X		X				
R2.3			X		X	X					
R2.4						X					
R2.5					X						
R2.6						X					
R2.7				X							
R3.1	X	X						X			
R3.2	X	X						X	X		
R3.3	x	X						X			
R4.1					X						
R4.2					X						
R4.3					X						
R4.4					X	X			X	X	
R4.5					X	X					
R4.6						X					X

6. Requerimientos no funcionales

6.1 Requerimientos de rendimiento (performance)

RN1.1: Toda funcionalidad del sistema debe responder al usuario en menos de 3 segundos.

RN1.2: El sistema deberá ser capaz de operar correctamente con hasta por lo menos 1000 usuarios en línea.

RN1.3: El sistema debe tener una disponibilidad del 99,9% de las veces en que un usuario intente acceder

6.2 Requerimientos de eficiencia

RN2.1: La aplicación no deberá ocupar más de 200 Megabytes de espacio en disco

RN2.2: La aplicación deberá consumir menos de 80 Megabytes de RAM.

RN2.3: La aplicación no deberá pedir accesos a datos del sistema alguno. Ejemplo: localización, contactos, acceso a mensajes, imágenes, etc.

RN2.4: La aplicación no deberá solicitar permisos de uso a cámara, micrófono u otros periféricos.

6.3 Requerimientos de seguridad

RN3.1: El sistema incluirá un procedimiento de autorización de usuarios, en el cual los usuarios deben identificarse usando un nombre de usuario y contraseña. Sólo los usuarios autorizados de esta forma podrán acceder a los datos del sistema.

6.4 Atributos de calidad del software

RN4.1: El promedio del tiempo de aprendizaje del sistema por un usuario deberá ser menor a quince minutos.

7. Casos de prueba de sistema

7.1 Definición de casos de prueba de sistema

Identificador de caso de prueba: TR1.1

Descripción de la prueba: verificar que no se pueden ingresar caracteres distintos a letra o espacio en la vista activa

Funcionalidad a probar: teclado en pantalla.

Ejecución de la prueba (pasos):

1. Abrir la aplicación.
2. Iniciar sesión con cuenta existente.
3. Seleccionar dificultad aleatoriamente.
4. Iniciar juego.
5. Situar en la caja de texto
6. Observar el teclado y recorrer sus opciones durante el juego.

Resultado esperado: no presenta caracteres distintos a letras o espacios.

Identificador de caso de prueba: TR1.2

Descripción de la prueba: Verificar que letras en mayúscula o minúscula son indistintas.

Funcionalidad a probar: Algoritmo de validación de palabra.

Ejecución de la prueba (pasos):

1. Abrir la aplicación.
2. Iniciar sesión con cuenta existente.
3. Seleccionar dificultad aleatoriamente.
4. Ingresar la palabra correcta intercalando letras en mayúscula y minúscula.

Resultado esperado: la palabra ha sido aceptada correctamente por el sistema de validación de palabras.

Identificador de caso de prueba: TR1.3

Descripción de la prueba: palabra ingresada ha sido borrada luego de ingresar un espacio final.

Funcionalidad a probar: base de datos y algoritmo de validación de palabras

Ejecución de la prueba (pasos):

1. Abrir la aplicación.
2. Iniciar sesión con cuenta existente.
3. Seleccionar dificultad aleatoriamente.
4. Ingresar la palabra y un espacio correctamente.
5. Ingresar la siguiente palabra y un espacio correctamente.

Resultado esperado: verificación de ambas palabras correctamente.

Identificador de caso de prueba: TR1.4

Descripción de la prueba: verificar ingreso de texto de modo caracter por caracter.

Funcionalidad a probar: texto predictivo desactivado

Ejecución de la prueba (pasos):

1. Abrir la aplicación
2. Situar en el cuadro de texto para usuario y contraseña.
3. Ingresar correctamente a la aplicación con sesión válida
4. Iniciar juego.
5. Situar en cuadro de entrada de texto.
6. Ingresar un caracter cualquiera.

Resultado esperado: el teclado no muestra palabras auto-sugeridas.

Identificador de caso de prueba: TR2.1

Descripción de la prueba: observar y enumerar las vistas requeridas.

Funcionalidad a probar: correcta definición y acceso a las vistas

Ejecución de la prueba (pasos):

1. Abrir la aplicación
2. Ingresar con un usuario válido.
3. Seleccionar nivel fácil
4. Realizar la jugada
5. Finalizar el juego e ingresar a la vista de estadísticas
6. Volver a menú principal.

Resultado esperado: correcto acceso en orden y transferencia de las vistas.

Identificador de caso de prueba: TR2.2

Descripción de la prueba: probar el botón “salir” en vistas “Inicio de sesión” y “Selección de nivel”.

Funcionalidad a probar: botón Salir.

Ejecución de la prueba 1 (pasos):

1. Abrir la aplicación.
2. Hacer clic en el botón “Salir”.

Resultado esperado: se ha cerrado la aplicación.

Ejecución de la prueba 2 (pasos):

1. Abrir la aplicación.
2. Ingresar de forma válida al sistema.
3. Divisar el botón “Salir”
4. Hacer clic en el botón “Salir”

Resultado esperado: se ha cerrado la aplicación.

Identificador de caso de prueba: TR2.3

Descripción de la prueba: interrumpir la vista actual y volver al menú principal

Funcionalidad a probar: botones “Cancelar prueba” y “Nueva prueba”

Ejecución de la prueba 1 (pasos):

1. Abrir la aplicación
2. Ingresar de forma válida al sistema
3. Seleccionar el nivel “fácil”
4. En la vista “Activa” hacer clic en el botón “Cancelar prueba” y aceptar la advertencia.

Resultado esperado: el sistema muestra la vista de selección de nivel.

Ejecución de la prueba 2 (pasos):

1. Abrir la aplicación
2. Ingresar de forma válida al sistema
3. Seleccionar el nivel “fácil”
4. Realizar y finalizar la prueba.
5. En la vista de estadísticas, hacer clic en el botón “Nueva prueba”.

Resultado esperado: el sistema muestra la vista de selección de nivel.

Identificador de caso de prueba: TR2.4

Descripción de la prueba: observar los resultados de la jugada

Funcionalidad a probar: algoritmo de validación y de puntaje; uso, guardado y obtención de información de la base de datos.

Ejecución de la prueba (pasos):

1. Abrir la aplicación
2. Abrir la aplicación
3. Ingresar de forma válida al sistema
4. Seleccionar el nivel “fácil”
5. Realizar y finalizar un juego de forma correcta.

Resultado esperado: atributos de juego son mostrados en la tabla de estadísticas de forma correcta.

Identificador de caso de prueba: TR2.5

Descripción de la prueba: Verificar la cantidad de palabras modelo

Funcionalidad a probar: algoritmo de selección y carga de palabra.

Ejecución de la prueba (pasos):

1. Abrir la aplicación
2. Ingresar de forma válida al sistema
3. Seleccionar el nivel “fácil”
4. Completar una palabra y colocar un espacio al finalizar.

Resultado esperado: se observa una única palabra modelo por vez.

Identificador de caso de prueba: TR2.6

Descripción de la prueba: observar la organización de la tabla de estadísticas.

Funcionalidad a probar: registro de jugadas, uso, adición y obtención de datos de jugadas a la base de datos estadística.

Ejecución de la prueba 1 (pasos):

1. Llenar la tabla de puestos de dificultad fácil con 10 usuarios a una velocidad de 100.
2. Abrir la aplicación
3. Ingresar de forma válida al sistema
4. Seleccionar el nivel “fácil”
5. Colocar espacios por cada palabra modelo.
6. Verificar puntaje cero. Observar la posición en la tabla.

Resultado esperado: No se presenta el usuario dentro de los primeros 10 puestos.

Ejecución de la prueba 2 (pasos):

1. Vaciar la tabla de puestos de dificultad fácil.
2. Abrir la aplicación
3. Ingresar de forma válida al sistema
4. Seleccionar el nivel “fácil”
5. Colocar espacios por cada palabra modelo.
6. Verificar puntaje cero. Observar la posición en la tabla.

Resultado esperado: El usuario se haya en la posición 1.

Identificador de caso de prueba: TR2.7

Descripción de la prueba: observar la vista de selección de nivel.

Funcionalidad a probar: Vista “selección de nivel”.

Ejecución de la prueba 1 (pasos):

1. Abrir la aplicación
2. Ingresar de forma válida al sistema

Resultado esperado: Observar los botones “Fácil”, “Intermedio” y “Difícil”.

Identificador de caso de prueba: TR3.1

Descripción de la prueba: uso de formato UTF-8 para las contraseñas y usuarios.

Funcionalidad a probar: cifrado UTF-8 de datos de inicio de sesión.

Ejecución de la prueba (pasos):

1. Abrir la aplicación
2. Registrar un nuevo usuario
3. Verificar que se ha realizado la encriptación correctamente

Resultado esperado: Retorno de función de encriptación correcto.

Identificador de caso de prueba: TR3.2

Descripción de la prueba: obtener datos, comparar y validar usuario y contraseña con información en la base de datos de usuario.

Funcionalidad a probar: algoritmo de validación de cuenta.

Ejecución de la prueba (pasos):

1. Abrir la aplicación
2. Registrar usuario inexistente
3. Iniciar sesión con un usuario ya registrado
4. Verificar que se pueda ingresar correctamente

Resultado esperado: Inicio de sesión exitoso

Identificador de caso de prueba: TR3.3

Descripción de la prueba: Detectar y avisar a usuarios no registrados en la base de datos que la cuenta no es válida.

Funcionalidad a probar: algoritmo de validación de cuenta, base de datos, comparación con la base de datos.

Ejecución de la prueba (pasos):

1. Iniciar aplicación.
2. Ingresar un usuario no registrado actualmente

Resultado esperado: Mensaje de “Usuario inválido en pantalla”

Identificador de caso de prueba: TR4.1

Descripción de la prueba: se verifica la duración temporal del juego.

Funcionalidad a probar: Temporizador de prueba

Ejecución de la prueba (pasos):

1. Abrir la aplicación
2. Ingresar de forma válida al sistema
3. Seleccionar botón “Fácil”.
4. Verificar el tiempo hasta que la prueba finaliza, (se transfiere a la vista de estadística)

Resultado esperado: La duración de la transferencia de vistas es de 30 segundos.

Identificador de caso de prueba: TR4.2

Descripción de la prueba: se prueba la correcta transición de palabra a otra nueva palabra.

Funcionalidad a probar: algoritmo de validación de palabra

Ejecución de la prueba (pasos):

1. Abrir la aplicación
2. Ingresar de forma válida al sistema
3. Seleccionar botón “Fácil”.
4. Ingresar un espacio en la entrada

Resultado esperado: Observar que la palabra modelo se actualizó.

Identificador de caso de prueba: TR4.3

Descripción de la prueba: se probará la correcta obtención del puntaje.

Funcionalidad a probar: algoritmo de validación, obtención del puntaje, asociación de datos.

Ejecución de la prueba (pasos):

1. Abrir la aplicación

2. Ingresar de forma válida al sistema
3. Seleccionar dificultad fácil.
4. Realizar la prueba contando los caracteres correctos ingresados.

Resultado esperado: Verificar que se cumpla la ecuación de puntaje

Identificador de caso de prueba: TR4.4

Descripción de la prueba: el fin de cada jugada implica subir datos a la base de datos de estadística en forma automática.

Funcionalidad a probar: algoritmo de obtención de datos estadísticos, base de datos de resultados

Ejecución de la prueba (pasos):

1. Abrir la aplicación
2. Ingresar de forma válida al sistema
3. Seleccionar botón "Fácil".
4. Realizar una jugada asegurando que supere a los que actualmente están dentro de los 10 puestos más rápidos.

Resultado esperado: Actualización de los puestos con el nuevo usuario.

Identificador de caso de prueba: TR4.5

Descripción de la prueba: prueba de vista resultante al finalizar el tiempo de jugada

Funcionalidad a probar: algoritmo de validación, cronómetro, vistas.

Ejecución de la prueba (pasos):

1. Abrir la aplicación
2. Ingresar de forma válida al sistema
3. Seleccionar botón "Fácil".
4. Esperar que la prueba finalice
- 5.

Resultado esperado: despliegue de vista Estadísticas

Casos de prueba para requerimientos no funcionales

Identificador de caso de prueba: TRN1.1

Descripción de la prueba: evaluación de eficiencia temporal del sistema.

Funcionalidad a probar: tiempo de respuesta.

Ejecución de la prueba (pasos):

1. Iniciar la herramienta de testing en tiempo real.
2. Iniciar la aplicación.
3. Realizar el paso por las vistas.
4. Observar el tiempo transcurrido entre cada acción.

Resultado esperado: cada acción no debe tomar más de 3 segundos.

Identificador de caso de prueba: TRN1.2

Descripción de la prueba: prueba de sobrecarga de base de datos de usuario y palabras.

Funcionalidad a probar: base de datos.

Ejecución de la prueba (pasos):

1. Iniciar la herramienta de testing en tiempo real
2. Iniciar la aplicación
3. Iniciar correctamente con 10 usuarios distintos.
4. Seleccionar modo de prueba fácil
5. Realizar la prueba

Resultado esperado: correcto funcionamiento hasta por lo menos 10 usuarios en línea

Identificador de caso de prueba: TRN1.3

Descripción de la prueba: prueba de actualización de base de datos

Funcionalidad a probar: base de datos de usuario, palabras y estadísticas.

Ejecución de la prueba (pasos):

1. Iniciar aplicación
2. Iniciar sesión correctamente
3. Elegir dificultad fácil
4. Esperar que la prueba finalice
5. Verificar el tiempo desde que la prueba finaliza hasta que se inicia la vista de estadísticas.

Resultado esperado: tiempo de actualización menor a 2 segundos.

Identificador de caso de prueba: TRN1.4

Descripción de la prueba: prueba de base de datos y servidor.

Funcionalidad a probar: peticiones al servidor y accesibilidad a la base de datos.

Ejecución de la prueba (pasos):

1. Iniciar aplicación
2. Registrar una cantidad de 200 usuarios a la base de datos con nombre de usuario 1 hasta 200 y sin contraseña.

Resultado esperado: Verificar los 200 usuarios se han registrado correctamente.

Identificador de caso de prueba: TRN2.1-2

Descripción de la prueba: prueba de eficiencia del sistema

Funcionalidad a probar: uso de espacio en disco y memoria RAM.

Ejecución de la prueba (pasos):

1. Inicio de Android Studio
2. Inicio de aplicación
3. Inicio del monitor de memoria
4. Correr el sistema y realizar un juego.

Resultado esperado: uso de espacio en disco menor a 200 MB y memoria RAM menor a 80 [MB]

Identificador de caso de prueba: TRN2.3-4

Descripción de la prueba: prueba de eficiencia y acceso a recursos.

Funcionalidad a probar: acceso a periféricos del teléfono o datos de sistema.

Ejecución de la prueba (pasos):

1. Instalar la aplicación
2. Iniciar aplicación
3. Iniciar sesión
4. Realizar juego
5. Salir de la aplicación

Resultado esperado: no petición de acceso a elementos de sistema o periféricos

Identificador de caso de prueba: TRN4.1

Descripción de la prueba: accesibilidad.

Funcionalidad a probar: facilidad e intuitividad del sistema.

Ejecución de la prueba (pasos):

1. Iniciar cronómetro
2. Iniciar aplicación
3. Entregar a usuario sin previo contacto con la aplicación
4. Observar la interacción del mismo
5. Esperar a que el usuario pueda realizar al menos un test correctamente

Resultado esperado: tiempo transcurrido para la comprensión de la aplicación menor a 15 minutos.

7.2 Definición de casos de prueba alternativos

Identificador de caso de prueba: TA1.1

Descripción de la prueba: Prueba de velocidad incremental

Funcionalidad a probar: velocidad del algoritmo de generación de palabras y respuesta de la aplicación ante una situación atípica.

Ejecución de la prueba (pasos):

1. Iniciar aplicación
2. Iniciar sesión
3. Realizar la prueba con una velocidad incremental de caracteres correctos por minuto.

Resultado esperado: Verificar el límite de la respuesta de la aplicación ante velocidades superiores a la capaz por cualquier ser humano.

Identificador de caso de prueba: TA1.2

Descripción de la prueba: Forzar caracteres no permitidos por la aplicación

Funcionalidad a probar: algoritmo de validación de palabras

Ejecución de la prueba (pasos):

1. Iniciar aplicación

2. Iniciar sesión
3. Realizar la prueba forzando caracteres inusuales.

Resultado esperado: Correcto funcionamiento del sistema de validación.

Identificador de caso de prueba: TA1.3

Descripción de la prueba: Realizar la prueba e ir a la vista de estadísticas con una cantidad paralela de usuarios incremental.

Funcionalidad a probar: Sistema de actualización de estadísticas por parte del servidor.

Ejecución de la prueba (pasos):

1. Registrar una nueva cuenta
2. Iniciar sesión con la cuenta creada
3. Realizar la prueba en dificultad fácil
4. Observar actualizaciones en la tabla de estadísticas
5. Repetir desde 1.

Resultado esperado: Verificar el límite de la capacidad del servidor para manejar actualizaciones.

7.3 Pruebas de aceptación

De los casos de prueba establecidos en el apartado anterior, se eligen como pruebas de aceptación para el usuario las pruebas:

- | | | | |
|---------|---------|----------|------------|
| • TR1.1 | • TR2.4 | • TR4.1 | • TRN2.1-2 |
| • TR1.2 | • TR2.6 | • TR4.4 | • TRN4.1 |
| • TR1.3 | • TR3.1 | • TR4.5 | • TRA1.3 |
| • TR1.4 | • TR3.2 | • TRN1.1 | |
| • TR2.1 | • TR3.3 | • TRN1.4 | |

7.4 Pruebas smoke

- TR2.1
- TR2.4
- TR2.6
- TR3.2
- TR3.3
- TR4.4
- TRN1.3

7.5 Matriz de trazabilidad

Relación entre requerimientos funcionales y casos de prueba

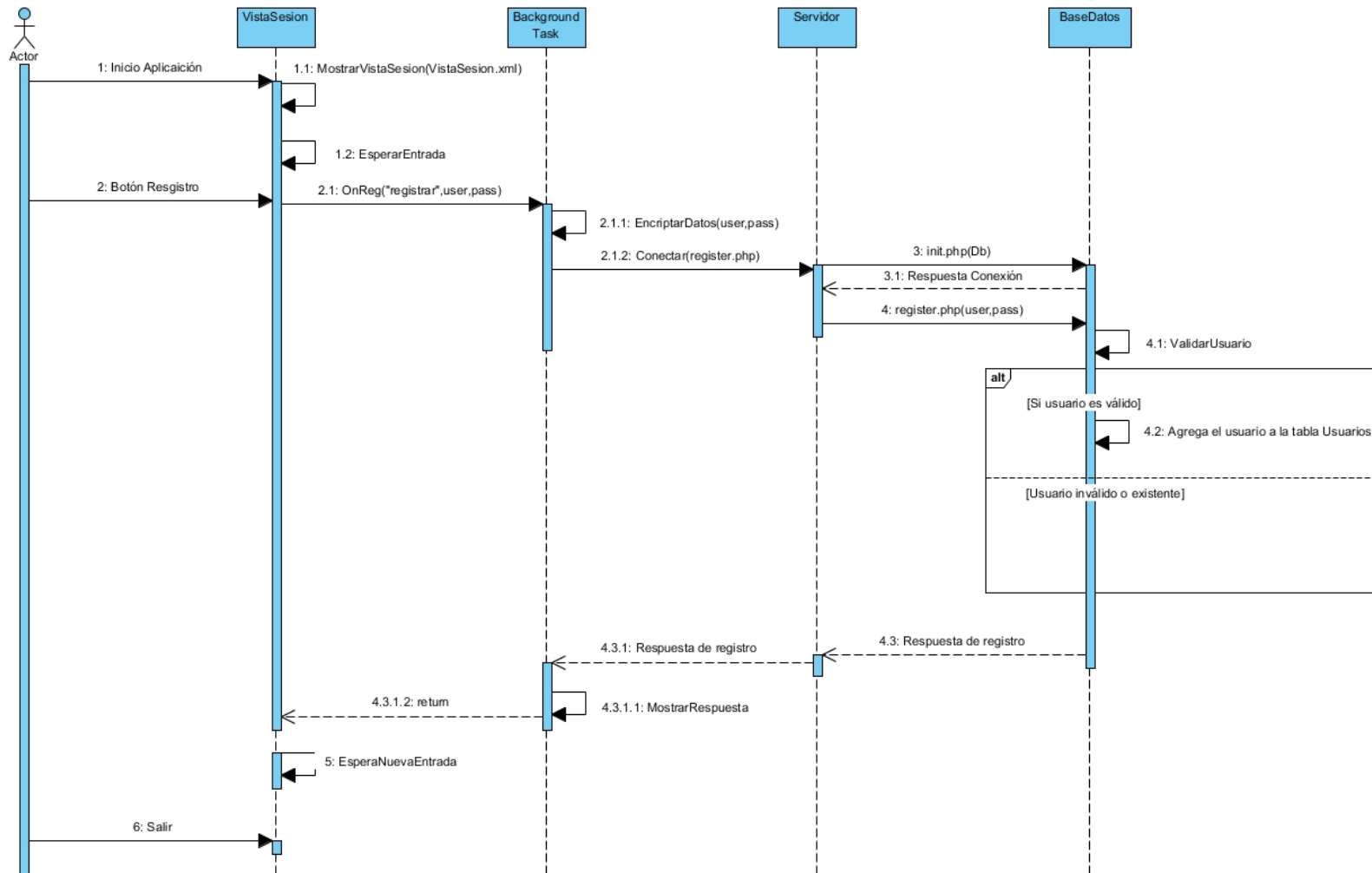
Casos de prueba / Requerimientos	TR1.1	TR1.2	TR1.3	TR1.4	TR2.1	TR2.2	TR2.3	TR2.4	TR2.5	TR2.6	TR2.7	TR3.1	TR3.2	TR3.3	TR4.1	TR4.2	TR4.3	TR4.4	TR4.5	TRN1.1	TRN1.2	TRN1.3	TRN1.4	TRN2.1-2	TRN2.3-4	TRN4.1	TA1.1	TA1.2	TA1.3
R1.1	X								X	X						X	X	X											
R1.2		X							X								X	X											
R1.3			X						X								X	X											
R1.4	X	X	X	X					X								X	X											
R2.1					X																						X		
R2.2						X																							
R2.3							X																						
R2.4								X																					
R2.5			X						X																				
R2.6										X																			
R2.7	X	X	X		X		X	X	X	X	X				X		X	X	X								X	X	X
R3.1	X			X								X	X			X													
R3.2	X	X	X	X	X	X	X	X	X	X	X	X	X		X	X	X	X	X								X	X	X
R3.3														X															
R4.1								X							X														X
R4.2	X	X	X						X	X						X	X	X	X										
R4.3										X							X	X											
R4.4										X								X											
R4.5					X			X		X									X										
R4.6										X								X											

Relación entre requerimientos no funcionales y casos de prueba

Casos de prueba / Requerimientos	TA1.3	TA1.2	TA1.1	TRN4.1	TRN2.3-4	TRN2.1-2	TRN1.4	TRN1.3	TRN1.2	TRN1.1	TR4.5	TR4.4	TR4.3	TR4.2	TR4.1	TR3.3	TR3.2	TR3.1	TR2.7	TR2.6	TR2.5	TR2.4	TR2.3	TR2.2	TR2.1	TR1.4	TR1.3	TR1.2	TR1.1
RN1.1	X		X				X	X	X	X	X	X	X	X	X	X	X	X	X	X	X		X	X	X	X	X	X	X
RN1.2		X					X	X	X																				
RN1.3	X	X						X	X	X	X	X	X	X	X	X	X	X	X	X	X		X	X	X				
RN2.1						X																							
RN2.2						X																							
RN2.3					X																								
RN2.4					X																								
RN3.1	X	X	X	X				X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
RN4.1				X																									

8. Diagramas adicionales

8.1 Diagrama de secuencia para el registro de usuario



8.2 Diagrama de secuencia para el juego

