Especificación de requerimientos

**IType**

Aplicación Android para

pruebas de velocidad de tipeo

**Versión 1.0**

**Presentado por:**

Cancinos, José María

Giraudo, Juan Pablo

Oliva, Nahuel David

**Fecha de la Versión:** 10/04/2019

**Índice**

**Historial de versiones ………………………………………………………… 3**

**1- Introducción …………………………………………………………………….. 4**

1.1 Propósito del documento …………………………………………………... 4

1.2 Convenciones del documento ………………………………………….. 4

1.3 Alcance del producto ………………………………………………….. ……….. 4 1.4 Referencias …………………………………………………... 4

**2- Descripción General ………………………………………………………….. 5**

2.1 Perspectiva del producto …………………………………………………..... 5

2.2 Funciones del producto ……………………………………………………. 5

**3- Restricciones generales …..………………………………………….. 5**

**4- Dependencias y asunciones …..………………………………………….. 6**

**5- Lista de requerimientos …..………………………………………….. 6**

**Historial de versiones**

|  |  |  |
| --- | --- | --- |
| **Nombre de la versión** | **Fecha** | **Número de versión** |
| IType | 10/04/2019 | 1.0 |

1. **Introducción**
   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. **Convenciones del documento**

Este documento ha sido creado en base a la planilla IEEE[[1]](#footnote-1) para documentos de especificación de requerimientos.

* 1. **Destinatarios del documento**

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

* 1. **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 palabras correctamente en el menor tiempo posible. La cantidad de palabras correctas se suma. Luego, al finalizar el juego, se da a conocer mediante datos estadísticos el nivel de habilidad que se posee; dependiendo de la cantidad de palabras acertadas.

La aplicación, también, ofrece la posibilidad de escoger distintos niveles de dificultad en base a la habilidad que se posee. A mayor dificultad, mayor cantidad de letras en palabras se posee. De tal forma, se incentiva a los usuarios mejorar sus habilidades para obtener mayores puntos.

* 1. **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**

1. **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 puntaje: el usuario puede publicar el puntaje en el servidor del sistema para obtener estadísticas sobre el juego y su desempeño

1. **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 del 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

A medida que el usuario finalice de escribir las palabras que se muestran en la vista activa, éstas se actualizarán automáticamente hasta cumplir el tiempo de duración de la prueba.

La duración de cada prueba es de 30 segundos. Iniciando la cuenta desde el instante en que aparece la primera palabra 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 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.

1. **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.

Tiene un conocimiento bueno sobre el idioma español ya que los nombres en menú y botones están dicho 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).

1. **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 Android[[2]](#footnote-2) 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

1. **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 SQLite.

Uso de sistema de manejo de datos apropiado para el control de la base de datos.

1. **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.

1. **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.

**5. Lista de 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.

**REQ 1: Entrada del usuario:**

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

**R1.2:** No se deberá permitir doble espacio. En caso de que suceda se lo tomará como uno solo.

**R1.3:** La entrada será insensible a las mayúsculas por lo que se comparará siempre en minúscula.

**R1.4:** La palabra escrita será reconocida luego de un espacio como palabra si por lo menos contiene un carácter.

**R1.5:** La palabra escrita será eliminada al momento de ser reconocida; imposibilitando la corrección de la misma.

**R1.6:** La entrada mediante teclado será únicamente caracter por caracter. La función de texto predictivo será deshabilitada dentro de la aplicación.

**REQ 2: Interfaz:**

**R2.2:** Las vistas deberán ser:

* Menú principal.
* Selección de nivel.
* Vista activa – Juego en marcha.
* Pantalla de puntaje.

**R2.2:** Cada vista deberá poder ser intercambiada por el usuario en cualquier momento. En caso de haber empezado la prueba, ésta se cancelará automáticamente.

**R2.3:** En la vista de estadísticas se deberán mostrar:

* el nombre del usuario
* el puesto en la tabla de puntajes general
* la velocidad de la prueba en palabras por minuto??
* el nivel de dificultad de la prueba
* el número de usuarios en línea??.

**R2.4:** Al finalizar todas las palabras en pantalla, éstas se deberán actualizar automáticamente mostrando un nuevo texto actualizado listo para “tipear”.

**R2.5:** Estos textos de actualización serán formados por cinco palabras.

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

La entrada del usuario en la tabla con mejor jugada deberá ser desplegada también.

**R2.7:** Tanto en la vista de inicio de sesión como en la del menú principal se deberá tener un botón que permita salir de la aplicación.

**R2.8:** Se deberá permitir la elección de tres niveles de dificultad en la vista “selección de nivel”. Ellos son:

* Fácil.
* Intermedio.
* Difícil.

**R2.9:** Al terminar la prueba deberán presentarse:

* Un botón que permita realizarla de nuevo.
* Otro botón para ver estadísticas.
* Otro para salir al menú principal.

**REQ 3: Sesión:**

**R3.1:** La contraseña deberá contener al menos 5 caracteres.

**R3.2:** En caso de que el usuario no exista, se deberá preguntar si se desea crearlo. En caso afirmativo, se deberá agregar a 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.

PREGUNTA:

Cómo hacemos el control de sesión? Un SLQLite en un servidor Web?

**REQ 4: Funcionamiento general:**

**R4.1:** La prueba de velocidad deberá finalizar en un tiempo de 45 segundos.

**R4.2:** En caso que se hayan completado todas las palabras, al ingresar el último espacio se deberá detener el temporizador y finalizar la prueba.

~~Teniendo en cuenta los caracteres de la última palabra escrita si es que coinciden con el texto mostrado,en caso contrario no se tendrá en cuenta.~~

**R4.3:** Las palabras escritas deberán ser válidas sólo si todos sus caracteres coinciden con el de la palabra respectiva mostrada en el texto como modelo.

**R4.4:** Al terminar la prueba se deberá mostrar automáticamente la vista de estadísticas. Se deberá mostrar la calificación y el puesto. Ver R2.3.

**R4.5:** La entrada deberá ser validada con el texto mostrado en pantalla. Se realiza palabra por palabra. Y se deberá remarcar en color verde si coincide el resultado, o en rojo en caso contrario.

**R4.6:** Las palabras deberán ser automáticamente comparadas con las respectivas en el texto luego de un espacio.

**R4.7:** La velocidad final se calculará como los caracteres totales “tipeados” por sesenta, dividido por trescientas veces el tiempo total de la prueba.

Esto es:



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.

**R4.8:** En caso que se interrumpa la prueba, se deberá terminar la misma como si ha sido una manera normal de finalización.

**REQ 5: Base de Datos**

**R5.1:** La base de datos deberá implementarse como SQLite.

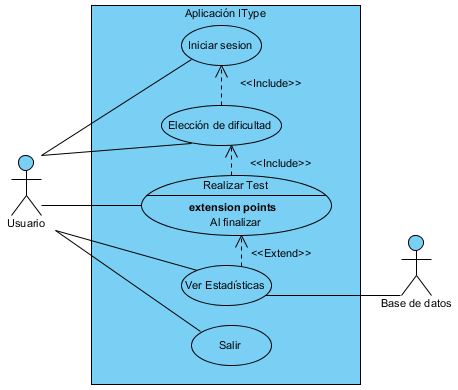
**R5.2:** Cada nivel de dificultad deberá contener una base de datos distinta. Siendo entonces la cantidad de base de datos igual a 3.

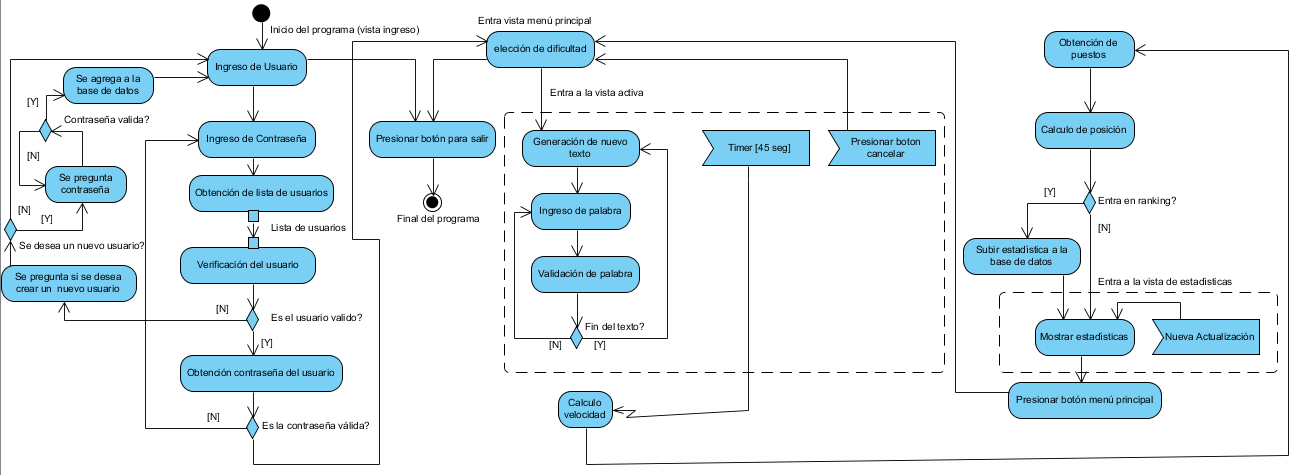
**R5.3:** Cada base de datos debe contener las siguientes columnas:

* Nº ID
* Palabra
* Dificultad
* Referencia

**R5.4:** Deben existir como mínimo 50 palabras de cada nivel de dificultad.

**R5.4:** Estas palabras deben ser las palabras a comparar para poder ganar el juego.





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> [↑](#footnote-ref-1)
2. Para Android 7 “Nougat”. Android. https://www.android.com/enterprise/recommended/requirements/ [↑](#footnote-ref-2)