

Universitat de Lleida
Escola Politècnica Superior
Màster en Interacció Persona-Ordinador

Treball de final de màster

TAPADÚ: Un lanzador de aplicaciones Android por voz

Autora: Vanessa Andrés Blas
Directores: Montserrat Sendín Veloso
Francisco Jurado Monroy
Septiembre, 2014

Agradecimientos

A mis padres y mi hermano, que siempre han estado conmigo, y me han ayudado a levantar de mis caídas.

A Miguel, por ser mi constante en todo momento.

A mis directores de proyecto por sus consejos y directrices.

Al señor Domingo, director de la agencia administrativa de la fundación ONCE de Lleida, por dedicarme su tiempo y enseñarme la realidad del estado de accesibilidad en los smartphones.

Finalmente, a todos aquellos que han participado en las pruebas, a los que me han dado su apoyo, a mi resolvedor de dudas internacional, a mis vecinitos. A todos.

Gracias, sin vosotros no hubiera llegado hasta aquí.

Contenido

1	Introducción.....	1
1.1	Justificación del TFM	1
1.2	Descripción general.....	1
1.3	Experiencia previa	1
1.4	Objetivos del TFM	1
1.5	Planificación	2
1.6	Resultados deseados.....	3
1.7	Estructura de la memoria.....	3
2	Estado del arte	4
2.1	Accesibilidad en Android.....	4
2.2	Accesibilidad en iOS	5
2.3	Asistentes virtuales	6
3	Análisis del sistema.....	8
3.1	Descripción general.....	8
3.1.1	Las rutinas	8
3.1.2	Las palabras clave.....	8
3.1.3	Los componentes del sistema	8
3.1.4	El público objetivo.....	9
3.2	Casos de uso: Aplicación Android	10
3.2.1	Crear rutina	11
3.2.2	Consultar rutinas	12
3.2.3	Modificar rutinas.....	12
3.2.4	Eliminar rutinas	14
3.2.5	Subir el catálogo al servidor	15
3.2.6	Descargar catálogo.....	16
3.2.7	Realizar registro.....	17
3.2.8	Modificar opciones.....	18
3.3	Casos de uso: Widget	18
3.3.1	Ejecutar rutina.....	19
4	Diseño del sistema	20
4.1	Tecnologías a usar	20
4.1.1	Aplicación Android y Widget.....	20

4.1.2	Aplicación web	21
4.1.3	Comunicación entre Android y Aplicación web	23
4.2	Diseño conceptual.....	24
5	Prototipo.....	25
5.1	Aplicación Android	25
5.2	Widget lanzador	26
6	Evaluación de accesibilidad y pruebas con usuarios	27
6.1	Evaluación de accesibilidad.....	27
6.1.1	Resultados	27
6.2	Pruebas con usuarios	28
6.2.1	Propósito y objetivo del test	28
6.2.2	Participantes.....	29
6.2.3	Tareas	29
6.2.4	Entorno del test.....	30
6.2.5	Métricas de usabilidad	33
6.2.6	Resultados	34
6.3	Interacción con lectores de pantalla.....	37
7	Conclusiones	39
7.1	Resultados obtenidos.....	39
7.2	Trabajo futuro	40
7.3	Reflexión final.....	41
8	Bibliografía	43
Anexo A.	Patrón MVC.....	45
A.1	Componentes del MVC	45
Anexo B.	Diagramas de colaboración, clases y base de datos.....	46
B.1	Diagrama de colaboración	46
B.1.1	Aplicación Android	46
B.1.2	Widget	47
B.1.3	Aplicación web	47
B.2	Diagrama de clases.....	47
B.2.1	Aplicación Android	48
B.2.2	Widget	48
B.2.3	Aplicación web	48
B.3	Diseño de la base de datos.....	48

Anexo C. Accesibilidad en Android y evaluación con expertos de TAPADÚ	49
C.1 Introducción	49
C.2 Recomendaciones	50
C.2.1 Recomendaciones generales.....	50
C.2.2 Teclado	50
C.2.3 Dispositivos apuntadores	51
C.2.4 Pantalla.....	51
C.2.5 Sonidos y Multimedia.....	52
C.2.6 Notificación al usuario.....	52
C.2.7 Información de objetos	52
C.2.8 Tiempo.....	53
C.2.9 Documentación	53
C.2.10 Otros.....	53
C.3 Resultados detallados	54
C.3.1 Prioridad 1	54
C.3.2 Prioridad 2	57
Anexo D. Pruebas con usuarios.....	58
D.1 Formulario de consentimiento en la participación del proyecto.....	58
D.2 Transcripción del facilitador.....	59
D.3 Cuestionarios.....	60
D.3.1 Cuestionario pre-test	60
D.3.2 Cuestionario post-tarea.....	61
D.3.3 Cuestionario post-test.....	62
Anexo E. Resultados de las pruebas con usuarios.....	63
E.1 Tarea 1: Lanzar una aplicación con éxito	63
E.1.1 Resumen de la tarea por participantes y perfiles	63
E.1.2 Efectividad.....	63
E.1.3 Eficiencia.....	64
E.1.4 Satisfacción.....	65
E.1.5 Problemas detectados.....	65
E.2 Tarea 2: Intentar lanzar una aplicación sin éxito	66
E.2.1 Resumen de la tarea por participantes y perfiles	66
E.2.2 Efectividad.....	66
E.2.3 Eficiencia.....	67

E.2.4	Satisfacción.....	68
E.2.5	Problemas detectados.....	68
E.3	Tarea 3: Crear una nueva rutina.....	69
E.3.1	Resumen de la tarea por participantes y perfiles	69
E.3.2	Efectividad	69
E.3.3	Eficiencia.....	70
E.3.4	Satisfacción.....	71
E.3.5	Problemas detectados.....	71
E.4	Cuestionario de satisfacción de la aplicación.....	72

Índice de figuras

Ilustración 1. Planificación del proyecto inicial.....	2
Ilustración 2. Interacción entre los componentes del sistema	9
Ilustración 3. Casos de uso de la aplicación Android	10
Ilustración 4. Casos de uso del widget Android	19
Ilustración 5. Distribución de las versiones en Android en agosto 2013.	20
Ilustración 6. Diagrama conceptual.....	24
Ilustración 7. Esquema general del prototipo de la aplicación Android	25
Ilustración 8. Esquema general del prototipo del widget	26
Ilustración 9. El widget no tiene asociado que pertenece a la aplicación TAPADÚ.....	38
Ilustración 10. Patrón MVC con las tecnologías usadas en el proyecto.....	45
Ilustración 11. Diagrama de colaboración para "Crear rutina"	46
Ilustración 12. Diagrama de colaboración "Consultar catálogo"	46
Ilustración 13. Diagrama de colaboración "Consultar rutina"	46
Ilustración 14. Diagrama de colaboración "Modificar rutina"	46
Ilustración 15. Diagrama de colaboración "Eliminar rutina"	46
Ilustración 16. Diagrama de colaboración "Descargar catálogo".....	46
Ilustración 17. Diagrama de colaboración "Subir catálogo"	47
Ilustración 18. Diagrama de colaboración "Ejecutar rutina"	47
Ilustración 19. Diagrama de colaboración "Eliminar rutina"	47
Ilustración 20. Diagrama de colaboración "Descarga catálogo"	47
Ilustración 21. Diagrama de colaboración "Subir catálogo"	47
Ilustración 22. Diagrama de clases de la aplicación Android.	48
Ilustración 23. Diagrama de clases del widget.	48
Ilustración 24. Diagrama de clases de la aplicación web.	48
Ilustración 25. Diagrama de la base de datos.	48
Ilustración 26. El menú principal no recoge el foco	54
Ilustración 27. Hay información que al ser leída por el lector de pantalla es confusa	54
Ilustración 28. Se duplica la palabra botón al ser leído por el lector de pantalla.	55
Ilustración 29. El lector de pantalla no detecta la modificación del control.....	55
Ilustración 30. El mensaje de rutina no encontrada debe mostrarse en mensaje de error	56
Ilustración 31. No existe la opción salir de la aplicación.....	56
Ilustración 32. El contraste de los botones lilas, es insuficiente.....	57
Ilustración 33. El menú principal no recoge el foco	57

Ilustración 34. Porcentaje finalización de la tarea 1 por perfiles. La media del perfil 1 y 2 al ser la misma se solapan.	64
Ilustración 35. Grado de problemas de la tarea 1 por perfiles. La media del perfil 1 y 2 al ser la misma se solapan.	64
Ilustración 36. Tiempos de la tarea 1 por perfiles.....	64
Ilustración 37. Grado de complejidad de la tarea 1 por perfiles. La media del perfil 1 y 2 al ser la misma se solapan.	65
Ilustración 38. Confusión entre la aplicación y el widget.....	65
Ilustración 39. Porcentaje de finalización de la tarea 2 por perfiles. La media del perfil 1 y 2 al ser la misma se solapan.	67
Ilustración 40. Grado de problemas de la tarea 2 por perfiles. La media del perfil 1 y 2 al ser la misma se solapan.	67
Ilustración 41. Tiempos de la tarea 2 por perfiles.....	67
Ilustración 42. Grado de complejidad de la tarea 2. La media del perfil 1 y 2 al ser la misma se solapan.	68
Ilustración 43. Pantalla consulta rutina.....	68
Ilustración 44. Porcentaje de finalización de la tarea 3 por perfiles.....	70
Ilustración 45. Grado de problemas de la tarea 3 por perfiles.	70
Ilustración 46. Tiempos de la tarea 3 por perfiles.....	70
Ilustración 47. Grado de complejidad de la tarea 3 por perfiles.....	71
Ilustración 48. Pantalla seleccionar rutina y consultar rutina.....	71
Ilustración 49. Pantalla en la que se asocia la palabra clave a la aplicación.....	72
Ilustración 50. Cuestionario SUS por perfiles.....	72

1 Introducción

1.1 Justificación del TFM

Tras haber cursado la asignatura optativa de Accesibilidad y el creciente interés de las aplicaciones en dispositivos móviles, se ha decidido presentar como proyecto final del máster Interacción persona-ordenador, una aplicación en Android que cumpla con los requisitos de accesibilidad requeridos.

La motivación personal para la realización del proyecto es adentrarse en el mundo de desarrollo de Android, y conseguir diseñar, desarrollar y evaluar una aplicación que siga las directrices de accesibilidad que se recomiendan.

1.2 Descripción general

La finalidad de este proyecto es crear una aplicación (TAPADÚ) que pueda lanzar otras aplicaciones, facilitando así el punto de inicio de éstas. Por ejemplo un reproductor de música que pudiera ser controlado únicamente por comandos de voz, para iniciarlo se debería o bien acceder a él a través de un acceso directo o buscarlo a través del menú. A través de TAPADÚ se facilita esta tarea y se evitan pantallas repletas de accesos directos, que pierden su funcionalidad.

Se espera conseguir una aplicación accesible en su manejo, y que tenga un público objetivo que incluya personas mayores y personas con discapacidades visuales.

1.3 Experiencia previa

La única experiencia sobre Accesibilidad que se posee es la de haber cursado la optativa de Accesibilidad del Máster Interacción Persona-Ordenador.

Por otro lado, no se posee experiencia ni conocimiento sobre el desarrollo de aplicaciones en Android, aunque sí se poseen extensos conocimientos sobre JAVA, lenguaje utilizado mayoritariamente para el desarrollo en Android.

1.4 Objetivos del TFM

El objetivo final es construir una aplicación en Android que a través de un comando de voz inicie una aplicación. Por consiguiente es necesario crear una aplicación con una fuerte base en usabilidad y accesibilidad que permita configurar estos comandos de voz, que en adelante serán llamados rutina, y un widget que se encargue de escuchar al usuario para que inicie la aplicación tras escuchar el comando de voz. Además se añade la opción de guardar el catálogo de rutinas configuradas en un servidor externo, permitiendo descargar todas las rutinas configuradas en un nuevo dispositivo tras haberse logueado.

Durante el desarrollo del TFM se pretenden cumplir los siguientes objetivos y requerimientos.

Objetivos

1. Diseñar y desarrollar una aplicación que cumpla con los requisitos de accesibilidad implantados por Android.
2. Diseñar y desarrollar una aplicación que cumpla con los aspectos de usabilidad.
3. Diseñar y desarrollar una aplicación con un público objetivo que incluya personas mayores y personas con discapacidades visuales.
4. Diseñar y desarrollar una aplicación que se comunice con un servidor externo a través de servicios web.
5. Diseñar y desarrollar una aplicación y widget que utilice reconocimiento de voz como entrada.
6. Distribuir la aplicación a través de Google Play.
7. La aplicación debe ser sencilla e intuitiva. Cumpliendo con las guías de diseño que recomienda Android y principios de accesibilidad.
8. La aplicación debe ser construida con vistas de que en un futuro se puedan añadir más acciones que no sea sólo la ejecución de aplicaciones.

1.5 Planificación

Para planificar mejor el proyecto, se divide en 5 etapas:

- Trabajo previo: búsqueda de información, estado del arte accesibilidad en dispositivos móviles, planificación, análisis y diseño del sistema.
- Desarrollo de la aplicación Android de creación de rutinas: Aprendizaje de Android, reconocimiento de voz y desarrollo de la aplicación.
- Desarrollo del widget en Android de ejecución de rutinas.
- Desarrollo de aplicación web que incorpore un servicio web para gestionar el catálogo.
- Pruebas, resultados y final de memoria.

El proyecto tiene una estimación de 765 ut. La ilustración 1 muestra las tareas en las que se ha dividido cada fase así como la planificación realizada teniendo en cuenta un trabajo de 5 horas diarias de lunes a viernes.

Task Mode	Task Name	Duration	Start	Finish	Predecessors	Resource Names
1	Fase 1: Trabajo previo	253 hrs	Mon 03/02/14	Mon 14/04/14		
2	Búsqueda de información sobre la accesibilidad en Android y iOS.	58 hrs	Mon 03/02/14	Tue 18/02/14		Vanessa
3	Planificación del trabajo.	13 hrs	Wed 19/02/14	Fri 21/02/14	2	Vanessa
4	Desarrollo de los requerimientos de la aplicación.	38 hrs	Mon 24/02/14	Wed 05/03/14	3	Vanessa
5	Desarrollo de los casos de uso.	58 hrs	Thu 06/03/14	Fri 21/03/14	4	Vanessa
6	Desarrollo del diseño de la aplicación.	58 hrs	Mon 24/03/14	Tue 08/04/14	5	Vanessa
7	Documentación	18 hrs	Wed 09/04/14	Mon 14/04/14	6	Vanessa
8	Fase 2: Desarrollo de la aplicación en Android	218 hrs	Tue 15/04/14	Fri 13/06/14		
9	Búsqueda de información sobre la accesibilidad en Android	13 hrs	Tue 15/04/14	Thu 17/04/14		Vanessa
10	Búsqueda de información sobre reconocimiento de voz.	13 hrs	Fri 18/04/14	Tue 22/04/14	9	Vanessa
11	Configuración del entorno de trabajo.	23 hrs	Wed 23/04/14	Tue 29/04/14	10	Vanessa
12	Implementación de la aplicación	163 hrs	Wed 30/04/14	Fri 13/06/14	11	Vanessa
13	Fase 3: Desarrollo del widget	98 hrs	Mon 16/06/14	Fri 11/07/14		
14	Búsqueda de información sobre los widget de Android.	23 hrs	Mon 16/06/14	Fri 20/06/14		Vanessa
15	Implementación del widget conforme al diseño realizado en la etapa de trabajo previo.	73 hrs	Mon 23/06/14	Fri 11/07/14	14	Vanessa
16	Fase 4: Desarrollo del servidor en la nube	73 hrs	Mon 14/07/14	Fri 01/08/14		
17	Búsqueda de información sobre el almacenamiento en la nube y su comunicación en aplicaciones Android.	23 hrs	Mon 14/07/14	Fri 18/07/14		Vanessa
18	Implementación de la nube y una pequeña interfaz para gestionar las rutinas subidas por el usuario.	48 hrs	Mon 21/07/14	Fri 01/08/14	17	Vanessa
19	Fase 5: Pruebas, resultados y documentación	123 hrs	Mon 04/08/14	Fri 05/09/14		
20	Pruebas con diferentes usuarios sobre diferentes escenarios.	73 hrs	Mon 04/08/14	Fri 22/08/14		Vanessa
21	Análisis de los resultados y propuesta de mejoras	23 hrs	Mon 25/08/14	Fri 29/08/14	20	Vanessa
22	Finalizar la documentación.	23 hrs	Mon 01/09/14	Fri 05/09/14	21	Vanessa

Ilustración 1. Planificación del proyecto inicial

1.6 Resultados deseados

Tras finalizar el proyecto se desea haber conseguido los siguientes resultados:

- Haber desarrollado una aplicación funcional que esté alojada en Google Play.
- Haber conseguido los objetivos.
- Haber realizado pruebas con usuarios con un resultado satisfactorio.
- Conseguir una lista de mejoras y nuevas funcionalidades para una versión posterior.

1.7 Estructura de la memoria

Esta memoria se desarrolla en 7 capítulos:

- El primer capítulo es la introducción en la que se explica la justificación del proyecto, los objetivos, el alcance y la planificación del proyecto.
- El segundo capítulo describe el estado del arte de la accesibilidad en los dispositivos móviles: las herramientas y software de los sistemas operativos Android e iOS, los asistentes virtuales etc.
- El tercer capítulo describe el análisis funcional de la aplicación que se desea desarrollar. En ella se especifican y aclaran conceptos, se desarrollan los casos de uso para cada componente, etc.
- El cuarto capítulo describe el diseño técnico de la aplicación: la justificación del uso de cada tecnología en cada apartado, y el diseño de la aplicación.
- El quinto capítulo muestra el prototipado de la aplicación Android y widget.
- El sexto capítulo detalla las pruebas realizadas y los resultados obtenidos.
- En el séptimo capítulo se encuentran las conclusiones que contienen: un análisis de los resultados obtenidos, la lista de objetivos cumplidos y una lista de mejoras.

2 Estado del arte

Desde la aparición de los SmartPhones, los diferentes sistemas han realizado grandes avances en cuanto a accesibilidad, facilitando el uso de dispositivos móviles a usuarios que poseen alguna discapacidad que les obliga a interactuar y utilizarlos de formas distintas.

A pesar de haber diferentes sistemas operativos, se realiza una breve descripción del estado del arte de accesibilidad en los dos sistemas operativos más populares: Android e iOS, explicando qué herramientas son las más usadas, la facilidad de desarrollo de aplicaciones accesibles y debilidades de cada uno.

2.1 Accesibilidad en Android

Android es un sistema operativo basado en Linux y orientado a dispositivos móviles. Al ser un sistema abierto, son muchos los fabricantes de este tipo de terminales los que lo utilizan en sus productos.

Android, con cada nueva versión, ha ido incluyendo una serie de características generales de accesibilidad en su sistema. Por ejemplo, en la última versión lanzada 4.4 KitKat[1] ha implementado el apartado de subtítulos, para que el usuario diga en qué idioma, tamaño y estilo quiere ver los subtítulos de los vídeos, con lo que todas las aplicaciones que muestran subtítulos en sus vídeos tendrían que respetar esa configuración.

Las características más importantes en accesibilidad[2] que se encuentran en Android son:

- **Talkback:** Es el lector de pantallas desarrollado por Eyes Free Project bajo la tutela de Google. Normalmente ya viene preinstalado en la mayoría de los dispositivos, sin embargo, algunos fabricantes como Huawei y LG no lo incluyen por defecto en alguno de sus modelos, aunque puede ser instalado a través de la tienda Google Play[3].
- **Gestos de ampliación:** tocando 3 veces con un dedo en la pantalla se realiza una ampliación de la zona tocada para ayudar a ver algo en detalle.
- **Personalización del tamaño general de la fuente de letra para todo el sistema:** de esta forma se consigue centralizar y hacer que todos los mensajes, contenidos y controles amplíen su tamaño o lo reduzcan adaptándose a las necesidades del usuario.
- **Tecla bloqueo descuelga:** una función de atajo de teclado para facilitar la función del teléfono a personas con problemas para localizar rápidamente elementos en la pantalla.
- **Control sobre el giro de la pantalla:** para aquellos usuarios que siempre utilicen el dispositivo en modo apaisado o vertical y no quieren que eso se modifique.
- **Google TTS:** Es el motor de síntesis de voz que por lo general ya viene preinstalado en la mayoría de dispositivos. Sin embargo, como ocurre con TalkBack, algunos fabricantes podría no incluirlos y deben ser instalados desde Google Play.
- **Exploración por toque:** Es una característica que funciona con Talkback permitiendo tocar la pantalla y escuchar qué se está tocando. Esta característica es útil para personas con problemas de visibilidad.

Pero la verdadera innovación en accesibilidad para Android viene de la mano de las aplicaciones desarrolladas por usuarios concienciados e interesados en utilizar los dispositivos

móviles como intermediarios entre el usuario con necesidades especiales y el entorno que le rodea.

Así existen alternativas a las herramientas proporcionadas por Google como por ejemplo *Spiel*[4] como lector de pantalla o *eSpeak*[5], una voz metálica de muy poco peso que es más ágil que la voz de Google y tiene mejor gestión de los acentos y algunas consonantes que la última versión de la voz de Google. Pero además existen muchas otras aplicaciones que facilitan la interacción de los dispositivos móviles como pueden ser: *Parkible*[6] una aplicación accesible que encuentra aparcamiento de minusválidos, simplificar la pantalla mostrando grandes iconos como la aplicación *Big Launcher*[7] etc.

Android facilita numerosas guías[8] para desarrollar aplicaciones que sean accesibles y puedan ser usadas por un mayor número de usuarios.

La gran debilidad de Android, es a la vez una de sus características principales que se ha destacado al principio. Al ser un sistema abierto, es usado en multitud de dispositivos con diferentes especificaciones técnicas y configuraciones que pueden hacer que un servicio o producto no funcione correctamente en algún dispositivo. Además imposibilita la opción de que el desarrollador pueda probarlo en todos los dispositivos.

2.2 Accesibilidad en iOS

iOS es un sistema operativo móvil de la empresa Apple que fue originalmente desarrollado para el iPhone (iPhone OS), y después usado en dispositivos como el iPod Touch, iPad y el Apple TV. Apple, no permite la instalación de iOS en hardware de terceros.

iOS proporciona diversas opciones de accesibilidad[9] que facilitan el uso de los dispositivos a personas con discapacidad. Algunas de las características más relevantes son:

- **Voice Over:** Es el lector de pantalla que viene instalado en el Sistema operativo que lee todo aquello que se toca. Modifica algunos comandos táctiles y en lugar de presionar una vez para accionar un botón o un vínculo, se debe presionar dos veces, la primera para que el sistema lea y la segunda para activar la acción
- **Zoom:** Es un ampliador de pantalla integrado que funciona en cualquier lugar de iOS. Se activa pulsando con 3 dedos dos veces un sector de la pantalla.
- **Invertir colores:** el ajuste se aplica a todo el sistema, incluso al vídeo.
- **Leer selección:** Al seleccionar un texto, y tras haber habilitado la opción, se muestra una opción de Voz que lee el texto seleccionado.
- **Letra más grande, texto en negrita, aumentar contraste:** Opciones para mejorar la visualización de la pantalla, ajustándose a las necesidades de los usuarios.
- **Reducir movimiento:** Desde iOS 7 las capas que conforman la interfaz de la pantalla de inicio tienen movimiento propio que responde a la posición del giroscopio. Esta opción se puede desactivar.
- **Subtítulos y subtítulos opcionales:** Su característica más destacada es que permite al usuario configurar el estilo de los subtítulos
- **Audio Mono:** es una característica que permite a los usuarios con pérdida auditiva que utilicen los auriculares, reproducir todo el sonido en un mismo canal.

- **FaceTime.** Los usuarios con discapacidades auditivas que prefieran comunicarse en lengua de signos pueden utilizar esta aplicación para realizar videollamadas. Además, pueden intercambiar mensajes de texto a través de la aplicación iMessage.
- **Control por botón:** Esta opción da la posibilidad de controlar el iPhone o iPad mediante movimientos de cabeza. Aunque viene desactivada y se debe configurar a través de la cámara, para decidir qué gestos asociar a qué acciones.
- **AssistiveTouch:** permite adaptar la los gestos compatibles con iOS a las necesidades del usuario. Por ejemplo, si algunos gestos, como pellizcar, al usuario les resulta difíciles de realizar, se puede hacer lo mismo solo tocando con un dedo o incluso creando un gesto propio.
- **Siri:** es un asistente de voz virtual y navegador de conocimiento que trabaja como una aplicación para iOS. La aplicación utiliza una interfaz de usuario de lenguaje natural para responder preguntas, hacer recomendaciones y ejecutar acciones delegando solicitudes a un conjunto de servicios web.

Al igual que en Android, Apple facilita numerosas guías para desarrollar aplicaciones accesibles[10], listas de comprobaciones etc. y puedan ser usadas por un mayor número de usuarios. No obstante, la cuota de mercado de iOS es menor (15.2%) frente a Android (78.6%)¹, esto hace que desarrollar aplicaciones en iOS cuente con una comunidad menor y además es más restrictiva las aplicaciones que se pueden adquirir sobre el iTunes.

Como debilidad en iOS, aunque los manuales de usuario de Apple, incluyen un extenso apartado dedicado a las características de accesibilidad en sus dispositivos, los cambios que se van realizado no son documentados activamente. De hecho, resulta complicado encontrar información sobre las actualizaciones más recientes en materia de accesibilidad. Además, en el manual de usuario, esta información se incluye al final del todo como parte de los apéndices. Esto obliga al usuario tener que descubrir por sí mismo las nuevas funciones de accesibilidad.

2.3 Asistentes virtuales

Un asistente virtual utiliza procesamiento del lenguaje oral para responder preguntas, hacer recomendaciones y realizar acciones mediante la delegación de solicitudes hacia un conjunto de servicios web. En este aspecto, TAPADÚ, podría verse como una especie de asistente virtual ya que comparte las siguientes funcionalidades:

- La interacción se produce mediante reconocimiento de voz, consiguiendo una interacción natural.
- Permite la ejecución de determinadas funcionalidades sobre algunas aplicaciones.

Tras el lanzamiento de SIRI por parte de iOS en 2011 junto a la versión iOS 5, los desarrolladores de Android dieron respuesta creando sus propias aplicaciones con unas funcionalidades similares a las ofrecidas por SIRI, algunas de las más importantes son:

- **Sherpa** es un asistente desarrollado completamente en español. Con él se pueden conocer los resultados de fútbol, leer noticias, enviar sms o hacer llamadas, escuchar

¹ <http://www.xatakamovil.com/mercado/idc-lo-confirma-android-ha-arrasado-en-2013-con-una-cuota-de-mercado-del-78-6>

las canciones que se pidan, ser guiados al lugar de una cita, escribir comentarios en nuestro muro de Facebook o tuitear etc. con simples comandos de voz que interpreta con gran solvencia.

- **Everfriends:** es un asistente que funciona en español. La aplicación dispone de funciones como la información de actualidad, la meteorológica, la opción de configurar la hora a la que sonará el despertador, realice una llamada, envíe mensajes de texto etc. Sin embargo su uso es complejo, lento y muy poco ágil.
- **Speaktoit Assistant:** Es otro asistente virtual con un buen manejo y que ejecuta correctamente las acciones encomendadas como: tareas habituales de agenda, envío de emails, búsquedas en Google... que son de gran ayuda al ir por la calle o conduciendo.

No obstante, existen diferencias entre TAPADÚ y un asistente virtual como son:

- TAPADÚ sólo permite la ejecución de otras aplicaciones y sólo responde a las palabras clave previamente configuradas.
- Es configurable, adaptándose a aquellas necesidades reales del usuario.

3 Análisis del sistema

3.1 Descripción general

TAPADÚ pretende ser una aplicación escalable y flexible, que permita añadir nuevas funcionalidades y se adapte a los futuros cambios que el sistema operativo pueda incorporar.

3.1.1 Las rutinas

A lo largo del documento se menciona en varias ocasiones la palabra rutina. Dentro del contexto de la aplicación, una rutina se define como *una secuencia de acciones a realizar de forma automática, y que se ejecutan al ser pronunciada y procesada una "palabra clave"*.

En este proyecto sólo se abarca como rutina la ejecución de otras aplicaciones, sin embargo, en un futuro se podría ampliar a otras acciones e incluso acciones encadenadas como por ejemplo escribir un correo indicando un escrito común a un destinatario particular.

3.1.2 Las palabras clave

La palabra clave es una o más palabras que deben ser únicas. Al ser pronunciadas por el usuario y procesadas por la aplicación, busca qué acción y sobre qué aplicación debe ejecutarla.

Además, deben ser vocablos exclusivos, es decir, que una misma palabra no esté asociada a diferentes rutinas, y tampoco pueden ser una parte de otra palabra clave. Por ejemplo si ya existe la palabra "correo" no se debe permitir como nueva palabra clave "nuevo correo", ya que el ruido de alrededor podría hacer que parte del mensaje no se entendiera y sólo cogiera un fragmento de la palabra clave, ejecutando una aplicación no deseada.

3.1.3 Los componentes del sistema

Para conseguir esta finalidad se ha diseñado un sistema compuesto por los siguientes 3 componentes:

- Una aplicación que se ejecuta en Android y cuyas funcionalidades son:
 - o Creación de una nueva rutina.
 - o Gestión de las rutinas ya generadas: modificar su palabra clave, eliminar rutinas ya instaladas.
 - o Subir a la nube el propio catálogo de rutinas.
 - o Descargar un catálogo propio de rutinas.
 - o Modificar las opciones de la aplicación.
- Un widget, que al ser lanzado capture la palabra clave y realice la acción que tiene asignada.
- Una aplicación web que contenga los servicios web necesarios para añadir y consulta el catálogo de los usuarios.

La ilustración 2 muestra la interacción entre cada uno de estos componentes:

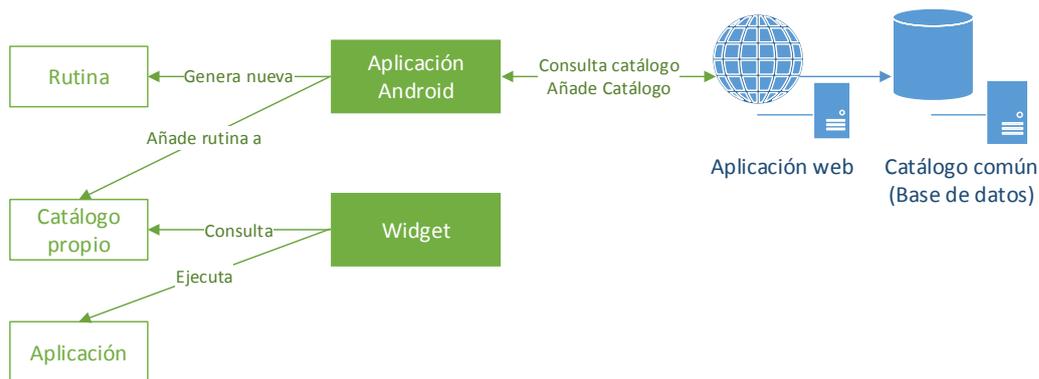


Ilustración 2. Interacción entre los componentes del sistema

3.1.4 El público objetivo

El público objetivo que comprende la aplicación se puede dividir en tres categorías:

- **Usuarios jóvenes** que usan dispositivos tecnológicos de forma frecuente. Para ellos, la aplicación TAPADÚ puede ser vista como un acceso más rápido para acceder a sus aplicaciones favoritas. Son usuarios que, al usar muchas aplicaciones, suelen tener las pantallas repletas de iconos. Además, se añaden aquellas situaciones en las que no se puede prestar atención al dispositivo y se quiere lanzar una aplicación, como por ejemplo, conduciendo y usando el manos libres, corriendo...
- **Usuarios mayores** que hacen un uso esporádico de dispositivos tecnológicos y que, por consiguiente, no tienen soltura en su manejo. En estos casos, la aplicación TAPADÚ elimina el proceso de búsqueda de la aplicación que quiera ejecutar. También se debe tener en cuenta, que las personas mayores suelen tener algunos pequeños problemas de visión que dificulta el visionado de la pantalla sin gafas. En estas ocasiones, al disponer la aplicación de un widget grande, se facilita el lanzamiento de la aplicación.
- **Usuarios con alguna discapacidad visual o motora**, que se apoyan en lectores de pantalla para el uso de dispositivos móviles. En esta ocasión, los lectores de pantalla leen uno a uno los componentes de la pantalla que el usuario va tocando. La aplicación TAPADÚ facilitaría el punto de partida a otras aplicaciones, evitando que tenga que recorrer toda la lista de componentes hasta dar con el icono que quiere iniciar.

En las fases de diseño y desarrollo, se han tenido en cuenta los grupos de usuarios descritos, por lo que se ha podido adaptar la aplicación a cada una de las diferentes necesidades que cada uno de los grupos requiere, especialmente el aspecto de la usabilidad para las personas mayores y la accesibilidad para personas con discapacidades visuales o motoras.

3.2 Casos de uso: Aplicación Android

La aplicación permite la creación y gestión de las rutinas. La interfaz gráfica debe ser sencilla y cumplir con las guías de accesibilidad mencionadas en el apartado 2.1.

Las acciones que se pueden realizar la aplicación Android son:

- Crear rutina.
- Consultar rutinas.
- Modificar rutina.
- Eliminar rutina.
- Subir el catálogo de rutinas.
- Descargar catálogo de rutinas
- Realizar registro
- Modificar las opciones de la aplicación.

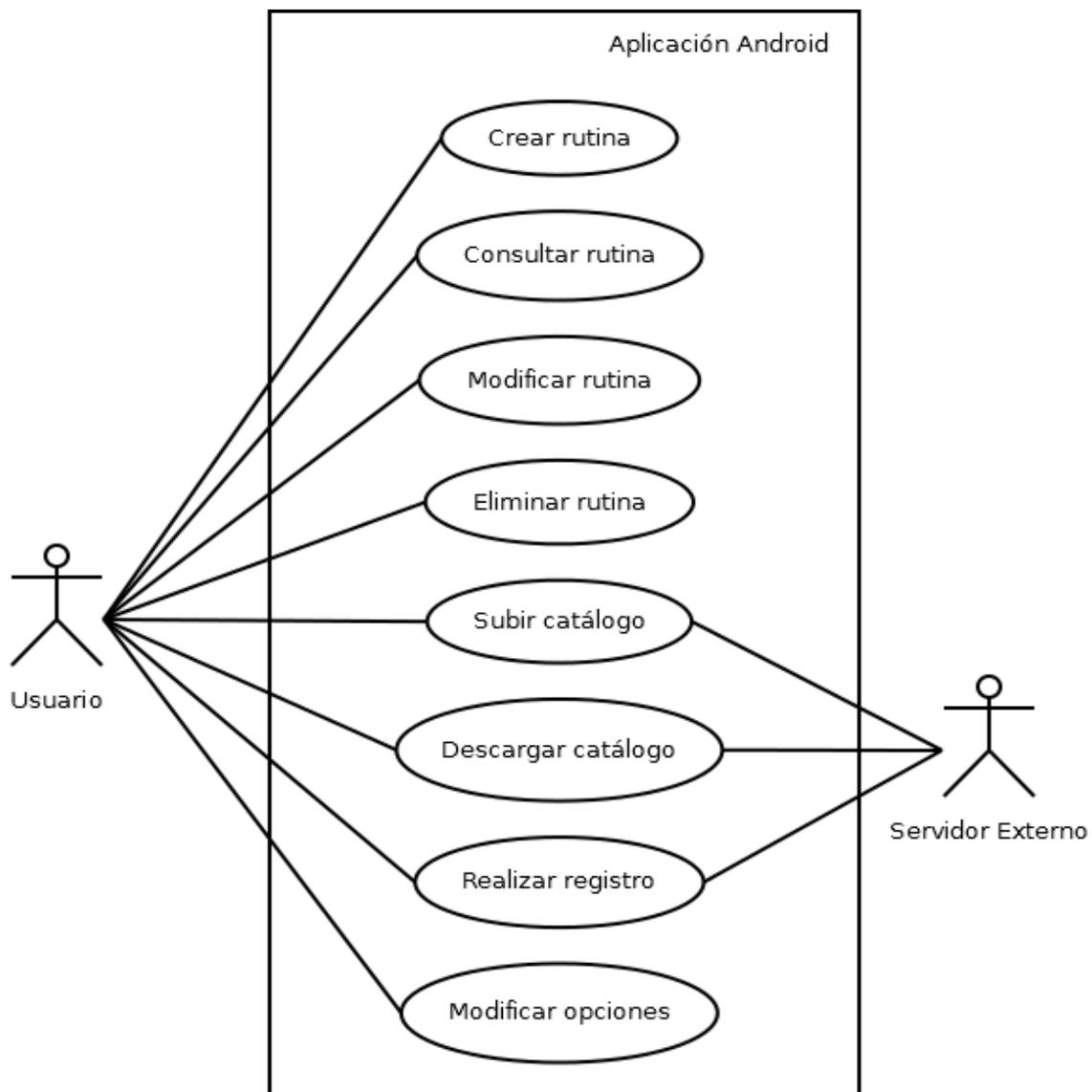


Ilustración 3. Casos de uso de la aplicación Android

3.2.1 Crear rutina

Especificación de alto nivel

Caso de uso:	Crear rutina
Actores:	Usuario (principal e iniciador)
Propósito:	Asignar una aplicación a una palabra clave
Descripción:	El usuario selecciona la opción "Crear nueva rutina", selecciona la aplicación y le asigna una palabra clave.

Especificación expandida

Curso típico de los acontecimientos	Acciones del actor	Respuestas del sistema
	1. El usuario ejecuta la aplicación pulsando en el icono.	2. El sistema se abre y muestra las opciones disponibles.
	3. El usuario selecciona la opción "Nueva rutina".	4. El sistema muestra todas las aplicaciones instaladas en el dispositivo.
	5. El usuario elige la aplicación de la lista mostrada.	6. El sistema pide la palabra clave.
	7. El usuario dice la palabra clave	8. El sistema almacena la pareja aplicación, palabra clave seleccionada. Vuelve al menú principal.
Cursos alternativos:		
7a El usuario escoge una palabra clave que ya está asignada o es parte de una.		
1.	El sistema informa que debe escoger una nueva palabra clave.	
Req. especiales:	Es importante destacar que la palabra clave debe ser única, y puede estar compuesta por más de una palabra. Además, para un reconocimiento de voz óptimo tiene que haber una distancia adecuada entre el usuario y el micrófono, así como un ruido ambiental que no sea excesivo.	

3.2.2 Consultar rutinas

Especificación de alto nivel

Caso de uso:	Consultar rutina
Actores:	Usuario (principal e iniciador)
Propósito:	Consultar las rutinas que ya están instaladas en el dispositivo.
Descripción:	El usuario accede a la opción "Consultar rutinas". Pulsa sobre una de las rutinas del catálogo para acceder al detalle.

Especificación expandida

Curso típico de los acontecimientos	Acciones del actor	Respuestas del sistema
	1. El usuario ejecuta la aplicación pulsando en el icono.	2. El sistema se abre y muestra las opciones disponibles.
	3. El usuario selecciona la opción "Consultar rutinas".	4. El sistema muestra todas las rutinas configuradas.
	5. El usuario elige una rutina de la lista mostrada.	6. El sistema muestra el detalle de la rutina.
Cursos alternativos:		
4a No hay rutinas dadas de alta en el catálogo.		
1.	El sistema muestra la lista vacía con un mensaje que indica que no hay rutinas para mostrar.	
Req. especiales:	No existen	

3.2.3 Modificar rutinas

Especificación de alto nivel

Caso de uso:	Modificar rutina
Actores:	Usuario (principal e iniciador)
Propósito:	Modificar la palabra clave asociada a una rutina.
Descripción:	El usuario accede a la opción "Consultar rutinas". Pulsa sobre una de las rutinas del catálogo para acceder al detalle y pulsa sobre modificar. Modifica la palabra clave y guarda la modificación.

Especificación expandida

Curso típico de los acontecimientos

Acciones del actor

1. El usuario ejecuta la aplicación pulsando en el icono.

3. El usuario selecciona la opción "Consultar rutinas".

5. El usuario elige una rutina de la lista mostrada.

7. El usuario pulsa modificar.

9. El usuario dice la nueva palabra clave y pulsa sobre guardar.

Respuestas del sistema

2. El sistema se abre y muestra las opciones disponibles.

4. El sistema muestra todas las rutinas configuradas.

6. El sistema muestra el detalle de la rutina.

8. El sistema pide la nueva palabra clave.

10. El sistema almacena la pareja aplicación, palabra clave. Vuelve al menú principal.

Cursos alternativos:

4a No hay rutinas dadas de alta en el catálogo.

1. El sistema muestra la lista vacía con un mensaje que indica que no hay rutinas para mostrar.

8a El usuario escoge una palabra clave que ya está asignada o es parte de una.

1. El sistema informa que debe escoger una nueva palabra clave.

Req. especiales:

Es importante destacar que la palabra clave debe ser única, y puede estar compuesta por más de una palabra

3.2.4 Eliminar rutinas

Especificación de alto nivel

Caso de uso:	Eliminar rutina
Actores:	Usuario (principal e iniciador)
Propósito:	Eliminar la rutina del catálogo.
Descripción:	El usuario accede a la opción "Consultar rutinas". Pulsa sobre una de las rutinas del catálogo para acceder al detalle y pulsa sobre eliminar.

Especificación expandida

Curso típico de los acontecimientos

Acciones del actor

Respuestas del sistema

1. El usuario ejecuta la aplicación pulsando en el icono.

2. El sistema se abre y muestra las opciones disponibles.

3. El usuario selecciona la opción "Consultar rutinas".

4. El sistema muestra todas las rutinas configuradas.

5. El usuario elige una rutina de la lista mostrada.

6. El sistema muestra el detalle de la rutina.

7. El usuario pulsa eliminar.

8. El sistema elimina la rutina y vuelve al menú principal.

Cursos alternativos:

4a No hay rutinas dadas de alta en el catálogo.

1. El sistema muestra la lista vacía con un mensaje que indica que no hay rutinas para mostrar.

Req. especiales: No existen

3.2.5 Subir el catálogo al servidor

Especificación de alto nivel

Caso de uso:	Guardar catálogo
Actores:	Usuario (principal e iniciador), Servidor externo (actor de soporte)
Propósito:	Guardar el catálogo en un servidor externo.
Descripción:	El usuario accede a la opción "Guardar catálogo".

Especificación expandida

Curso típico de los acontecimientos	Acciones del actor	Respuestas del sistema
	1. El usuario ejecuta la aplicación pulsando en el icono.	2. El sistema se abre y muestra las opciones disponibles.
	3. El usuario selecciona la opción "Guardar catálogo".	4. El sistema pide los datos para loguearse.
	5. El usuario escribe los datos	6. El sistema envía petición al servicio web para guardar el catálogo en el servidor e informa al usuario de que todo ha funcionado correctamente.

Cursos alternativos:

5a Si el usuario no se ha registrado en la aplicación

1. En tal caso, deberá pulsar sobre registro y realizar el registro (caso de uso 3.2.7).

5b Los datos de acceso son incorrectos.

1. El sistema informa que los datos no son correctos y ofrece la opción de registrarse.

6a No hay Internet

1. El sistema informa que no existe conexión con Internet.

Req. especiales:	Se necesita conexión a Internet. Seguridad: acceso restringido a los usuarios registrados
-------------------------	--

3.2.6 Descargar catálogo

Especificación de alto nivel

Caso de uso:	Descargar catálogo
Actores:	Usuario (principal e iniciador), Servidor externo (actor de soporte)
Propósito:	Descargar el catálogo desde un servidor externo.
Descripción:	El usuario accede a la opción “Descargar catálogo”.

Especificación expandida

Curso típico de los acontecimientos	Acciones del actor	Respuestas del sistema
	1. El usuario ejecuta la aplicación pulsando en el icono.	2. El sistema se abre y muestra las opciones disponibles.
	3. El usuario selecciona la opción “Descargar catálogo”.	4. El sistema pide los datos para loguearse.
	5. El usuario escribe los datos	6. El sistema envía petición al servicio web para recuperar el catálogo en el servidor y guarda las rutinas.

Cursos alternativos:

5a Si el usuario no se ha registrado en la aplicación

1. En tal caso, deberá pulsar sobre registro y realizar el registro (caso de uso 3.2.7).

5b Los datos de acceso son incorrectos.

1. El sistema informa que los datos no son correctos y ofrece la opción de registrarse.

6a No hay Internet

1. El sistema informa que no existe conexión con Internet.

6b No hay un catálogo subido previamente

1. El sistema informa que no se ha subido un catálogo previamente.

Req. especiales:	Se necesita conexión a Internet. Seguridad: acceso restringido a los usuarios registrados
-------------------------	--

3.2.7 Realizar registro

Especificación de alto nivel

Caso de uso:	Realizar registro
Actores:	Usuario (principal e iniciador), Servidor externo (actor de soporte)
Propósito:	Registrarse en el servicio web externo para poder subir y descargar el catálogo de rutinas.
Descripción:	El usuario accede a la opción "Registro" del menú "Guardar catálogo" o "descargar catálogo"

Especificación expandida

Curso típico de los acontecimientos	Acciones del actor	Respuestas del sistema
	1. El usuario pulsa sobre el botón registro.	2. El sistema pide un nombre de usuario y una contraseña que deberá repetir.
	3. El usuario escribe los campos y pulsa en aceptar.	4. El sistema guarda los nuevos datos del registro.

Cursos alternativos:

3a El nombre de usuario escogido ya se está usando

1. El sistema muestra un aviso indicando que ese nombre ya está en uso y que escoja uno nuevo.

3b Las contraseñas no tiene el mínimo de 6 caracteres.

1. El sistema informa que la contraseña debe tener un mínimo de 6 caracteres.

3c Las contraseñas no coinciden.

1. El sistema informa que las contraseñas no coinciden.

3d No hay Internet

1. El sistema informa que no existe conexión con Internet.

Req. especiales: Se necesita conexión a Internet.

3.2.8 Modificar opciones

Especificación de alto nivel

Caso de uso:	Modificar opciones
Actores:	Usuario (iniciador)
Propósito:	Modificar las opciones de la aplicación.
Descripción:	El usuario accede a la opción "Opciones".

Especificación expandida

Curso típico de los acontecimientos	Acciones del actor	Respuestas del sistema
	1. El usuario ejecuta la aplicación pulsando en el icono.	2. El sistema se abre y muestra las opciones disponibles.
	3. El usuario selecciona la opción "Opciones".	4. El sistema muestra las opciones disponibles.
	5. El usuario modifica las opciones.	6. Al modificarse las opciones el sistema guarda automáticamente los cambios.
Cursos alternativos:	No existen	
Req. especiales:	No existen	

3.3 Casos de uso: Widget

El Widget tiene el objetivo de ejecutar la aplicación tras haber detectado la palabra clave. Se ha decidido desarrollar un Widget debido a las siguientes razones:

- A pesar de que los móviles han realizado grandes avances, su punto débil sigue siendo la poca duración de la batería. Tener el móvil "escuchando" las 24 horas a la espera de detectar una palabra clave para lanzar una aplicación, reduciría aún más la vida de la batería. Al desarrollarlo en forma de widget, es el usuario quien decide iniciar la aplicación, optimizando los recursos.
- Al elaborar un widget, se puede seleccionar el tamaño, colores y formas que facilite más su ejecución que un simple icono.

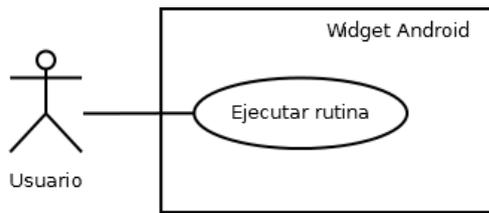


Ilustración 4. Casos de uso del widget Android

3.3.1 Ejecutar rutina

Especificación de alto nivel

Caso de uso:	Ejecutar rutina
Actores:	Usuario (principal e iniciador)
Propósito:	Ejecutar la rutina asociada a la palabra clave
Descripción:	El usuario pulsa sobre el widget y pronuncia la palabra clave.

Especificación expandida

Curso típico de los acontecimientos

Acciones del actor

1. El usuario ejecuta el widget pulsando sobre él.
3. El usuario dice la palabra clave.

Respuestas del sistema

2. El sistema muestra que ahora está escuchando.
4. El sistema procesa la palabra clave y ejecuta a aplicación asociada.

Cursos alternativos:

3a El usuario no dice una palabra clave.

1. El sistema informa que no ha sido capaz de procesar la palabra clave

4a El sistema no es capaz de procesar la palabra clave

1. El sistema informa que no ha sido capaz de procesar la palabra clave

4b No existe una aplicación asociada a la palabra clave

1. El sistema informa que no encontrado la rutina

Req. especiales: Se necesita el reconocimiento de Google Voice.
Conexión a Internet.

4 Diseño del sistema

4.1 Tecnologías a usar

4.1.1 Aplicación Android y Widget

Versiones de Android

Android tiene actualmente seis versiones activas posteriores a Android 2.2 que se distribuyen en los diferentes dispositivos móviles, tal y como muestra la ilustración 3:

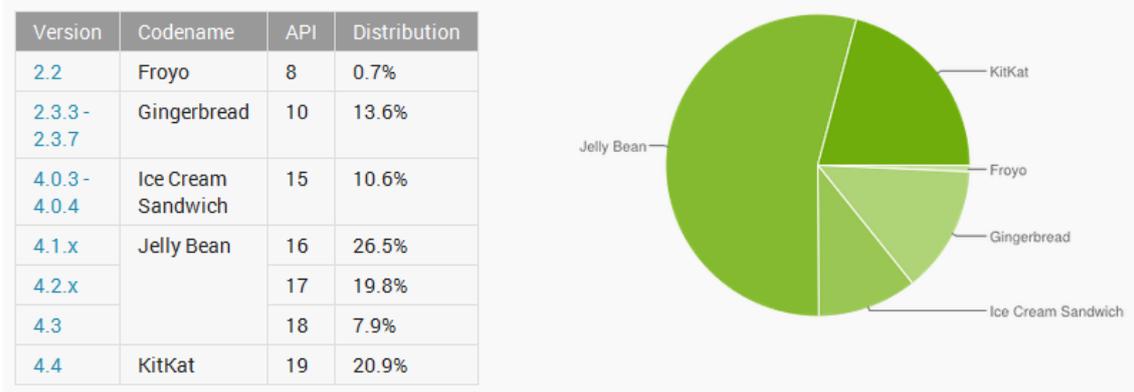


Ilustración 5. Distribución de las versiones en Android en agosto 2013.

Para entender mejor esta gráfica se tiene que tener en cuenta que las anteriores versiones a 2.2 no se contabilizan, que tal y como explican², en agosto de 2013 correspondía a 1%.

Teniendo en cuenta estos datos, se ha decidido que se desarrolle la aplicación pidiendo como mínimo la versión 2.3, lo que margina a un 0.7% de los dispositivos.

SDK

Android SDK es el kit de desarrollo necesario para programar e implementar las aplicaciones para Android. Este kit de desarrollo incluye las APIs y herramientas necesarias para desarrollar las aplicaciones utilizando JAVA como lenguaje de programación y testear el código, respectivamente.

El kit de desarrollo escogido incluye:

- Eclipse + ADT plugin: Entorno de desarrollo con facilidades para el desarrollo de aplicaciones Android.
- Android SDK Tools: Herramienta que facilita las descargas de nuevas versiones, documentación, herramientas...
- Android Platform-tools: Conjunto de herramientas para el desarrollo de aplicaciones Android que incluye el soporte para la debugación en el emulador.
- La última versión de Android platform: la última versión Android publicada
- La última imagen de sistema Android para el emulador: Imagen para poder emular la aplicación en el AVD (Android Virtual Device).

² http://developer.android.com/about/dashboards/index.html?utm_source=ausdroid.net

Reconocimiento de voz

Android no puede realizar el reconocimiento de voz con sus librerías nativas, por consiguiente, un dispositivo Android típico tampoco puede realizar el reconocimiento de voz. La solución más sencilla, es usar otra aplicación que realice el reconocimiento y envíe los resultados obtenido a TAPADÚ. Esta delegación de acciones, se conoce en Android como Intents.

Google Voice Search[15] es una de las aplicaciones más populares de reconocimiento de voz disponibles para Android y es compatible con varios idiomas. Este servicio requiere una conexión con Internet debido a que el reconocimiento de voz se lleva a cabo en los servidores de Google. Esta aplicación tiene una Actividad muy simple que informa a los usuarios que pueden hablar. El momento en que el usuario deja de hablar, se cierra el diálogo y la aplicación que la ha llamado (intent caller) recibe un conjunto de cadenas con el reconocimiento del habla.

Accesibilidad

Tanto en el widget como en la aplicación Android es muy importante que se desarrolle siguiendo las directrices de accesibilidad que presenta Android:

1. Asegurarse de poner descripciones en los controles de interfaz: botones, checkbox etc.
2. Navegación por foco accesible, lo que significa, que se permita navegar con diferentes hardwares: teclados, trackball, d-pad etc.
3. Si se diseñan controles de interfaz personalizados, se deben implementar siguiendo las directrices de accesibilidad: proporcionando una descripción de contenido, implementar métodos API de accesibilidad etc.
4. No se puede poner como único feedback el audio, ya que las personas con problemas de audición, no lo recibirá correctamente.
5. Realizar pruebas de la aplicación usando controles direccionales, sin mirar usando Talkback y siguiendo las directrices del checklist de accesibilidad que ofrece Android.

En el Anexo C se detalla mejor las guías y checklists que se han usado para asegurar una correcta accesibilidad de la aplicación.

4.1.2 Aplicación web

Patrón y Framework

La aplicación web ha sido diseñada siguiendo el patrón Modelo Vista Controlador (cuya explicación breve se encuentra en el Anexo A). Además se ha desarrollado mediante el lenguaje JAVA (debido a los conocimientos que se poseen) y el framework Struts 2[16].

La justificación del uso del patrón MVC es que sus ventajas se corresponden con los puntos críticos de la aplicación:

- Escalabilidad: La principal ventaja de separar la aplicación en tres capas reales es la escalabilidad, permitiendo ampliar y modificar las funcionalidades con un cambio mínimo.
- Facilidad de mantenimiento: el seguir métodos comunes de programación, hace que el código sea más entendible, permitiendo continuar el trabajo por otros.

- Abstracción de datos: Se puede utilizar la abstracción de datos usando un mapeo objeto-relacional (más conocido por su nombre en inglés, Object-Relational mapping, o sus siglas ORM como Hibernate[17]) para Java, facilitando la realización de consultas a la base de datos.
- Facilidades: Los frameworks están creados para facilitar el trabajo de los desarrolladores: dispone clases para controlar fechas, URL's, servicios web, etc.

La justificación de haber elegido JAVA junto a Struts 2 se debe a los siguientes puntos:

- La experiencia previa en desarrollo de aplicaciones en JAVA y el uso del framework Struts 2.
- La separación de los componentes en capas (MVC) que simplifica notablemente el desarrollo y su mantenimiento.

Por consiguiente, al usar un framework conocido como Struts 2 y aplicar el patrón MVC, se ha conseguido una aplicación sencilla, escalable y que facilita la búsqueda de soluciones ante los posibles problemas que vayan surgiendo.

Base de datos

Las opciones que se barajan para la base de datos que almacene las rutinas son: PostgreSQL[18] y MySQL[19]. Sus pros y contras son:

PostgreSQL

- ✓ Posee una gran escalabilidad. Es capaz de ajustarse al número de CPUs y a la cantidad de memoria que posee el sistema de forma óptima, haciéndole capaz de soportar una mayor cantidad de peticiones simultáneas de manera correcta.
- ✓ Implementa el uso de *rollback's*, subconsultas y transacciones, haciendo su funcionamiento mucho más eficaz.
- ✓ Tiene la capacidad de comprobar la integridad referencial, y almacenar procedimientos en la propia base de datos.
- × Consume gran cantidad de recursos.
- × Tiene un límite de 8K por fila, aunque se puede aumentar a 32K, con una disminución considerable del rendimiento.
- × Es de 2 a 3 veces más lento que MySQL.

MySQL

- ✓ Su mayor ventaja es la velocidad a la hora de realizar las operaciones, lo que le hace uno de los gestores que ofrecen mayor rendimiento.
- ✓ Su bajo consumo lo hacen apto para ser ejecutado en una máquina con escasos recursos sin ningún problema.
- ✓ Las utilidades de administración de este gestor son envidiables para muchos de los gestores comerciales existentes, debido a su gran facilidad de configuración e instalación.
- ✓ Tiene una probabilidad muy reducida de corromper los datos, incluso en los casos en los que los errores no se produzcan en el propio gestor, sino en el sistema en el que está.
- ✓ Es el gestor más habitual en los servidores.

- × Carece de soporte para transacciones, *rollback's* y subconsultas.
- × No es viable para su uso con grandes bases de datos, a las que se acceda continuamente, ya que no implementa una buena escalabilidad.

Conclusión

Teniendo en cuenta los pros y contra presentados, se ha decidido usar MySQL debido a su mayor rapidez, pero sobre todo, debido a que es el gestor más habitual en los servidores web. Además, los contras son asumibles en el proyecto y no parecen ser críticos.

Finalmente, para gestionar la base de datos desde el código, tal y como se ha mencionado anteriormente, se ha usado un ORM. Un ORM es una técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y la utilización de una base de datos relacional, utilizando un motor de persistencia. El motor escogido es Hibernate por la experiencia previa que se tiene con él.

4.1.3 Comunicación entre Android y Aplicación web

Como ya se ha definido en el análisis, es necesaria una aplicación web para administrar los catálogos de rutinas que suban los usuarios. Para desarrollar la comunicación entre Android y la aplicación web se ha pensado en dos opciones:

- a) Que la aplicación acceda directamente a la base de datos.
- b) Que la aplicación Android se comunique con la base de datos a través de un servicio web.

De estas dos opciones, se ha decidido implementar el servicio web por las siguientes ventajas:

- La carga del catálogo por parte de la aplicación de Android, implica que se construya dicho catálogo para que pueda mostrarse al usuario. Al hacerlo a través de un servicio web, quien realiza el procesamiento es el servidor donde se aloja la aplicación web, que es mucho más potente que el dispositivo donde está instalada la aplicación.
- El proceso de subir catálogos de rutinas, también recae sobre el servidor, ya que la aplicación Android sólo debe preocuparse de pasarle los datos al servicio web.

Finalmente, sólo queda decidir cómo construir el catálogo. Para ello, se han pensado en dos lenguajes: XML[20] (Siglas en inglés de *eXtensible Markup Language* - lenguaje de marcas extensible) o JSON[21] (Acrónimo en inglés de *JavaScript Object Notation* – Notación de Objetos de JavaScript). Se ha decidido hacerlo mediante JSON por las siguientes razones:

- Al igual que XML puede ser leído fácilmente y es independiente de la plataforma.
- JSON es capaz de representar la misma información en un formato más ligero que XML y por lo tanto, es más rápido de transportar y consume menos ancho de banda (factor muy importante en la opción de subida/descarga del catálogo de rutinas).

Dado que ya se ha elegido Struts 2 para el desarrollo de la aplicación web, se creará el servicio web usando el plugin de REST[22] para facilitar su integración en la aplicación.

4.2 Diseño conceptual

La ilustración 4 muestra el diseño conceptual de la aplicación:

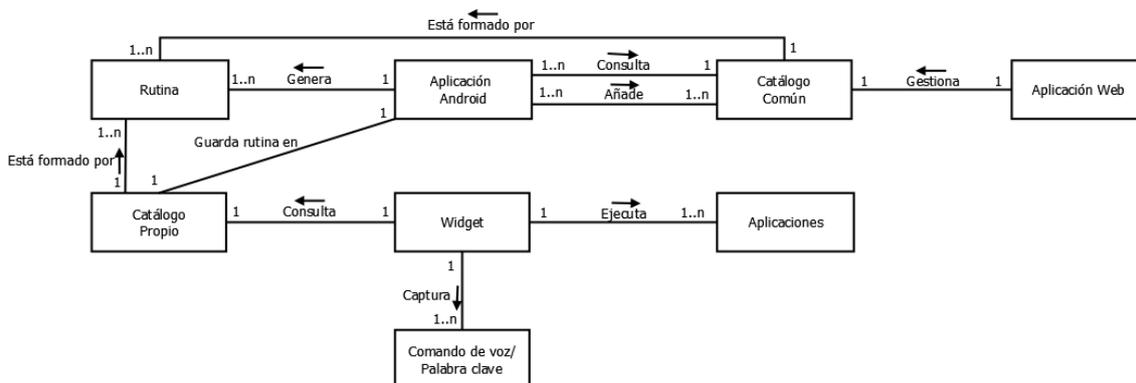


Ilustración 6. Diagrama conceptual

Los diagramas de colaboración, clases y base de datos, se encuentran en el Anexo B.

5 Prototipo

Para diseñar el prototipo, se ha usado la herramienta Fluid UI[23]. Esta herramienta está pensada específicamente para desarrollo de aplicaciones móviles y ofrece un extenso catálogo para iOS, Android y Windows 8.

Esta aplicación, basada en navegador Web, y por tanto multiplataforma, exporta y genera los prototipos en HTML5.

Pero la razón por la que ha sido escogida, es la facilidad de descarga para poder usar el prototipo en el dispositivo. Tras escanear el código QR, se permite navegar a través de la aplicación.

A continuación se puede visualizar el esquema general de las aplicaciones. A cada esquema, le acompaña un código QR que tras su escaneo, permite interactuar con el prototipo en el dispositivo móvil o bien al pulsar sobre él, se accede al prototipo a través de la web.

5.1 Aplicación Android

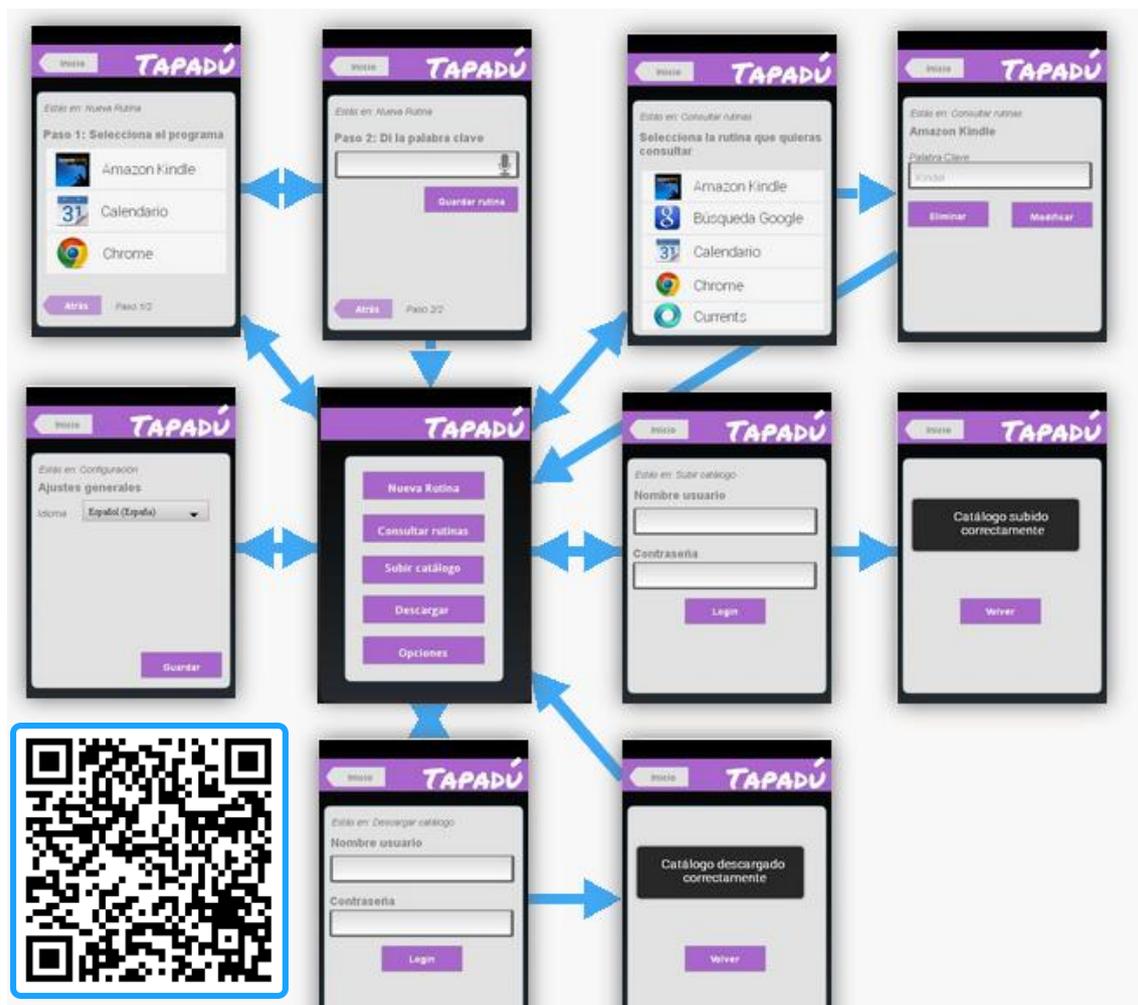


Ilustración 7. Esquema general del prototipo de la aplicación Android

5.2 Widget lanzador



Ilustración 8. Esquema general del prototipo del widget

6 Evaluación de accesibilidad y pruebas con usuarios

6.1 Evaluación de accesibilidad

Actualmente, no existe ninguna normativa específica, nacional o internacional, para el desarrollo de aplicaciones accesibles en dispositivos móviles, aunque sí para el desarrollo software: UNE 139802:2003 [24]

Es por ello, que Android ofrece a sus desarrolladores varias recomendaciones y sugerencias en cuanto al diseño, desarrollo y testeo de aplicaciones, para asegurar un mínimo de accesibilidad [25].

Para realizar la evaluación de accesibilidad de la aplicación TAPADÚ, se han recogido tanto las recomendaciones de Android como la UNE 139082:2003 (extrapolándolas y aplicando sólo aquellas que se corresponda a aplicaciones móviles), para asegurar así que las recomendaciones de accesibilidad básicas se cumplen. Este método de evaluación, se podría considerar una inspección de estándares, pero a falta de tener una normativa oficial que seguir, se tienen en cuenta las recomendaciones recogidas en el anexo C.

6.1.1 Resultados

A continuación se detallan los resultados de la evaluación siguiendo los 10 apartados de recomendaciones descritos en el anexo C. Dicha evaluación se considera muy informal, ya que quien la ha realizado, es la misma persona que ha diseñado y desarrollado la aplicación.

A continuación se listan las pautas que han fallado en la aplicación TAPADÚ clasificadas por prioridad, en el anexo C se encuentran los errores detallados y cómo se han solucionado.

Prioridad 1

Recomendaciones generales

- Recomendación 1.3: Se debe poder manejar el software de forma efectiva utilizando sólo uno de los posibles dispositivos de entrada. Esto implica que la aplicación puede ser navegable sin hacer uso de la pantalla táctil.
- Recomendación 1.5.a: Verificar que todos los componentes de la interfaz, ofrecen información clara al usuario, al ser descritos por el lector de pantalla Talkback.

Recomendaciones sonidos y multimedia

- Recomendación 5.5: Asegurarse que no se duplica información, lo que provocaría que el lector de pantalla repitiera información.

Recomendaciones sobre la información de objetos

- Recomendación 7.1.d: Controles que cambian de función hay que garantizar que la información que devuelve el control también ha cambiado apropiadamente dando una información de accesibilidad correcta.

Recomendaciones sobre tiempo

- Recomendación 8.2: La información sobre errores o los avisos relevantes para la tarea actual deben persistir hasta que el usuario confirme su lectura.

Otras recomendaciones

- Recomendación 10.1: Las aplicaciones deben ofrecer la opción de finalizar.

Prioridad 2

Recomendaciones sobre la pantalla

- Recomendación 4.10: Deben proporcionarse combinaciones de colores predefinidas que hayan sido diseñadas teniendo en cuenta las necesidades de las personas con deficiencias visuales.

Recomendaciones sobre la información de objetos

- Recomendación 7.3: El foco de entrada debe quedar reflejado en pantalla de forma inequívoca.

Actualmente se han corregido todos estos puntos en la aplicación.

6.2 Pruebas con usuarios

6.2.1 Propósito y objetivo del test

Uno de los objetivos que se planteaba en el proyecto era crear una aplicación que tuviera como público objetivo cualquier usuario de Smartphone Android incluyendo personas mayores y personas con discapacidades visuales o motoras. Para que este objetivo se cumpla, se ha diseñado y desarrollado la aplicación teniendo en cuenta las recomendaciones de accesibilidad y principios de usabilidad. Pero para testear que se ha cumplido con ello, se debe probar ante usuarios reales y obtener de ellos posibles problemas y mejoras en la aplicación.

La prueba con usuarios que se plantea, tiene por objetivo:

- Comprobar que se comprende la funcionalidad de los diferentes componentes de la aplicación.
- Comprobar que las tareas básicas (ejecución y creación de rutinas) se comprenden y se realizan sin dificultades.
- Detección de problemas y mejoras en la aplicación.
- Recoger las impresiones de los usuarios mediante el método "Think aloud" [27] durante el desarrollo de las pruebas y al final del test.

Para ello, se analizan los resultados teniendo en cuenta:

- **Efectividad**: Identificando si los usuarios han conseguido completar las tareas y qué problemas ha tenido.
- **Eficiencia**: Computando el tiempo que tardan los participantes en realizar las tareas.
- **Satisfacción**: Analizando el resultado del cuestionario final y la actitud y percepciones durante la prueba

A continuación se describen los perfiles, tareas y el procedimiento seguido. En el anexo D se encuentran los diferentes cuestionarios usados durante el test, el guion seguido por el facilitador y el documento de consentimiento para ser grabado. Finalmente en el anexo E se encuentra el detalle de los resultados del test analizado por tareas y perfiles

6.2.2 Participantes

Teniendo en cuenta el público objetivo descrito en el apartado 3.1.4, se han escogido un total de 8 participantes que se han encargado de realizar 3 tareas en la aplicación. Los participantes escogidos poseen y usan un Smartphone y se pueden dividir en los siguientes perfiles:

- **Perfil 1:** Persona joven entre 20-50 años. Usa el Smartphone de forma habitual y un manejo avanzado de tecnologías.
- **Perfil 2:** Persona mayor de más de 50 años. Usa el Smartphone de forma habitual y un uso esporádico de tecnologías.

A continuación se muestran las características de cada participante:

	Género	Edad	Horas uso Smartphone	Experiencia en Smartphone	Perfil
Participante 1	Masculino	20 – 29 años	< 1 hora	3 – 4 años	Perfil 1
Participante 2	Femenino	> 50 años	< 1 hora	< 1 año	Perfil 2
Participante 3	Femenino	20 – 29 años	1 – 2 horas	3 – 4 años	Perfil 1
Participante 4	Masculino	30 – 39 años	> 3 horas	3 – 4 años	Perfil 1
Participante 5	Masculino	> 50 años	> 3 horas	3 – 4 años	Perfil 2
Participante 6	Femenino	30 – 39 años	1 – 2 horas	3 – 4 años	Perfil 1
Participante 7	Femenino	> 50 años	< 1 hora	1 – 2 años	Perfil 2
Participante 8	Masculino	> 50 años	1 – 2 horas	3 – 4 años	Perfil 2

Es posible que, teniendo en cuenta el público objetivo descrito en el apartado 3.1.4, se eche en falta un tercer perfil que incluya usuarios con discapacidad visual que usen lectores de pantalla para interactuar con el dispositivo. En un primer momento, se creó este tercer perfil, pero ante la imposibilidad de haber encontrado a usuarios de este perfil que usen dispositivos Android (como se verá en el apartado 6.3, sólo se localizaron usuarios de iOS) y la falta de tiempo, finalmente se ha tenido que suprimir.

6.2.3 Tareas

El test consta de 3 tareas que están ordenadas por complejidad, de la más sencilla a la más compleja:

- Lanzar una aplicación a través de una palabra clave.
- Intentar lanzar una aplicación con una palabra clave que no está asociada a ninguna aplicación.
- Crear una nueva rutina.

Tarea 1: Lanzar una aplicación con éxito

Esta es la tarea más sencilla. El usuario debe decir la palabra clave y la aplicación debe saber procesarla y buscar la aplicación que tiene asignada. Al usuario se le plantea el siguiente escenario:

TAPADÚ: un lanzador de aplicaciones Android por voz

Evaluación de accesibilidad y pruebas con usuarios

Estás en el supermercado haciendo la compra y no recuerdas todos los productos que tenías que comprar. Antes de salir de casa, has apuntado la lista en una aplicación del móvil y ahora quieres consultarla.

Tienes asociada la palabra clave “notas” a la aplicación donde has apuntado la lista de compras. Usa TAPADÚ para lanzar la aplicación.

La tarea se da por finalizada correctamente cuando el usuario ha dicho la palabra clave, pulsando el widget de la aplicación.

Tarea 2: Intentar lanzar una aplicación sin éxito

Se ha escogido una tarea en la que el resultado no sea el que espera el usuario, para poder medir si la retroalimentación del error es correcta y el usuario la entiende. Al usuario se le plantea el siguiente escenario:

Estás en casa y escuchas una canción en la tele que te gusta. Deseas saber cómo se llama y decides usar la aplicación Shazam para ello.

Usando TAPADÚ di la palabra clave “música”.

La tarea se da por finalizada correctamente cuando el usuario ha dicho la palabra clave proporcionada (la cual es incorrecta) y comprende que no hay una aplicación asociada a la palabra clave.

Tarea 3: Crear una nueva rutina

Esta es la tarea más compleja del test, pero es básica para el correcto funcionamiento de la aplicación. El usuario debe asignar una palabra clave a una aplicación y lanzarla a través del widget. Al usuario se le plantea el siguiente escenario:

Acabas de descargarte una nueva aplicación llamada “IMDB” y quieres asignarle una palabra clave (a tu elección) a la aplicación para que pueda ser lanzada a través de TAPADÚ.

Cuando termines, prueba a lanzar la aplicación a través de TAPADÚ.

La tarea se da por finalizada correctamente cuando el usuario ha generado la nueva rutina y la ha conseguido lanzar a través del widget.

6.2.4 Entorno del test

Se ha planteado realizar un test de campo, en un entorno doméstico con la intención de obtener una interacción por parte del usuario más natural que si se hubiera realizado en un laboratorio de usabilidad. De esta forma, se ha facilitado conseguir participantes que se adecuen a los perfiles buscados, además de usar la aplicación en entornos más reales, como en una cafetería y probar si el ruido ambiental afecta a la escucha.

Dispositivo utilizado

Se ha utilizado un dispositivo con las siguientes características para las pruebas:

Smartphone	Samsung Galaxy Nexus (GT-I9250)
Aplicaciones	TAPADÚ con su widget. Las aplicaciones usadas durante las pruebas: Keep, Shazam, iMDB.
Pantalla	4.6''
Resolución	16 M Color, sAMOLED (1,280 x 720)
Sistema operativo	Android 4.3 Jelly Bean
Lector de pantalla	Talkback 3.5.1

Herramientas del administrador

Se ha decidido realizar el test apoyándose en la aplicación Morae Recorder 3.3.1 para poder grabar la prueba y facilitar así el análisis posterior. Se ha configurado para la prueba el pre-test, las tareas a realizar, los cuestionarios post-tarea y el cuestionario SUS como post-test. De este modo ha sido Morae quien ha conducido la prueba, mostrando los cuestionarios y tareas en el orden establecido.

El montaje de la prueba ha consistido en hacer uso de un portátil con cámara web, y mientras el usuario interactuaba con el dispositivo móvil, se realizaba la grabación de la prueba a través de la cámara web del portátil que tenía instalado MORAE. Así pues, el usuario ha realizado los cuestionarios sobre el portátil y las tareas sobre el dispositivo.

Las grabaciones de video sólo muestran a los usuarios realizando las pruebas, y no se muestra la pantalla del dispositivo en ningún momento. Esto es debido a las siguientes causas:

- La primera opción que se sopesó, fue realizar un screenCast del dispositivo, para que MORAE pudiera grabar la pantalla del móvil que se visualizaba en la pantalla del ordenador. Esta idea se descartó tras usar varias aplicaciones y comprobar que con todas ellas, el lag producido era tan grande que impedían la interacción natural con el dispositivo.
- La segunda opción que se sopesó, fue usar un cámara externa pero al realizar el test en un entorno doméstico (hogar, cafetería, trabajo...) no se ha dispuesto de una cámara externa sobre la que apuntar hacia la pantalla del dispositivo. Además, de haberlo hecho se hubiera cohibido el uso natural del móvil, ya que tendrían que preocuparse de que la cámara enfocara la pantalla en todo momento. De este modo, cuando en alguna ocasión el usuario al coger el móvil, ha tapado el micrófono apareciendo el consiguiente error de que no se ha reconocido la palabra clave, se ha podido analizar si los usuarios tienden a acercarse el móvil al realizar el reconocimiento de voz...

TAPADÚ: un lanzador de aplicaciones Android por voz

Evaluación de accesibilidad y pruebas con usuarios

El equipo que se ha usado para hacer el test y sobre el que se ha instalado MORAE para la grabación de las sesiones es el siguiente:

Tipo ordenador	Sony Vaio VPCYB1S1E/S AMD Doble nucleo 1.6GHz 4 GB DDR3
Pantalla	13"
<i>Resolución</i>	1366 x 768
Sistema operativo	Windows 7
Periféricos	Ratón inalámbrico
Cámara	Sony Visual Communication Camera (320x240)

Para el análisis de los resultados, se ha utilizado el software Morae Manager 3.3.1 con el que se ha podido analizar los parámetros como duración de las tareas, resultados recogidos por los cuestionarios, etc.

Procedimiento seguido

Durante el test de usabilidad, los 8 participantes (4 del perfil 1 y 4 del perfil 2) realizaron las 3 mismas tareas para probar las funcionalidades de la aplicación, durante unos 30 minutos cada sesión, en la cual:

- Completaron un cuestionario sobre sus datos generales
- Realizaron las tareas que se le encomendaron en la aplicación.
- Respondieron a un cuestionario sobre cada tarea realizada.
- Respondieron a un cuestionario sobre la aplicación en general.

En las 3 tareas se han medido los siguientes parámetros: tiempo necesario para realizar la tarea y porcentaje de finalización mediante el análisis de las tareas y grado de complejidad mediante los cuestionarios post-tarea.

No se ha establecido un límite de tiempo para las tareas, sino que el usuario ha decidido cuando terminar la tarea o bien porque creía que había alcanzado el objetivo o bien porque ha decidido dejarlo.

Protocolo seguido por el facilitador

Al llegar el usuario, el facilitador le explica que a continuación va a realizar un test de una aplicación en pruebas para detectar fallos en la aplicación y corregirlos. Es muy importante dejarle claro que ante todo se trata de evaluar la aplicación y no sus capacidades.

Se le explica que nos es de gran ayuda grabar la prueba para facilitar el análisis posterior. Para ello es necesario que firme un formulario en el que se dictamina que los datos personales podrán ser utilizados únicamente con finalidades científicas y/o académicas para el análisis de los datos recogidos en el proyecto. Está en su total derecho de negarse a ser grabado sin ninguna penalización.

A continuación se le debe explicar cómo funciona este tipo de test y las etapas de las que se compone: un pre-test, 3 tareas con sus siguientes post-tareas y finalmente un post-test.

Se le explica que el facilitador estará siempre con él, pero que no puede ayudarlo a completar las tareas, aunque sí dudas en las descripciones de las tareas. Tiene que interactuar de la forma más natural posible, que sea sincero y que diga todo lo que piense en voz alta.

Seguidamente, tras asegurarse que el usuario no tiene ninguna duda, se inicia el test con el cuestionario pre-test en el que se le piden los datos generales. Para seguir con las tareas.

Una vez realizado el último cuestionario post-test de la prueba, se realiza un último feedback con el facilitador preguntándole, qué le ha parecido y si tiene algún comentario que desee realizar.

Finalmente se le agradece su ayuda y el facilitador se despide de él.

6.2.5 Métricas de usabilidad

Siguiendo las métricas que recomienda CENATIC (Centro Nacional de referencia de Aplicaciones de las TIC basadas en fuentes abiertas)[29], se establecen:

Métricas de Efectividad

- Porcentaje de finalización: Porcentaje de los participantes que ha llegado correctamente al objetivo de cada tarea. Se distinguen 3 valores: Completado con éxito, completado con errores y no completado.
- Fracasos:
 - Tarea 1: Si el usuario no consigue ejecutar la aplicación tras decir la palabra clave, el reconocimiento de voz no reconoce la voz...
 - Tarea 2: Si el usuario no comprende el error y vuelve a intentarlo repitiendo la palabra clave.
 - Tarea 3: Que no logre dar de alta la rutina porque le resulte compleja la aplicación.

Métricas de Eficiencia

- Tiempo en tarea: Tiempo empleado por el usuario para completar con éxito una tarea.

Métricas de Satisfacción

Estas métricas serán recogidas mediante cuestionarios que son rellenados por el participante.

Se recomienda recurrir a cuestionarios establecidos ya que los resultados apuntan a una mayor fiabilidad que los hechos a medida. Para este test, se propone utilizar el cuestionario SUS[30] (System Usability Scale) que es un cuestionario de 10 elementos desarrollado por John Brooke que proporciona una visión general de la satisfacción con el software.

6.2.6 Resultados

6.2.6.1 Análisis de resultados

A continuación se presentan los resultados de los tests divididos por perfiles junto a las métricas de efectividad, eficiencia y satisfacción recogidas. También se muestran los errores más comunes y las soluciones que se han decidido implementar.

Para las métricas de efectividad se muestra, para cada tarea, qué partes de ésta se han completado correctamente y cuántas personas han fallado en cada una de ellas.

Para calcular los diferentes porcentajes de finalización se tienen en cuenta las diversas etapas de cada tarea:

- T1: Lanzar la aplicación 100%. Si no se consigue 0%.
- T2: Tras decir la palabra clave, comprender que no tiene asociada ninguna aplicación 100%. Si no se entiende el mensaje de error 0%.
- T3: Generar la rutina 75% y ejecutar la aplicación con la rutina 25%. Se resta por cada problema 25%.

Además se tienen en cuenta el grado de problemas detectados:

- 0 ningún problema detectado,
- 1 problemas detectados pero se ha finalizado la tarea
- 2 no se ha finalizado la tarea.

Para las métricas de eficiencia se muestra el tiempo total de tarea. Además en las gráficas se muestran el tiempo medio necesario.

Para las métricas de satisfacción se muestran cómo de complejas les ha parecido la tarea (1 nada y 4 mucho) teniendo en cuenta los datos recogidos de los cuestionarios post-tarea.

6.2.6.2 Resumen ejecutivo de resultados

Resultados por participante

A continuación se muestra una tabla en la que se indica el porcentaje de finalización por participante, teniendo en cuenta el análisis descrito en el apartado 6.2.6.1:

Participante (perfil)	Tarea 1	Tarea 2	Tarea 3
Participante 1 (P1)	100% OK	100% OK	100% OK
Participante 2 (P2)	100% OK	100% OK	50%. KO
Participante 3 (P1)	75%. OK con problemas	100% OK	100% OK
Participante 4 (P1)	100% OK	100% OK	100% OK
Participante 5 (P2)	75%. OK con problemas	100% OK	75%. OK con problemas
Participante 6 (P1)	100% OK	100% OK	100% OK
Participante 7 (P2)	100% OK	100% OK	50%. KO
Participante 8 (P2)	100% OK	100% OK	100% OK

TAPADÚ: un lanzador de aplicaciones Android por voz

Evaluación de accesibilidad y pruebas con usuarios

Resultados combinados por perfil y tarea

A continuación se muestra una tabla con las medias por perfil y tarea:

- El porcentaje de finalización según los criterios descritos en el apartado 6.2.6.1.
- Los problemas detectados según los criterios descritos en el apartado 6.2.6.1.
- El tiempo que ha llevado realizar la tarea.
- La complejidad que han indicado los participantes en los cuestionarios post-tarea.

Tarea	EFECTIVIDAD				EFICIENCIA		SATISFACCIÓN	
	Porcentaje de finalización		Grado problemas		Tiempo tarea		Complejidad	
	P1	P2	P1	P2	P1	P2	P1	P2
Tarea 1	93,75%	93,75%	0,25	0,25	47,75s	60,25s	1,25	1,25
Tarea 2	100%	100%	0	0	18,5s	25,75s	1,25	1,25
Tarea 3	100%	68,75%	0	1,25	65,75s	309,75s	1	1,5
Media	97,91%	87,5%	0,08	0,5	44s	131,91s	1,08	1,33

De los datos expuestos en la tabla, se extrae:

- La **efectividad** en ambos perfiles es alta. El perfil 2 se ha encontrado con más problemas que han reducido el porcentaje de finalización y consecuentemente, el grado de problemas es más alto (0,08 del perfil 1 frente a un 0,5 del perfil 2, en donde 2 es el grado más alto y 0 el más bajo).
- La **eficiencia** del perfil 1 es mucho más alta que la del perfil 2 (44 segundos frente a 131,91 segundos de media por tarea), aproximadamente 4 veces más. Esta gran diferencia se puede explicar por la mayor experiencia que posee el perfil 1, en cuanto al uso de dispositivos móviles, lo que provoca tener un modelo mental más claro, encontrarse con menos problemas y en caso de encontrarlos, darle una solución de forma más rápida.
- La **satisfacción** obtenida en ambos perfiles también es similar (1,08 del perfil 1, frente al 1,33 del perfil 2, teniendo un mínimo de 1 y máximo de 4). La percepción de complejidad es curiosa en el perfil 2, pues mientras se han encontrado con más problemas, el porcentaje de finalización de tareas es más bajo, y la duración más alta, una vez han conseguido finalizar la tarea, les ha parecido que los pasos eran sencillos. Esto podría representar que el factor de aprendizaje y el modelo mental que representa la aplicación es alto.

Finalmente indicar que la media de satisfacción obtenida del cuestionario SUS ha sido de 95 para el perfil 1 y de 78,75 para el perfil 2, es decir, que de promedio es 86,875. Con lo que se considera que los usuarios muestran una satisfacción alta en el uso de la aplicación.

Conclusiones

Las conclusiones obtenidas en las pruebas se estructuran en los diferentes apartados:

- **Éxitos:** Son conceptos de la aplicación que les ha gustado a un alto porcentaje de usuarios.
- **Problemas graves:** Son problemas que deben solucionarse de inmediato, pues son los causantes de que algunas tareas no se hayan conseguido finalizar.
- **Problemas leves:** Son problemas que se deben solucionar pero no son críticos en la finalización de las tareas. No obstante han confundido a algunos usuarios.
- **Sugerencias:** Son las sugerencias que los usuarios han indicado que les gustaría que aparecieran en próximas versiones.

A continuación se enumeran cada una de las conclusiones categorizadas:

Éxitos

7 participantes

Consideran que el número de pasos para terminar la tarea son adecuados.

7 participantes

Consideran que los pasos a seguir son intuitivos.

7 participantes

Consideran que las tareas son sencillas.

Problemas graves

5 participantes

No comprenden la idea de Widget y Aplicación. Confunden ambos componentes.

2 participantes

En la tarea 3, no han detectado que la lista de aplicaciones es una lista que se puede recorrer.

Problemas leves

3 participantes

Confunden la palabra clave y el nombre de la aplicación.

2 participantes

En la tarea 3, decían la palabra clave sin pulsar el botón.

Sugerencias

3 participantes

Han sugerido que la aplicación permita realizar tareas más complejas como: llamar a un contacto, escribir mensajes...

2 participantes

Integrar un buscador en el listado de aplicaciones instaladas y de las rutinas configuradas.

1 participante

Ha sugerido que se pueda personalizar la aplicación con otros colores.

6.3 Interacción con lectores de pantalla

Cuando se realizó el plan de test para las pruebas con usuarios se añadió un tercer perfil para poder probar la aplicación con usuarios que necesitaban de lectores de pantalla para usar los dispositivos móviles.

Para facilitar el reclutamiento, se contactó con el Director de la agencia Administrativa de Lleida de la fundación ONCE y se le explicó el objetivo de la aplicación y la importancia que tendría unas pruebas reales con usuarios de lectores de pantalla.

El Director, estuvo comentando que las aplicaciones en Android que fueran accesibles, no les podían servir, ya que el lector de pantallas que incorporaba Android estaba muy atrás del que incorporaba iOS (conocido como VoiceOver), lo que les obligaba a usar exclusivamente iPhone. Así pues, de todos los afiliados de Lleida en la ONCE, tan sólo había un pequeño grupo de unas seis u ocho personas que usaran Smartphone y todas ellas usaban iPhone. Aun con todo ello, me dio la oportunidad de reunirme con él para hablar sobre el tema de accesibilidad en dispositivos móviles.

Tras esta conversación, se descartó el tercer perfil de usuarios con lector de pantalla debido a que eran usuarios de iOS, y teniendo en cuenta, el trauma que le supone a un usuario sin discapacidad visual el cambio de sistema operativo, un usuario que use un lector de pantalla y se le cambié por otro que no ha manejado nunca anteriormente, no se podrían obtener unos resultados cercanos a la realidad debido a que la situación estaría muy adulterada y el sentimiento de frustración sería muy alto.

Durante la reunión, se le permitió usar el lector de pantalla de Android en un dispositivo móvil y de esta forma se consiguió:

- Por mi parte pude apreciar cómo interactuaba realmente una persona con discapacidad visual (pues había hecho pruebas con el lector de pantalla para probar la accesibilidad en el apartado 6.1 pero lo manejaba como yo creía que funcionaba y condicionada a que yo ya había visto la aplicación).
- Por su parte, comprobó que aunque bien es cierto que la mayoría de aplicaciones de iOS tiene más cuidada la parte de accesibilidad (especialmente en el apartado de lectura de etiquetas), Android ofrecía un lector de pantallas de prestaciones muy similares y que además muchos de los gestos de iOS eran iguales, facilitando así una posible adaptación al lector de pantallas de Android.

También se debe tener en cuenta que estos usuarios, usan determinadas aplicaciones en el dispositivo, pero siempre las más imprescindibles (especialmente porque no todas ofrecen la accesibilidad mínima que se espera) y que además, generalmente es tras la explicación de una tercera persona que le enseña cómo usar la aplicación. En este sentido, TAPADÚ debe cumplir que sea fácil de entender y fácil de usar mediante un lector de pantallas, y estos puntos (aun a falta de hacer pruebas reales con el lector de pantallas) quedan testeados en los apartados 6.1 y 6.2.

TAPADÚ: un lanzador de aplicaciones Android por voz

Evaluación de accesibilidad y pruebas con usuarios

Tras entender cómo se interactúa con la aplicación a partir del lector de pantallas, se desveló un problema con el widget que debe ser solucionado:

- El widget no tiene asociado que pertenece a la aplicación TAPADÚ excepto por el icono que no tiene texto alternativo. Esto provoca que cuando se pasa el dedo encima del widget el lector de pantallas lee “Toca para decir la palabra clave”, pero no dice en ningún momento que pertenece a la aplicación TAPADÚ.

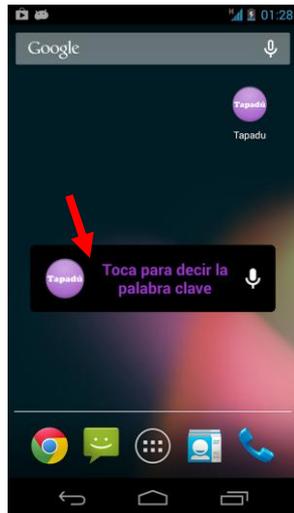


Ilustración 9. El widget no tiene asociado que pertenece a la aplicación TAPADÚ

7 Conclusiones

7.1 Resultados obtenidos

En el apartado 1.6 se establecieron los resultados deseados al finalizar el proyecto. A continuación se discuten uno a uno el grado de consecución de los mismos:

1. Haber desarrollado una aplicación funcional que esté alojada en Google Play

Actualmente, se ha conseguido desarrollar una aplicación Android con las funcionalidades especificadas y alojado en el Google Play.

2. Haber conseguido los objetivos planteados

Los objetivos marcados fueron:

2.1 Diseñar y desarrollar una aplicación que cumpla con los requisitos de accesibilidad implantados por Android: Tras las pruebas realizadas en la aplicación mediante la inspección informal de estándares (recordando, que no existe estándar y que se ha tomado las recomendaciones de Android y la norma UNE 139802:2003), se puede concluir que la aplicación se ha diseñado teniendo en cuenta los requisitos mínimos de accesibilidad.

2.2 Diseñar y desarrollar una aplicación que cumpla con los aspectos de usabilidad: Tras realizar las pruebas con usuarios sobre la aplicación, se han detectado algunos problemas y mejoras que se deberían implementar para mejorar la usabilidad, aunque el alto éxito de las tareas, indica que la usabilidad ha sido cuidada en la aplicación. No obstante, se deberían realizar otras evaluaciones sobre la aplicación para detectar nuevos errores y mejoras.

2.3 Diseñar y desarrollar una aplicación con un público objetivo que incluya personas mayores y personas con discapacidades visuales. Se han realizado pruebas con personas mayores, obteniendo resultados positivos. No obstante, no se han podido realizar pruebas con usuarios de lector de pantalla, con lo que a pesar de cumplir con los requisitos de accesibilidad recomendados por Android, no se puede asegurar qué resultados se habrían obtenido en unas pruebas reales.

2.4 Diseñar y desarrollar una aplicación que se comunique con un servidor externo a través de servicios web: Conseguido a través de servicios web REST alojados en el servidor AWS de Amazon.

2.5 Diseñar y desarrollar una aplicación y widget que utilice reconocimiento de voz como entrada: Conseguido a través de la aplicación Google Voice.

2.6 Distribuir la aplicación a través de Google Play: Actualmente, la aplicación TAPADÚ se puede descargar a través de Google Play.

2.7 La aplicación debe ser sencilla e intuitiva. Cumpliendo con las guías de diseño que recomienda Android y principios de accesibilidad. A pesar de que se han seguido las recomendaciones de diseño y accesibilidad que propone Android, y obtener unos resultados satisfactorios por parte de las pruebas con usuarios, sería necesarias nuevas evaluaciones sobre la aplicación, además de integrar a usuarios con discapacidad visual y motora.

2.8 La aplicación debe ser construida con vistas de que en un futuro se puedan añadir más acciones que no sea sólo la ejecución de aplicaciones. En todo momento se ha desarrollado utilizando componentes de JAVA y Android comúnmente usados y conocidos por la comunidad de desarrolladores para facilitar futuras versiones y mejoras.

3. Haber realizado pruebas con usuarios con un resultado satisfactorio

Los resultados descritos en el apartado 6.2, demuestran que las tareas analizadas tuvieron un alto porcentaje de finalización, y que a pesar de la eficiencia más baja del perfil 2, las métricas de satisfacción fueron buenas (con un grado de complejidad bajo y una puntuación en el cuestionario SUS alta).

No obstante serían necesarias nuevas pruebas una vez solucionados los problemas detectados y añadir el perfil de usuarios con discapacidades visuales o motoras.

4. Conseguir una lista de mejoras y nuevas funcionalidades para una versión posterior

De las diferentes pruebas realizadas, se ha conseguido una lista de mejoras en la aplicación y sugerencias por parte de los usuarios que son descritas en el apartado 7.2.

En resumen, se han conseguido en un alto porcentaje todos los resultados que se deseaban de la aplicación. No obstante, para sacar pruebas más concluyentes sobre usabilidad y accesibilidad se deberían realizar nuevas evaluaciones y añadir nuevos perfiles.

7.2 Trabajo futuro

De las pruebas con usuarios se han detectado algunos problemas y obtenido varias sugerencias de los usuarios que se tendrían que abordar en una segunda versión de la aplicación.

A continuación se describe la lista de tareas a realizar en un futuro:

1. Problemas graves

- La aplicación debe tener una opción para lanzar las rutinas, con la misma funcionalidad que el widget.
- La lista que muestra las aplicaciones debe mostrar que es una lista claramente, con su scroll además de remarcar en el enunciado que se trata de una lista.
- Se deben corregir todas aquellas etiquetas que han llevado a confusión al usuario.
- Se debe corregir el widget para que muestre que está asociado a TAPADÚ al ser leído por un lector de pantallas.

2. Mejoras importantes

- La autenticación al servidor se debería realizar usando OAuth.
- Añadir la opción de recordar la contraseña del registro a través del correo.
- Se debería filtrar aquellas aplicaciones sobre las que se pueda asociar una rutina, impidiendo el mal uso de ésta.
- Posible versión para iOS.

3. Sugerencias

- La aplicación debería permitir tareas más complicadas que lanzar aplicaciones. Por ejemplo llamar a un contacto de la agenda.
- Debería mostrarse un pequeño buscador en la lista de aplicaciones para facilitar la búsqueda, al igual que en las rutinas generadas.
- Permitir personalizar la aplicación con skins.

7.3 Reflexión final

Con la realización de este proyecto se han asentado muchos de los conocimientos adquiridos en el máster:

- Interacción natural
- Principios de usabilidad
- Arquitectura de la información
- Accesibilidad
- Prototipado
- Evaluaciones sobre la aplicación: inspección de estándares, pruebas con usuarios.

Además, se ha adquirido nuevos conocimientos sobre el desarrollo de aplicaciones Android de la que no se tenía experiencia y recordado principios y pautas de ingeniería de software, frameworks, y desarrollo.

Pero uno de los mayores hallazgos que he realizado a lo largo del proyecto es el estado actual de accesibilidad de las aplicaciones en Smartphones y comprobar de primera mano que aplicaciones hechas por empresas importantes como Google o Spotify, tengan botones sin etiquetar y que al ser leídas por un lector de pantalla lea la referencia del botón. Todo ello pone de manifiesto que, si bien existe información en la red sobre las pautas de accesibilidad, no existe un estándar o herramientas (al igual que ocurren con las páginas web) que puedan testear la aplicación de una forma sencilla, y ayude a divulgar la importancia de desarrollar las aplicaciones teniendo en cuenta los principios básicos accesibles, que además son sencillos de adquirir y mantener.

Las aplicaciones para Android, al estar desarrolladas por una comunidad abierta, no tiene un control tan estricto de las aplicaciones subidas a Google Play, lo cual, va en detrimento de imponer unas normas básicas de accesibilidad. Quizás ayudaría en gran medida una herramienta que permita que las aplicaciones puedan ser evaluadas en cuanto a su accesibilidad, y que dicha información se obligue a mostrar en la página de Google Play, junto al número de descargas. De este modo se conseguiría una mayor divulgación de los principios de accesibilidad.

A pesar de todo ello, con la versión 4.0 se introdujeron grandes mejoras en accesibilidad, haciendo que por ejemplo, no haya tanta diferencia de uso entre los lectores de pantalla iOS (Voice Over) y Android (TalkBack) ofreciendo las mismas funcionalidades. Lo cual no obliga a personas que necesitan usarlos, tener que adquirir un dispositivo de la marca de Apple, y gozando de una comunidad más abierta con un mayor número de aplicaciones como es Android.

Con este proyecto, considero que he conseguido poner un pequeño granito de arena en las aplicaciones que facilitan la interacción de los usuarios con sus dispositivos, en diversas situaciones y con todo tipo de usuarios, incluidas personas mayores y personas con discapacidad visual.

8 Bibliografía

- [1] KitKat 4.4 en Android. [Fecha de consulta: 01/09/2014] Disponible en <http://www.android.com/versions/kit-kat-4-4/>
- [2] Accessibility en Developers Android. [Fecha de consulta: 01/09/2014] Disponible en <http://developer.android.com/design/patterns/accessibility.html>
- [3] Google Play en Google Play. [Fecha de consulta: 01/09/2014] Disponible en <https://play.google.com/store?hl=es>
- [4] Spiel Screen Reader en Spiel Project. [Fecha de consulta: 01/09/2014] Disponible en <http://spielproject.info/>
- [5] eSpeak TTS en Google Code. [Fecha de consulta: 01/09/2014] Disponible en <https://code.google.com/p/eyes-free/>
- [6] Parkible en Innube parkible. [Fecha de consulta: 01/09/2014] Disponible en <http://innube.com/productos/parkible/>
- [7] Big Launcher en Big Launcher. [Fecha de consulta: 01/09/2014] Disponible en <http://biglauncher.com/es/>
- [8] Implementing accessibility en Developers Android. [Fecha de consulta: 01/09/2014] Disponible en <http://developer.android.com/training/accessibility/index.html>
- [9] Accessibility in iOS en Apple. [Fecha de consulta: 01/09/2014] Disponible en <https://www.apple.com/es/accessibility/ios/>
- [10] Accessibility in iOS en Developer Apple. [Fecha de consulta: 01/19/2014] Disponible en <https://developer.apple.com/technologies/ios/accessibility.html>
- [11] SIRI en Apple. [Fecha de consulta: 01/09/2014] Disponible en <https://www.apple.com/es/ios/siri/>
- [12] SHERPA Personal Assistant en sher.pa. [Fecha de consulta: 01/09/2014] Disponible en <http://sher.pa/>
- [13] Everfriends en Everfriends. [Fecha de consulta: 01/09/2014] Disponible en <http://www.everfriends.com/>
- [14] Speaktoit Assistant en speaktoit. [Fecha de consulta: 01/09/2014] Disponible en <http://www.speaktoit.com/>
- [15] Google Voice Search en Google Play. [Fecha de consulta: 01/19/2014] Disponible en <https://play.google.com/store/apps/details?id=com.google.android.voicesearch&hl=es>
- [16] Struts 2 en The Apache Software foundation. [Fecha de consulta: 01/09/2014] Disponible en <http://struts.apache.org/development/2.x/>

- [17] Hibernate en Hibernate. [Fecha de consulta: 01/09/2014] Disponible en <http://hibernate.org/>
- [18] PostgreSQL en PostgreSQL. [Fecha de consulta: 01/09/2014] Disponible en <http://www.postgresql.org.es/>
- [19] MySQL en MySQL. [Fecha de consulta: 01/09/2014] Disponible en <http://www.mysql.com/>
- [20] XML Tutorial en w3schools. [Fecha de consulta: 01/09/2014] Disponible en <http://www.w3schools.com/xml/>
- [21] JSON en JSON. [Fecha de consulta: 01/09/2014] Disponible en <http://json.org/>
- [22] REST Plugin en Apache Software Foundation. [Fecha de consulta: 01/09/2014] Disponible en <http://struts.apache.org/release/2.3.x/docs/rest-plugin.html>
- [23] FLUID UI en FLUID UI. [Fecha de consulta: 01/09/2014] Disponible en <https://www.fluidui.com/>
- [24] UNE 139802:2003 – Requisitos de accesibilidad del software. [Fecha de consulta: 01/09/2014] Disponible en:
<http://www.udc.gal/fcs/ga/web-to/terapia/asignaturas/toyafam/08tema/UNE139802-2003.pdf>
- [25] Accessibility Testing Checklist en Android developers. [Fecha de consulta: 01/09/2014] Disponible en http://developer.android.com/tools/testing/testing_accessibility.html
- [26] Cómo hacer “Apps” Accesibles en CEAPAT. [Fecha de consulta: 01/09/2014] Disponible en
<http://www.ceapat.es/InterPresent2/groups/imserso/documents/binario/appsaccesibles.pdf>
- [27] LACALLE A. Pensando en alto: Thinking Aloud. [Fecha de consulta: 01/09/2014] Disponible en <http://albertolacalle.com/hci/thinking-aloud.htm>
- [28] Accessibility Developer Checklist en Android developers. [Fecha de consulta: 01/09/2014] Disponible en <http://developer.android.com/guide/topics/ui/accessibility/checklist.html#special-cases>
- [29] Usabilidad en Cenatic. [Fecha de consulta: 01/09/2014] Disponible en <https://forja.cenatic.es/plugins/mediawiki/wiki/alsigm/index.php/Usabilidad>
- [30] BROOKE J. SUS - A quick and dirty usability scale. [Fecha de consulta: 01/09/2014] Disponible en www.usability.serco.com/trump/documents/Suschapt.doc

Anexo A. Patrón MVC

Modelo Vista Controlador (MVC) es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. El patrón MVC se ve frecuentemente en las aplicaciones Web.

La ilustración 8 muestra cómo interactúan las 3 partes.

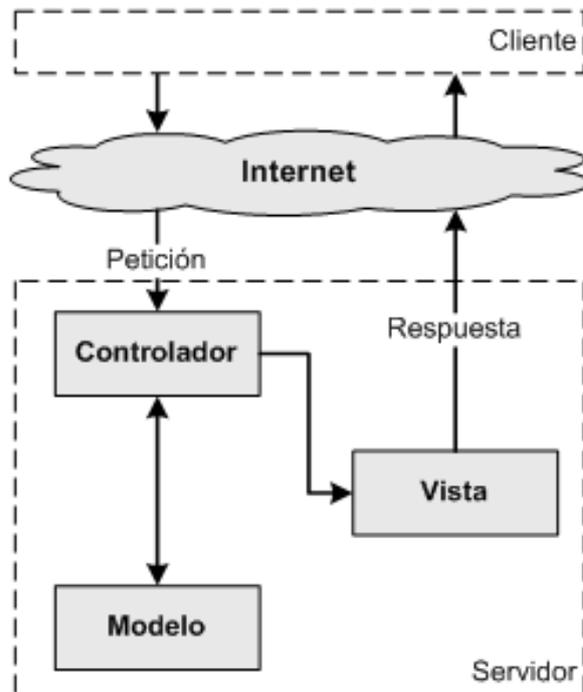


Ilustración 10. Patrón MVC con las tecnologías usadas en el proyecto.

A.1 Componentes del MVC

- **Modelo:** es la parte encargada de representar el dominio de la aplicación, es decir, todos los datos que pueden ser tratados.
- **Vista:** es la interfaz gráfica que permite al usuario comunicarse con el usuario. También se encarga de presentarle los datos del dominio de la aplicación.
- **Controlador** Se encarga de comunicar al usuario (las vistas) con el dominio de la aplicación (el modelo). Este componente centraliza todas las peticiones del usuario, delega las tareas entre las diferentes clases que tiene subordinadas, las cuales se encargan de operar sobre el modelo, y devuelve una respuesta al usuario.

Anexo B. Diagramas de colaboración, clases y base de datos

B.1 Diagrama de colaboración

B.1.1 Aplicación Android

A continuación se muestran los diagramas de colaboración de los casos de uso descritos para la aplicación Android:

crearRutina(String nomPackage, String palabraClave)

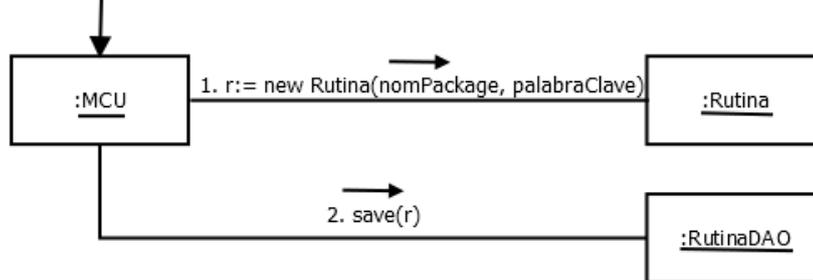


Ilustración 11. Diagrama de colaboración para "Crear rutina"

consultarCatalogo()

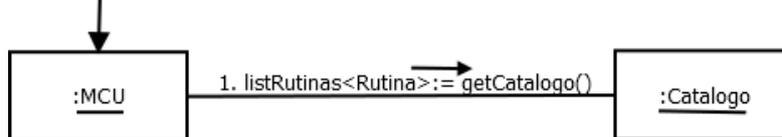


Ilustración 12. Diagrama de colaboración "Consultar catálogo"

consultarRutina(String nomPackage)

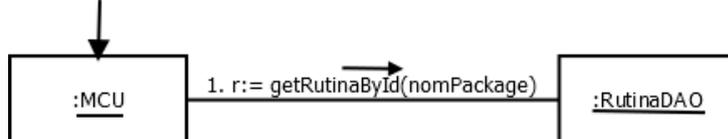


Ilustración 13. Diagrama de colaboración "Consultar rutina"

modificarRutina(String nomPackage, String palabraClave)

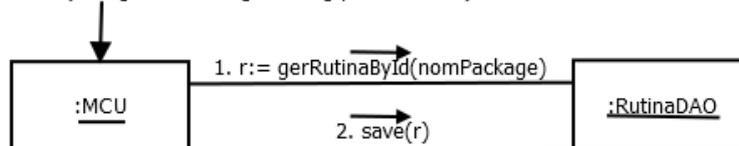


Ilustración 14. Diagrama de colaboración "Modificar rutina"

eliminarRutina(String nomPackage)

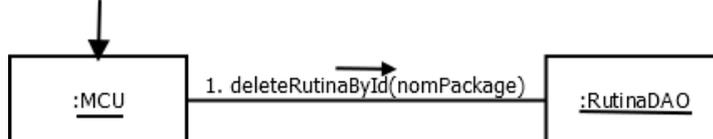


Ilustración 15. Diagrama de colaboración "Eliminar rutina"

descargarCatalogo(String user, String pass)

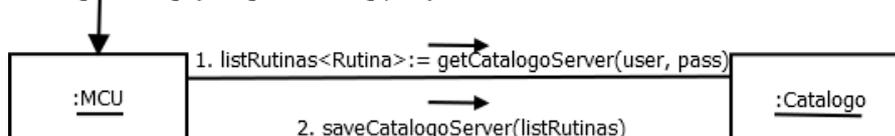


Ilustración 16. Diagrama de colaboración "Descargar catálogo"

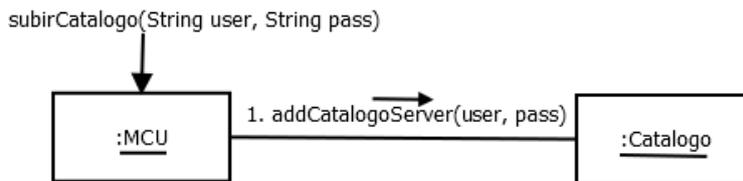


Ilustración 17. Diagrama de colaboración "Subir catálogo"

B.1.2 Widget

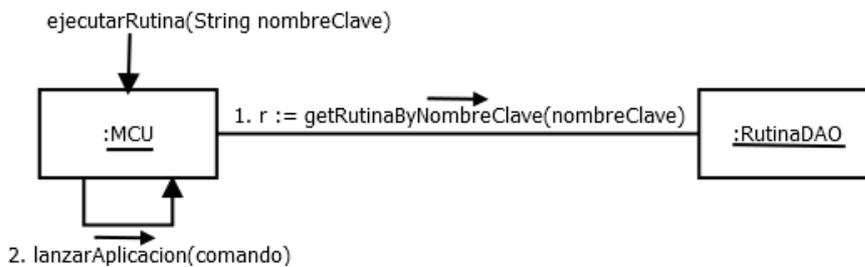


Ilustración 18. Diagrama de colaboración "Ejecutar rutina"

B.1.3 Aplicación web

Ilustración 19. Diagrama de colaboración "Eliminar rutina"

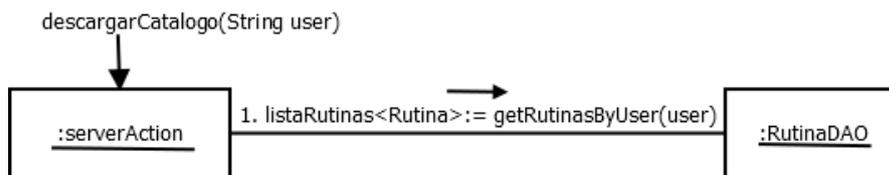


Ilustración 20. Diagrama de colaboración "Descarga catálogo"

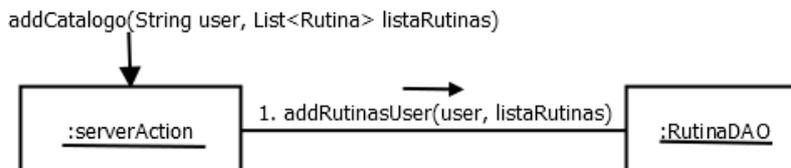


Ilustración 21. Diagrama de colaboración "Subir catálogo"

B.2 Diagrama de clases

A continuación se muestran los diagramas de clase para las 3 aplicaciones diseñadas.

B.2.1 Aplicación Android

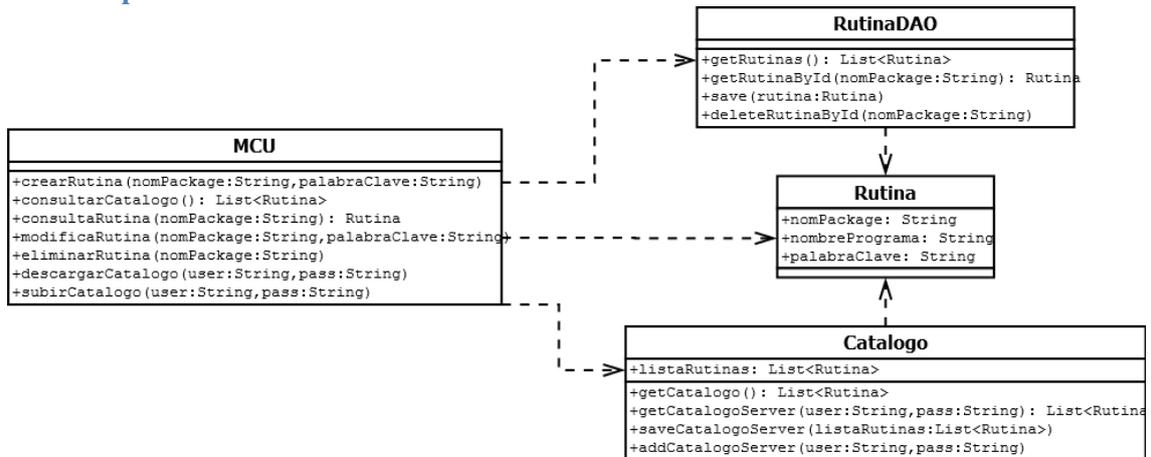


Ilustración 22. Diagrama de clases de la aplicación Android.

B.2.2 Widget

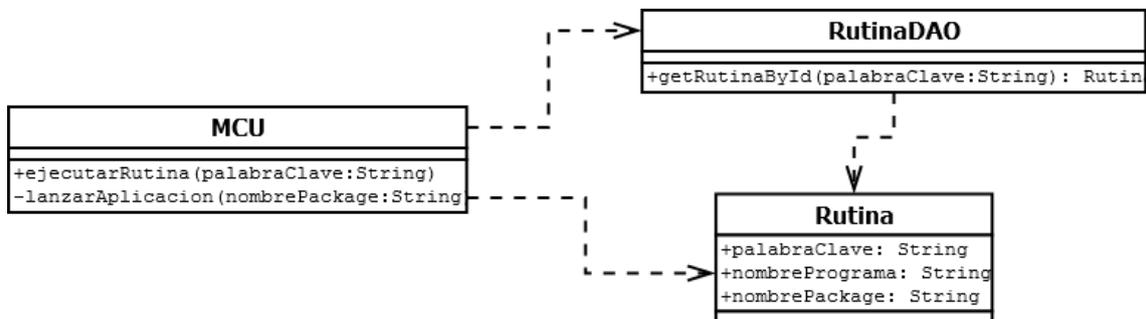


Ilustración 23. Diagrama de clases del widget.

B.2.3 Aplicación web

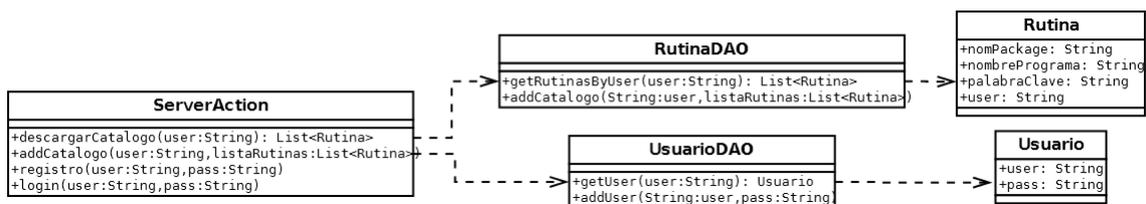


Ilustración 24. Diagrama de clases de la aplicación web.

B.3 Diseño de la base de datos

A continuación se muestran las tablas que compondrán la base de datos que guarde el catálogo común y que gestionará la aplicación web.

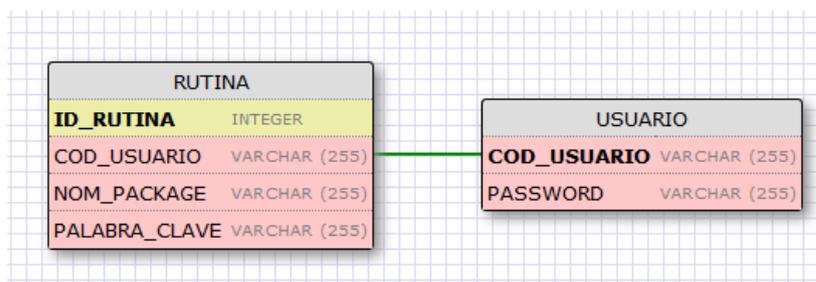


Ilustración 25. Diagrama de la base de datos.

Anexo C. Accesibilidad en Android y evaluación con expertos de TAPADÚ

C.1 Introducción

A pesar de que se hayan seguido las guías de accesibilidad en las etapas de diseño y desarrollo, pueden existir problemas con la accesibilidad que no se descubren hasta que se prueba la aplicación.

Para evaluar la accesibilidad de la aplicación se han tenido en cuenta las recomendaciones ofrecidas por Android en las etapas de diseño, desarrollo y testeo y la norma UNE 139802:2009 - Requisitos de accesibilidad al ordenador Software - (extrapolando los principios a las aplicaciones móviles). Con todo ello se han conseguido 10 categorías de recomendaciones que se han clasificado por prioridad según afecte a la accesibilidad del sistema.

Las 10 categorías son:

1. Recomendaciones generales
2. Teclado
3. Dispositivos apuntadores
4. Pantalla
5. Sonidos y multimedia
6. Notificación al usuario
7. Información de objetos
8. Tiempo
9. Documentación
10. Otros

Las prioridades son:

- **Prioridad 1:** El producto debe satisfacer este requisito. En otro caso será imposible para uno o más grupos de personas con discapacidad utilizar el producto. Cumplir un requisito de esta prioridad es básico para que algunos grupos puedan usar el producto.
- **Prioridad 2:** El producto debería satisfacer este requisito. En otro caso será muy difícil para uno o más grupos de personas con discapacidad utilizar el producto. Cumplir un requisito de esta prioridad permite eliminar barreras significativas a la hora de usar el producto.
- **Prioridad 3:** El producto puede satisfacer este requisito. En otro caso algunos grupos podrían encontrar una dificultad relativa para utilizar el producto. Cumplir un requisito de esta prioridad mejorará la accesibilidad del producto.

Los objetivos finales de esta lista, es comprobar que la aplicación puede:

- Ser configurada y usada sin necesidad de la visión.
- Todas las tareas pueden ser usadas a través de controles direccionales, y ofrecen un feedback claro y conciso.

C.2 Recomendaciones

C.2.1 Recomendaciones generales

Prioridad 1

1. El software debe estar diseñado para minimizar el número de pasos que debe realizar el usuario para activar cualquier opción.
2. Se debe permitir al usuario elegir dispositivos de entrada/salida.
3. Se debe poder manejar el software de forma efectiva utilizando sólo uno de los posibles dispositivos de entrada. Esto implica que la aplicación puede ser navegable sin hacer uso de la pantalla táctil.
4. Las aplicaciones deben utilizar los servicios ofrecidos por el sistema operativo para facilitar su accesibilidad.
5. Las aplicaciones no deben desactivar o interferir en las características de accesibilidad del sistema operativo o de otros productos.
 - a. Verificar que todos los componentes de la interfaz, ofrecen información clara al usuario, al ser descritos por el lector de pantalla Talkback
 - b. Verificar que los gestos de la aplicación (como pueden ser zoom sobre imágenes, desplazarse por listas, cambiar páginas...) continúan funcionando cuando Talkback está activo.

Prioridad 2

6. Se debe proporcionar una función que permita a los usuarios deshacer los efectos de acciones no intencionadas. Si una acción no puede deshacerse, se debe pedir confirmación antes de realizarla.
7. Si el usuario cambia de tarea, al regresar a la anterior, su interfaz debe recordar cuál era el control que tenía el foco.
8. Se debe permitir la definición de perfiles con las preferencias del usuario.
9. Cada usuario debe poder cambiar y mantener sus propias preferencias de usuario mediante la interfaz del sistema.
10. Las aplicaciones deben usar los servicios estándar de entrada/salida del sistema operativo.

Prioridad 3

11. Se debe permitir intercambiar rápidamente dispositivos alternativos para la entrada/salida.
12. Se debe permitir al usuario transferir sus preferencias a otro sistema compatible.

C.2.2 Teclado

Prioridad 1

1. Se deben poder activar todas las funciones (incluyendo la navegación) sólo mediante teclado.
2. Los comandos de navegación por teclado no deben activar los objetos de interfaz.

Prioridad 2

3. El desplazamiento mediante teclado de un elemento a otro en los cuadros de diálogo debe seguir una secuencia consistente con la distribución en pantalla.
4. La navegación entre elementos de la interfaz debe ser circular.
5. Las aplicaciones deben respetar las convenciones de funcionamiento del teclado en el sistema operativo.

C.2.3 Dispositivos apuntadores

Prioridad 1

1. Se deben ofrecer alternativas para pulsaciones simultáneas de teclas y botones del dispositivo apuntador.
2. El área sensible al tacto en una pantalla táctil, que activa o selecciona un elemento de la interfaz gráfica de un usuario debe tener una dimensión óptima de 9x9 mm, no debiendo ser inferior a 8mm de ancho por 7 mm de alto.
3. Se debe poder emular el clic múltiple mediante la pulsación única de una tecla. Es una función más propia del sistema operativo, pero puede ser también una facilidad implementada por un software para productos de apoyo.
4. Se debe poder emular la pulsación mantenida de un botón del dispositivo apuntador mediante la pulsación única de un botón. Es una función más propia del sistema operativo, pero puede ser también una facilidad implementada por un software para productos de apoyo

C.2.4 Pantalla

Prioridad 1

1. Todos los textos presentados en pantalla deben ser generados mediante las funciones del sistema dedicadas a mostrar texto.
2. No debe usarse el color como única fuente de información.
3. Todos los controles deben tener una dimensión óptima de 9x9 mm, no debiendo ser inferior a 8mm de ancho por 7 mm de alto
4. Deben existir opciones para modificar el tipo de letra, el tamaño y el color de todos los controles de la interfaz.
5. Cuando se presentan animaciones debe ofrecerse una versión alternativa no animada de su contenido.
6. Se debe evitar presentar elementos que parpadeen o destellen con una frecuencia entre 2 y 50 Hz.

Prioridad 2

7. Todos los iconos deben poder tener asociada una etiqueta de texto y debe existir la posibilidad de visualizar sólo esa etiqueta.
8. La interfaz de usuario debe adaptarse a la configuración de contraste, color, tamaño y demás atributos de visualización que haya definido el usuario en el sistema operativo.
9. Debe existir al menos un modo de presentación de información visual que sea legible para usuarios con agudeza visual entre 6/18 y 6/60 sin depender del sonido.
10. Deben proporcionarse combinaciones de colores predefinidas que hayan sido diseñadas teniendo en cuenta las necesidades de las personas con deficiencias visuales.

11. Las etiquetas de los campos de entrada o visualización de datos de los formularios deben estar próximas a estos campos.

C.2.5 Sonidos y Multimedia

Prioridad 1

1. Los contenidos relevantes en formato audio o vídeo deben ofrecerse también en otros formatos alternativos.
2. El usuario debe poder activar la presentación visual de avisos sonoros.
3. Deben ofrecerse funciones que permitan enviar cualquier información textual a una salida mediante síntesis de voz.
4. La salida en síntesis de voz debe aparecer inmediatamente después de ocurrir el evento que la originó.
5. Asegurarse que no se duplica información, lo que provocaría que el lector de pantalla repitiera información.
6. Asegurarse que los sonidos de la aplicación no interfieren en el volumen del lector de pantalla.

Prioridad 2

7. El usuario debe poder ajustar el volumen de los sonidos.
8. La aplicación debe proporcionar controles de reproducción para reproducir, pausar, saltar y avanzar o retroceder.

C.2.6 Notificación al usuario

Prioridad 1

1. Los mensajes emitidos deben ser cortos, sencillos y redactados en un lenguaje claro para el usuario no técnico.
2. Los mensajes del mismo tipo deben ser claramente identificables: siempre deben aparecer en la misma posición de pantalla, deben tener el mismo formato y deben estar etiquetados de forma unívoca y estándar.

C.2.7 Información de objetos

Prioridad 1

1. Se debe proporcionar a otras aplicaciones información semántica sobre los objetos de la interfaz de usuario. Esto implica que todos los componentes de la interfaz, ofrecen información clara al usuario, al ser descritos por el lector de pantalla Talkback
 - a. Instrucciones (Hint) en campos de texto.
 - b. Controles personalizados con alto contexto visual: debe proporcionarse al control una vista jerárquica virtual que permita a los servicios de accesibilidad un modelo de interacción simplificada para el usuario.
 - c. Si un control personalizado en la aplicación responder a un manejo específico de la interacción táctil del usuario, debe activarse a un clic y proporcionar la información necesaria para que los servicios de accesibilidad procesen esta acción para los usuarios.

- d. Controles que cambian de función hay que garantizar que la información que devuelve el control también ha cambiado apropiadamente dando una información de accesibilidad correcta.
 - e. Indicaciones para controles relacionados: debe garantizarse que cuando el usuario interactúa con los controles individuales proporcionen una información de audio útil.
 - f. Imágenes decorativas y gráficos no deberían tener descripción de accesibilidad del contenido.
2. Todos los controles, objetos, iconos e imágenes de la interfaz de usuario deben tener un texto asociado que indique su función o significado. Dicho nombre debe ser consistente, significativo, corto y conciso. Al ser activada la función “exploración por toque todos los componentes de la interfaz, ofrecen información clara al usuario.

Prioridad 2

3. El foco de entrada debe quedar reflejado en pantalla de forma inequívoca.

C.2.8 Tiempo

Prioridad 1

1. El usuario debe poder pausar o detener la presentación dinámica de información.
2. La información sobre errores o los avisos relevantes para la tarea actual deben persistir hasta que el usuario confirme su lectura.
3. Si se requiere una respuesta del usuario en un intervalo de tiempo determinado, se debe poder ajustar dicho intervalo, incluyendo la posibilidad de desactivar todos los límites de tiempo.

C.2.9 Documentación

Prioridad 1

1. La documentación del producto debe estar redactada de la forma más clara y sencilla posible, dentro del vocabulario del dominio de la aplicación.
2. Se deben proporcionar sistemas de ayuda en texto sencillo, complementado de forma opcional mediante lengua de signos.
3. La documentación del producto debe estar disponible en formatos alternativos bajo petición del usuario, ajustándose a sus necesidades específicas y sin coste adicional.
4. La información sobre las características de accesibilidad del producto debe estar disponible en formatos alternativos bajo petición del usuario, ajustándose a sus necesidades específicas y sin coste adicional.

C.2.10 Otros

Prioridad 1

1. Las aplicaciones deben ofrecer la opción de finalizar.

C.3 Resultados detallados

A continuación se presentan las recomendaciones que no se cumplen junto a la descripción del problema, las pantallas donde se produce y las soluciones propuestas para resolverlo.

C.3.1 Prioridad 1

Recomendaciones generales

- **Recomendación 1.3:** Se debe poder manejar el software de forma efectiva utilizando sólo uno de los posibles dispositivos de entrada. Esto implica que la aplicación puede ser navegable sin hacer uso de la pantalla táctil.

Problemas detectados: Ninguno de los botones de la aplicación recoge el foco. Esto es principalmente grave en el menú inicial:

Solución: Modificar el atributo del botón para que sea focusable.



Ilustración 26. El menú principal no recoge el foco

- **Recomendación 1.5.a:** Verificar que todos los componentes de la interfaz, ofrecen información clara al usuario, al ser descritos por el lector de pantalla Talkback.

Problemas detectados: El literal “1. Selecciona la aplicación” queda extraño el número. El literal “Paso 1/2” el lector de pantalla lo lee como “Paso un medio”.

Solución: Modificar los literales para eliminar el número y en el segundo literal se modifica “1/2” por “1 de 2”.



Ilustración 27. Hay información que al ser leída por el lector de pantalla es confusa

Recomendaciones sonidos y multimedia

- **Recomendación 5.5:** Asegurarse que no se duplica información, lo que provocaría que el lector de pantalla repitiera información.

Problemas detectados: El literal “Pulsar el botón para decir la palabra clave” el lector de pantalla lo lee como “Botón pulsar el botón para decir la palabra clave”.

Solución: Eliminar palabra botón del literal.

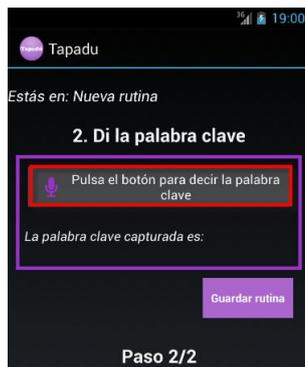


Ilustración 28. Se duplica la palabra botón al ser leído por el lector de pantalla.

Recomendaciones sobre la información de objetos

- **Recomendación 7.1.d:** Controles que cambian de función hay que garantizar que la información que devuelve el control también ha cambiado apropiadamente dando una información de accesibilidad correcta.

Problemas detectados: Al crear o modificar una rutina, y capturar la palabra clave, se actualiza el componente pero el lector de pantalla no informa de ello. Lo mismo ocurre con el widget al decir una rutina que no tiene asociada ninguna palabra.

Solución: Modificar el literal por un “Toast” o mensaje de alerta, provocando un evento que el lector de pantalla detecta y lee.

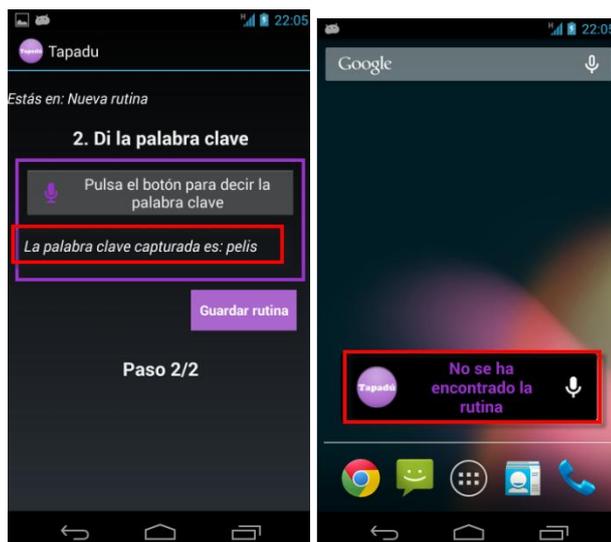


Ilustración 29. El lector de pantalla no detecta la modificación del control.

Recomendaciones sobre tiempo

- **Recomendación 8.2:** La información sobre errores o los avisos relevantes para la tarea actual deben persistir hasta que el usuario confirme su lectura.

Problemas detectados: En el widget, al decir una rutina que no tiene asociada ninguna palabra se muestra un mensaje de error hasta que vuelve a ser pulsado. Pero pasa desapercibido por el lector de pantalla.

Solución: Modificar el literal pro un mensaje de alerta, provocando un evento que el lector de pantalla detecta y lee y que sea el usuario quien decida cuando desaparece.

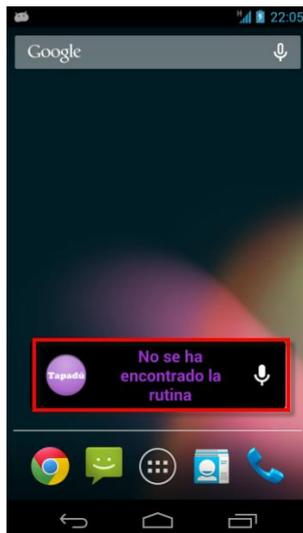


Ilustración 30. El mensaje de rutina no encontrada debe mostrarse en mensaje de error

Otras recomendaciones

- **Recomendación 10.1:** Las aplicaciones deben ofrecer la opción de finalizar.

Problemas detectados: En el menú principal no existe una opción para salir de la aplicación.

Solución: Se creará una nueva opción que cierre la aplicación.

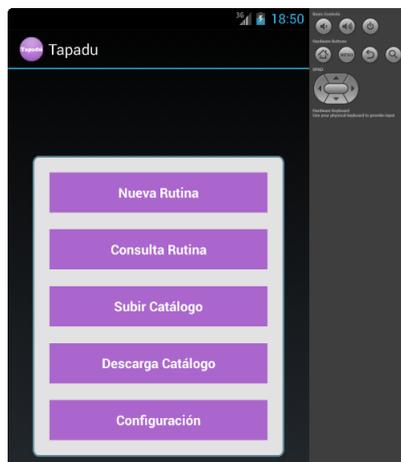


Ilustración 31. No existe la opción salir de la aplicación.

C.3.2 Prioridad 2

Recomendaciones sobre la pantalla

- **Recomendación 4.10:** Deben proporcionarse combinaciones de colores predefinidas que hayan sido diseñadas teniendo en cuenta las necesidades de las personas con deficiencias visuales.

Problemas detectados: Los colores del botón que son blanco sobre lila, ofrecen un contraste inadecuado.

Solución: Modificar el color de fondo AA66CC por 9966FF.



Ilustración 32. El contraste de los botones lilas, es insuficiente.

Recomendaciones sobre la información de objetos

- **Recomendación 7.3:** El foco de entrada debe quedar reflejado en pantalla de forma inequívoca.

Problemas detectados: Ninguno de los botones muestra el foco.

Solución: Modificar el atributo del botón para que sea focusable.

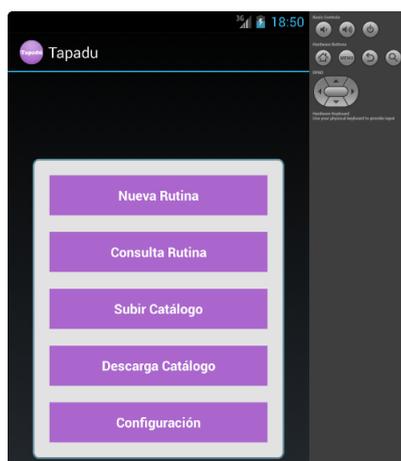


Ilustración 33. El menú principal no recoge el foco

Anexo D. Pruebas con usuarios

D.1 Formulario de consentimiento en la participación del proyecto

Hoja de información del proyecto

- La persona coordinadora del proyecto me ha explicado el proyecto y estoy de acuerdo en participar en él.
- Entiendo que si tengo alguna pregunta de la Hoja de Información o de su explicación, puedo preguntar al investigador antes de que acepte participar en el proyecto. Se me entregará una copia de esta Hoja de Consentimiento para guardarla.
- Entiendo que si decido, en cualquier momento de la investigación, que no deseo participar más en el proyecto, puedo hacerlo sin tener que dar explicación alguna. Más aún, entiendo que puedo retirar mis datos hasta tres semanas después de la fecha de la entrevista.
- Consiento el procesamiento de la información personal para las propuestas que me han explicado. Entiendo que tal información será tratada en concordancia con los términos de las Leyes de Protección de Datos vigentes.

Detalles de Contacto del participante:

Nombre (en mayúsculas):

Género:

Edad:

Tipo de discapacidad (para participantes con discapacidad):

Declaración del participante:

Yo _____ estoy de acuerdo en que se me ha facilitado toda la información del proyecto de investigación y en qué consiste mi participación, y estoy de acuerdo en formar parte del estudio. He leído la Hoja de Información del proyecto arriba incluida y entiendo lo que el estudio de investigación implica.

Marcar la siguiente casilla si se autoriza a la grabación de la sesión:

Autorizo a que la sesión sea grabada en video para su posterior análisis, teniendo en cuenta que los datos recogidos podrán ser utilizados únicamente con finalidades científicas y/o académicas.

D.2 Transcripción del facilitador

Introducción

Buenos días, mi nombre es Vanessa Andrés Blas y actualmente estoy cursando el Master de Interacción Persona–Ordenador en la Universidad de Lleida. Estoy desarrollando una aplicación Android que permite lanzar aplicaciones diciendo una palabra clave que anteriormente habremos seleccionado. Es decir, a una aplicación le asociamos una palabra clave y cuando decimos esa palabra clave, Android abre la aplicación.

Necesito realizaros unas preguntas para asegurarme que estoy desarrollando una aplicación que sea fácil de usar, entendible y se adapte a las necesidades de los usuarios. Conocer vuestras opiniones acerca de la aplicación para evitar esa sensación de frustración que tantas veces nos acompaña, cuando intentamos usar algo por primera vez. Introducción del equipo. Quiero remarcar que lo que voy a evaluar es la aplicación y los problemas que ofrece, y no sus habilidades en el uso de dispositivos móviles.

Permiso para ser grabado en video

Para tener en cuenta todos los comentarios que digas y no perder información, me es de gran ayuda poder grabar el test. Para ello, necesito que me firmes este papel en el que ofreces tu consentimiento para ser grabado para la prueba y los datos que se recojan, solo podrán ser utilizados únicamente con finalidades científicas y/o académicas para su posterior análisis.

No debes preocuparte, pues las imágenes sólo serán tratadas por mí y los datos que se obtengan serán tratados de forma anónima. Bajo ningún concepto estas grabaciones acabarán en manos de terceras personas. Si tienes alguna duda o problema con ello, no existe ninguna penalización por negarte.

Introducción al equipo

Sólo estaremos tú y yo mientras realizas el test y se va grabando las pruebas. Yo, siempre estaré aquí a tu lado para resolver tus dudas, pero piensa que la idea es simular la situación en la que tú te encuentras con la aplicación y quieres usarla.

Cuestionario post-tarea

Como ya he comentado, estamos aquí para testear una aplicación Android. Esto significa que queremos saber lo qué le parece a la gente, recoger opiniones, críticas, pensamientos... cualquier información que nos des será bienvenida. Por ello te pedimos, que digas en voz alta cualquier pensamiento que tengas, como “Esto no me gusta”, “No encuentro nada”, “Ohh, me parece muy útil esta parte”, “creo que voy a pulsar sobre...”

El test se compone de 3 tareas que te iré explicando y te repetiré todas las veces que creas necesario. Cuando creas que has terminado una tarea, debes decírmelo y entonces te pasará un pequeño cuestionario sobre las impresiones generales de la tarea, es rápido de rellenar, pues sólo tienes que elegir la respuesta con las que te sientas más identificado.

Finalmente, te pido que intentes prestar la misma atención y adoptar el mismo comportamiento que tendrías al realizar estas tareas en una situación real.

Cuestionario post-test

Al finalizar las tareas, te realizaré unas breves preguntas para que me expliques qué te ha parecido, si tienes alguna crítica o sugerencia... si te quedan dudas, cualquier cosa. Te agradezco profundamente que nos digas tu opinión más sincera, sin preocuparte por herir a nadie, necesito que los datos sean reales para poder introducir las mejoras que se necesiten.

Despedida

Muchas gracias por haber haberme prestado un poco de tu tiempo. Me ha sido de gran ayuda tu aportación.

D.3 Cuestionarios

D.3.1 Cuestionario pre-test

Generales

1. Género
 - Masculino
 - Femenino
2. Edad
 - 20-29 años
 - 30-39 años
 - 40-49 años
 - 50 años o +

Uso dispositivos móviles

1. Uso de tecnologías
 - Tablet
 - Portátil/Ordenador
 - Smartphone
 - Internet
2. ¿Qué SO tiene tu móvil?
 - Android
 - iOS
 - Windows 8
 - Blackberry
 - Otros: _____
3. ¿Durante cuántas horas al día haces uso del móvil?
 - < 1 hora
 - 1-2 horas
 - + 3 horas
4. ¿Cuánto tiempo hace que usas Smartphone?
 - < 1 año
 - 1-2 años
 - 3-4 años
 - 5 años o +
5. ¿Para qué usas el Smartphone? (selecciona tantas casillas como se desee)
 - Leer correos
 - Redes sociales
 - Comunicación (WhatsApp)
 - Juegos
 - Navegar internet
 - Otros: _____

D.3.2 Cuestionario post-tarea

1. ¿Has conseguido realizar la tarea?

Sí

No (En este caso, ¿puedes explicar por qué?)

2. ¿Te ha parecido complicada la tarea?

Nada

Un poco

Bastante

Mucho

3. ¿Te ha parecido que la cantidad de pasos para finalizarla era correcta?

Sí

No

4. ¿Te ha parecido que los pasos a seguir eran intuitivos?

Nada

Un poco

Bastante

Mucho

5. ¿Cambiarías algo en la tarea?

No

Sí (En este caso, ¿puedes explicar el qué?)

D.3.3 Cuestionario post-test

	En desacuerdo					De acuerdo				
1. Creo que me gustaría usar la aplicación de forma frecuente	1	2	3	4	5					
2. Encuentro la aplicación innecesariamente compleja	1	2	3	4	5					
3. Creo que la aplicación es fácil de usar	1	2	3	4	5					
4. Creo que necesitaría apoyo de un experto para usar la aplicación	1	2	3	4	5					
5. Considero que las diferentes opciones de la aplicación están bien integradas	1	2	3	4	5					
6. Considero que había demasiada inconsistencia en la aplicación.	1	2	3	4	5					
7. Imagino que la mayoría de las personas aprenderían rápidamente a utilizar la aplicación	1	2	3	4	5					
8. He encontrado la aplicación bastante incómoda de usar	1	2	3	4	5					
9. Me he sentido muy seguro usando la aplicación	1	2	3	4	5					
10. Necesito aprender muchas cosas antes de manejar la aplicación	1	2	3	4	5					

Anexo E. Resultados de las pruebas con usuarios

E.1 Tarea 1: Lanzar una aplicación con éxito

Esta tarea tenía una única etapa que consistía en lanzar una aplicación de notas diciendo correctamente la palabra clave.

E.1.1 Resumen de la tarea por participantes y perfiles

Participantes	EFECTIVIDAD		EFICIENCIA	SATISFACCIÓN
	Porcentaje de finalización	Grado problemas	Tiempo tarea	Complejidad
Participante 1	100%	0	23s	1
Participante 3	75%	1	120s	2
Participante 4	100%	0	34s	1
Participante 6	100%	0	14s	1
MEDIA PERFIL 1	93,75%	0,25	47,75s	1,25
Participante 2	100%	0	52s	1
Participante 5	75%	1	86s	1
Participante 7	100%	0	83s	2
Participante 8	100%	0	20s	1
MEDIA PERFIL 2	93,75%	0,25	60,25s	1,25

Las métricas de efectividad y satisfacción son iguales en los dos perfiles, mientras que las métricas de eficiencia (tiempo) es donde se muestran las mayores diferencias. El principal problema ha sido la falta de comprensión de las funcionalidades del widget y la aplicación. La diferencia en el uso habitual de dispositivos, ha marcado la diferencia de tiempo entre los perfiles.

E.1.2 Efectividad

A excepción de los participantes 3 (perfil 1) y 5 (perfil 2), el resto han completado las tareas sin problemas.

Los problemas detectados fueron:

- El participante 2 tapaba el micrófono y repitió 2 veces la palabra clave. Al ver que el widget no tenía un comportamiento adecuado, entró en la aplicación en la opción de menú "Consulta de rutinas". Al ver que no lo conseguía, lo intentó de nuevo, consiguiendo finalizar la tarea.
- El participante 5, no comprendió que era el widget quien lanzaba la aplicación y que primero debía ser pulsado. Así decía la palabra clave sin pulsar el widget ni entrar en la aplicación.

No existe diferencia entre perfiles.

TAPADÚ: Un lanzador de aplicaciones por voz

Resultados de las pruebas con usuarios

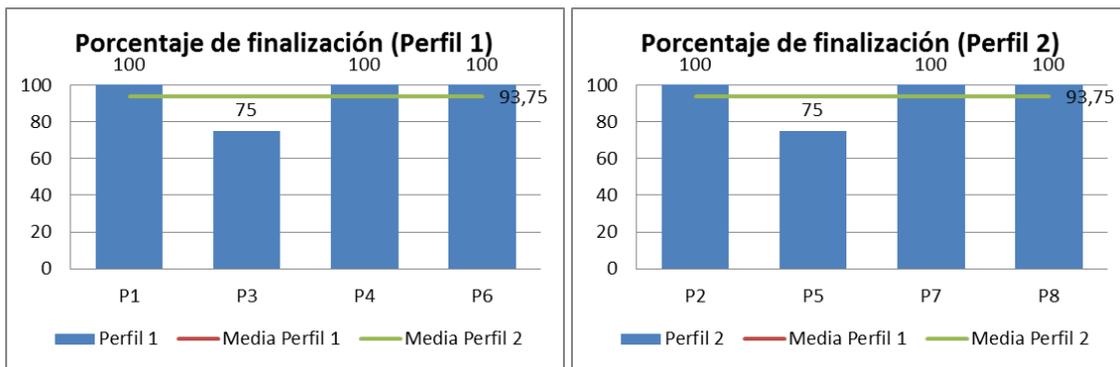


Ilustración 34. Porcentaje finalización de la tarea 1 por perfiles. La media del perfil 1 y 2 al ser la misma se solapan.

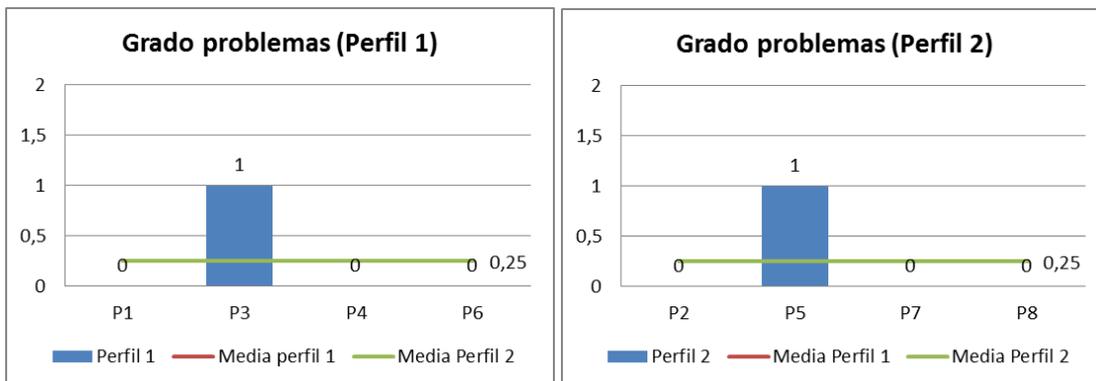


Ilustración 35. Grado de problemas de la tarea 1 por perfiles. La media del perfil 1 y 2 al ser la misma se solapan.

E.1.3 Eficiencia

La gráfica muestra el tiempo que cada participante tardó en realizar y la media por perfil.

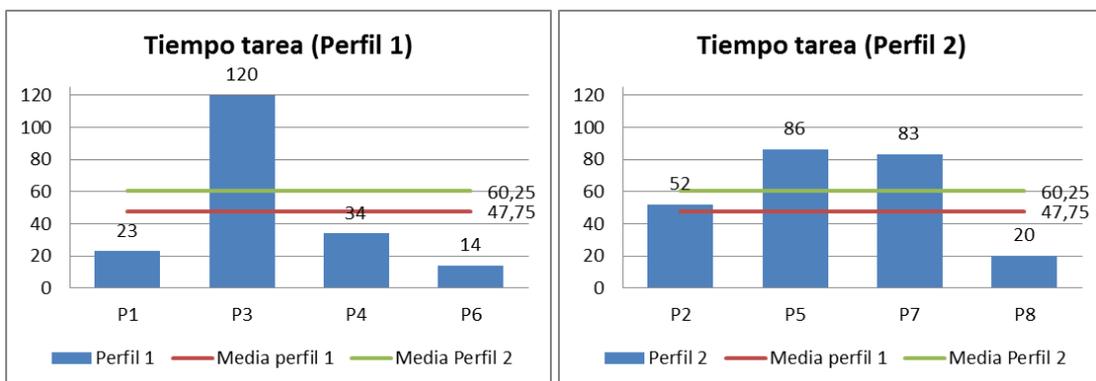


Ilustración 36. Tiempos de la tarea 1 por perfiles.

En esta gráfica es donde mejor se aprecia la diferencia entre los perfiles 1 y 2 (existe una diferencia de 13 segundos). El perfil 1 al estar más en contacto con el uso de dispositivos tecnológicos, comprende mejor el concepto de widget-aplicación, solventan mejor los problemas con los que se encuentran...

E.1.4 Satisfacción

Como se ve en la siguiente gráfica, los participantes están contentos con el grado de complejidad de la funcionalidad y no se ofrecen diferencias entre los diferentes perfiles.

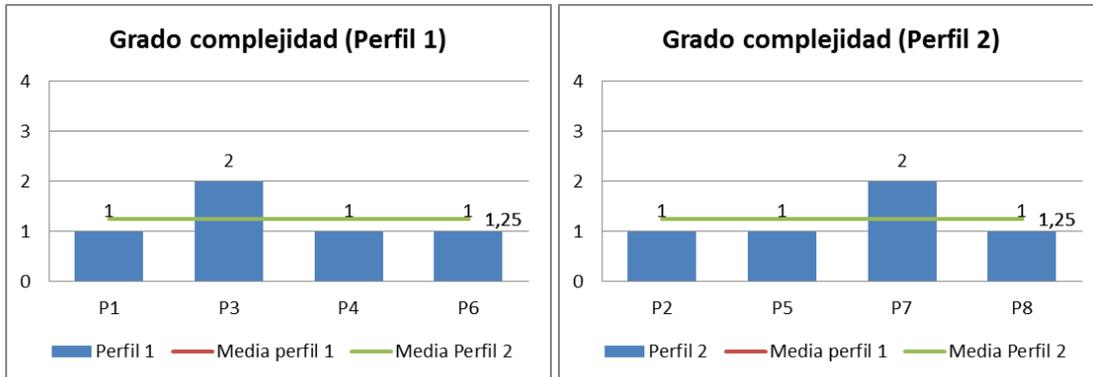


Ilustración 37. Grado de complejidad de la tarea 1 por perfiles. La media del perfil 1 y 2 al ser la misma se solapan.

E.1.5 Problemas detectados

Se ha detectado una mejora en esta tarea:

- **Problema:** A los usuarios con menos experiencia o menor uso de dispositivos móviles, les cuesta asociar las diferentes tareas que puede hacer una aplicación y el widget.

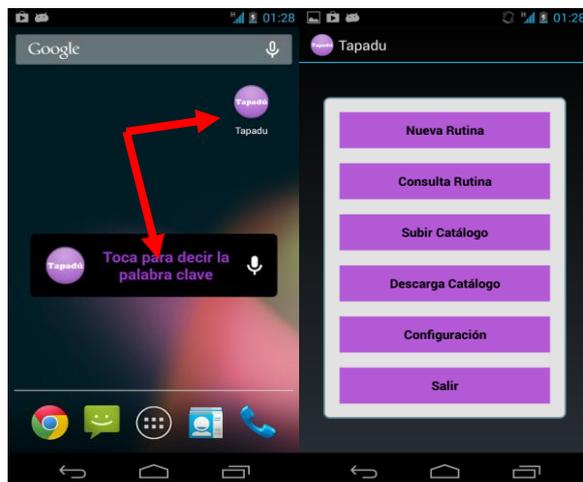


Ilustración 38. Confusión entre la aplicación y el widget

Solución: Añadir una nueva opción en el menú que se llame “lanzar rutina”. De este modo, se refuerza la funcionalidad y el usuario no tiene porqué saber cómo usar el widget aunque evidentemente ofrece un paso menos que por aplicación. En la página de Google Play donde se cuelgue la aplicación, se remarcará especialmente la necesidad del widget.

E.2 Tarea 2: Intentar lanzar una aplicación sin éxito

Esta tarea tenía una única etapa que consistía en lanzar una aplicación de notas diciendo correctamente la palabra clave, pero la palabra clave no estaba asociada a la aplicación que decía el enunciado.

El objetivo de esta tarea ha sido ver la reacción de los participantes ante un error y que se demuestre la comprensión del mismo.

E.2.1 Resumen de la tarea por participantes y perfiles

Participantes	EFECTIVIDAD		EFICIENCIA	SATISFACCIÓN
	Porcentaje de finalización	Grado problemas	Tiempo tarea	Complejidad
Participante 1	100%	0	14s	1
Participante 3	100%	0	14s	1
Participante 4	100%	0	16s	2
Participante 6	100%	0	30s	1
MEDIA PERFIL 1	100%	0	18,5s	1,25
Participante 2	100%	0	25s	2
Participante 5	100%	0	32s	1
Participante 7	100%	0	20s	1
Participante 8	100%	0	26s	1
MEDIA PERFIL 2	100%	0	60,25s	1,25

Al igual que en la tarea 1, es en la métrica de eficiencia donde se muestra una ligera diferencia. Al ser esta tarea, tan similar a la 1, el factor de aprendizaje influye haciendo que baje el tiempo por tarea.

Es interesante destacar que 3 de los participantes, no se conformaron con haber entendido el mensaje y accedieron a la opción de menú “Consulta rutina”, para averiguar si la aplicación tenía asociada una palabra clave, y en tal caso, cual era. También es importante destacar que los 3 participantes pertenecían al perfil 1.

E.2.2 Efectividad

Todos los participantes consiguieron finalizar la tarea sin ningún problema.

TAPADÚ: Un lanzador de aplicaciones por voz

Resultados de las pruebas con usuarios

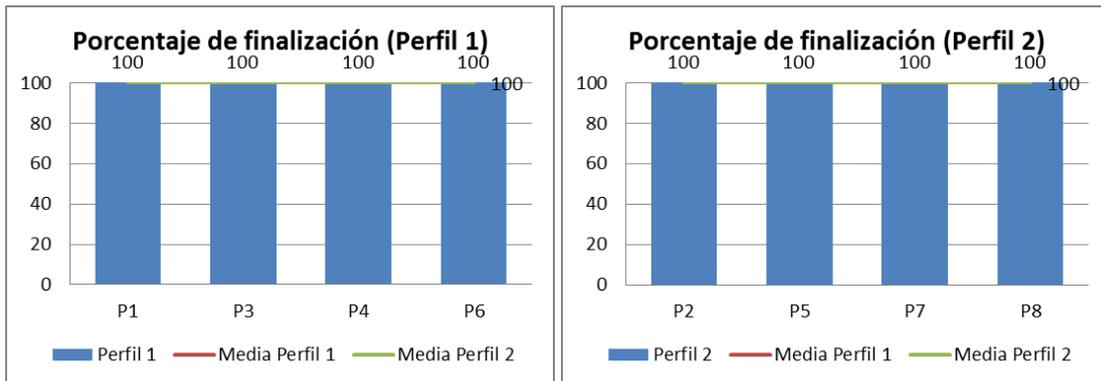


Ilustración 39. Porcentaje de finalización de la tarea 2 por perfiles. La media del perfil 1 y 2 al ser la misma se solapan.

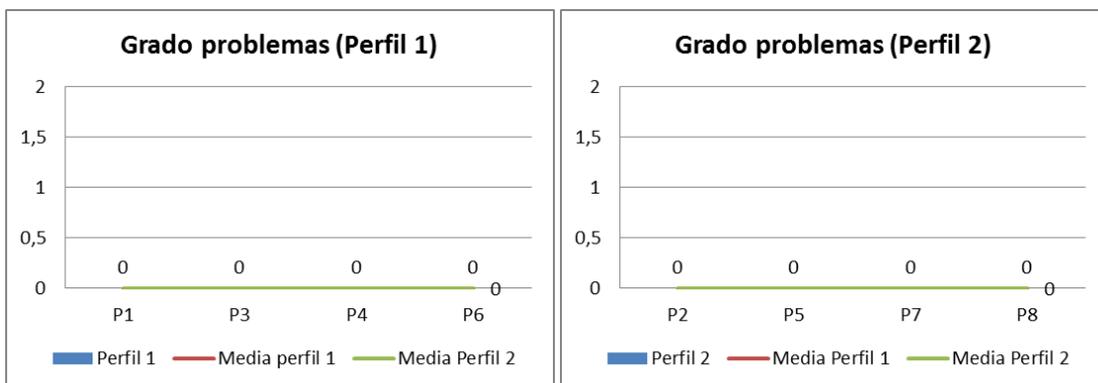


Ilustración 40. Grado de problemas de la tarea 2 por perfiles. La media del perfil 1 y 2 al ser la misma se solapan.

E.2.3 Eficiencia

La gráfica muestra el tiempo que cada participante tardó en realizar y la media por perfil.

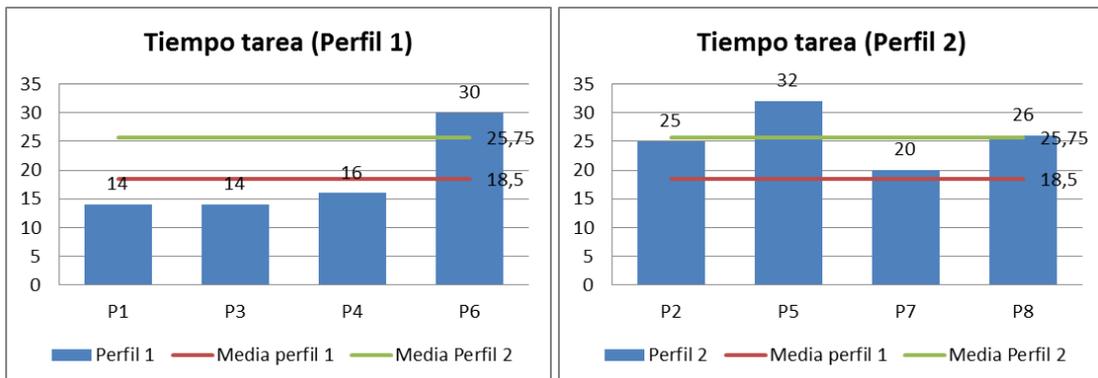


Ilustración 41. Tiempos de la tarea 2 por perfiles.

Al igual que ocurre en la tarea 1, es la única gráfica de esta tarea donde se aprecia una ligera diferencia. Pero es despreciable.

E.2.4 Satisfacción

Como se ve en la siguiente gráfica, los participantes están contentos con el grado de complejidad de la funcionalidad y no se ofrecen diferencias entre los diferentes perfiles.

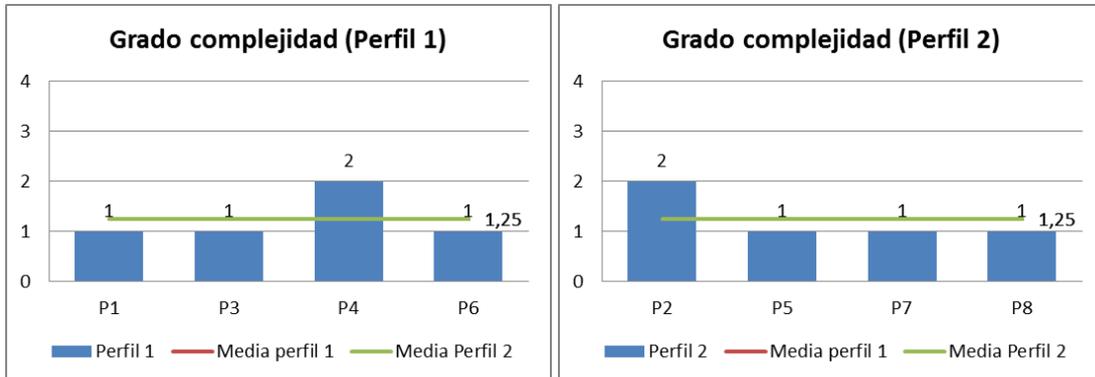


Ilustración 42. Grado de complejidad de la tarea 2. La media del perfil 1 y 2 al ser la misma se solapan.

E.2.5 Problemas detectados

Se ha detectado una mejora en esta tarea:

- **Problema:** Aquellos usuarios que no se han conformado con entender el mensaje, al acceder a la pantalla consulta de rutinas, confundían la palabra clave con el nombre de la aplicación.

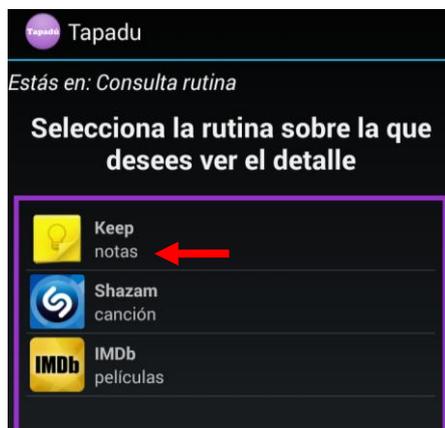


Ilustración 43. Pantalla consulta rutina

Solución: Añadir el literal “Palabra clave” antes de la palabra clave propiamente dicha.

E.3 Tarea 3: Crear una nueva rutina

Esta tarea tenía dos etapas:

- Generar una rutina, es decir, asignarle a una aplicación una palabra clave a elección del participante.
- Lanzar la aplicación con la nueva rutina.

Esta es la primera tarea en la que el usuario tiene que usar la aplicación y no sólo el widget.

E.3.1 Resumen de la tarea por participantes y perfiles

Participantes	EFECTIVIDAD		EFICIENCIA	SATISFACCIÓN
	Porcentaje de finalización	Grado problemas	Tiempo tarea	Complejidad
Participante 1	100%	0	49s	1
Participante 3	100%	0	42s	1
Participante 4	100%	0	97s	1
Participante 6	100%	0	75s	1
MEDIA PERFIL 1	100%	0	65,75s	1
Participante 2	50%	2	259s	3
Participante 5	75%	1	227s	1
Participante 7	50%	2	633s	1
Participante 8	100%	0	120s	1
MEDIA PERFIL 2	63,75%	1,25	309,75s	1,5

Esta es la tarea donde mayor diferencia se aprecia entre los perfiles 1 y 2 en las métricas de efectividad, eficiencia y satisfacción.

E.3.2 Efectividad

Todos los participantes del perfil 1 consiguieron finalizar la tarea, mientras que los usuarios de perfil 2 sólo hubo uno que lo consiguió sin errores. A continuación se detallan los problemas con los que se encontraron:

- A dos participantes, al seleccionar la aplicación, no asociaron que era una lista que se podía recorrer y eligieron una aplicación al azar. Son los participantes que no consiguieron finalizar la tarea.
- A tres participantes, les costó entender que tenían que usar la aplicación para crear la nueva rutina y no el widget.
- Un participante intentó asociar la palabra clave sin haber pulsado previamente el botón.

Finalmente, cabe destacar que una vez dada de alta la nueva rutina, la supieron lanzar de forma rápida a través del widget.

TAPADÚ: Un lanzador de aplicaciones por voz

Resultados de las pruebas con usuarios

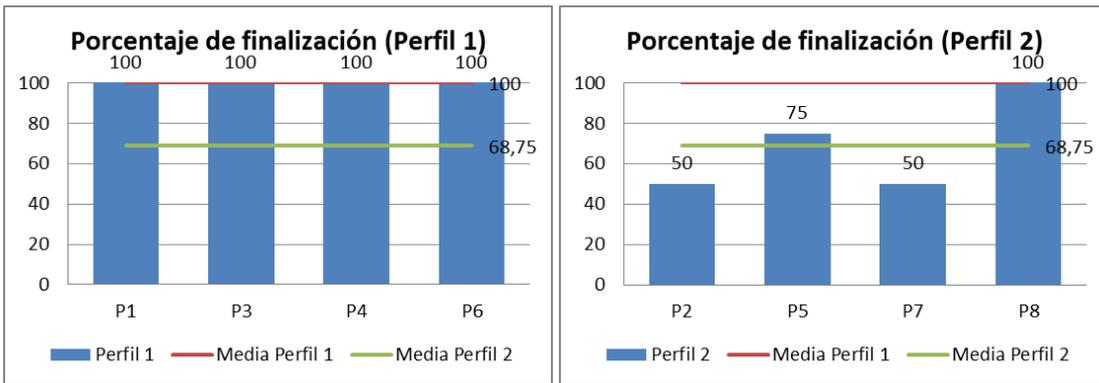


Ilustración 44. Porcentaje de finalización de la tarea 3 por perfiles.

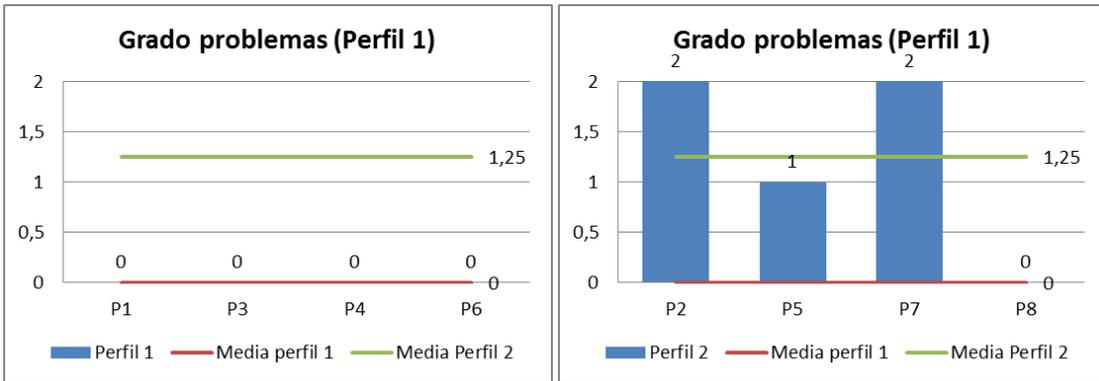


Ilustración 45. Grado de problemas de la tarea 3 por perfiles.

E.3.3 Eficiencia

Debido a los problemas descritos en el apartado E.3.2, los participantes necesitaron de un mayor tiempo para finalizar la tarea, quintuplicando la media del perfil 1.

La gráfica muestra el tiempo que cada participante tardó en realizar y la media por perfil.

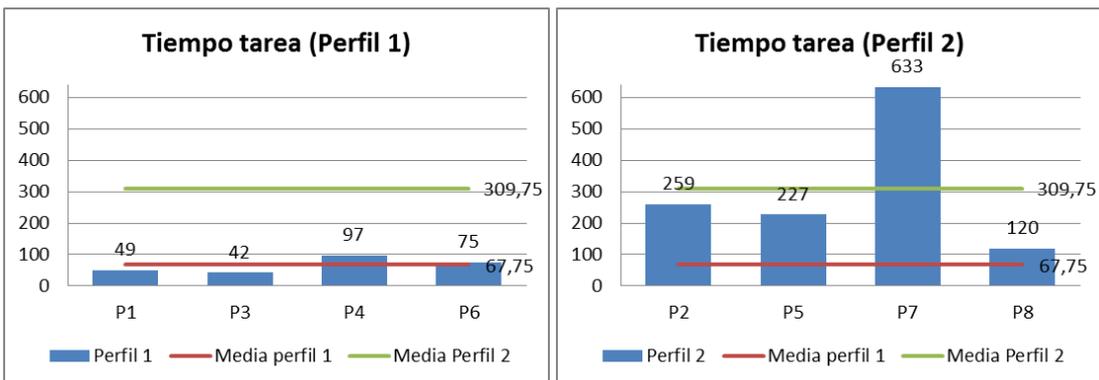


Ilustración 46. Tiempos de la tarea 3 por perfiles.

E.3.4 Satisfacción

Es importante destacar, que a pesar de las dificultades con las que se encontraron los participantes del perfil 2, cuando terminaban con la tarea, la calificaron como no compleja (a excepción del participante 2). Esto puede estar atribuido a que una vez aprendido el recorrido que deben seguir, les parece lógico y comprenden mejor las funcionalidades que ofrece la aplicación.

Para los participantes del perfil 1, no tuvo ninguna complejidad.

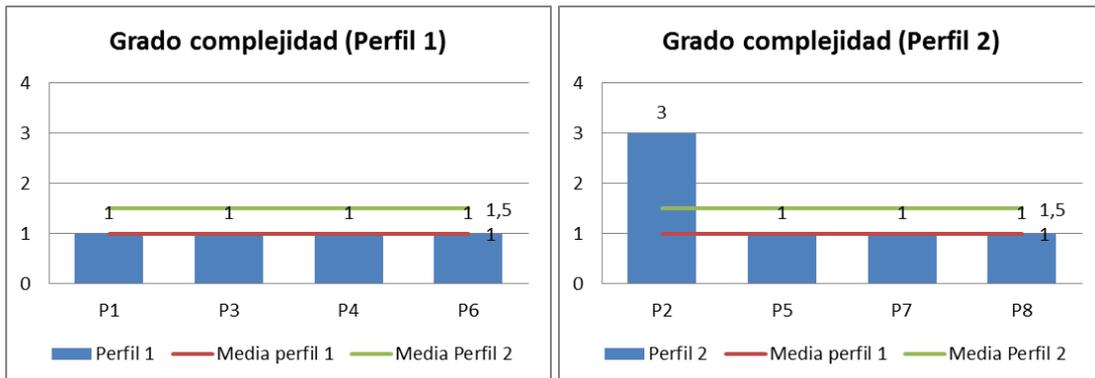


Ilustración 47. Grado de complejidad de la tarea 3 por perfiles.

E.3.5 Problemas detectados

Se han detectado dos mejoras en esta tarea:

- **Problema:** 2 participantes no supieron asociar que la pantalla donde se mostraban las aplicaciones instaladas, se trataba de una lista que se podía recorrer. Ocurrirá lo mismo para la pantalla “Consulta rutinas”



Ilustración 48. Pantalla seleccionar rutina y consultar rutina

Solución: Añadir en los literales, la palabra lista. Es decir: “Selecciona la rutina de la lista sobre la que desees ver el detalle” y “Selecciona una aplicación de la lista”. Además, se modificará el componente para que se muestre claramente un scroll.

- **Problema:** Uno de los participantes, intentó asociar la palabra clave sin pulsar el botón debido al literal “Di la palabra clave”.

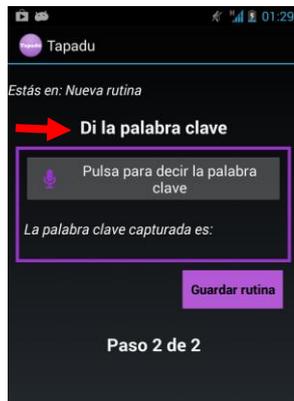


Ilustración 49. Pantalla en la que se asocia la palabra clave a la aplicación

Solución: Modificar el literal para que aparezca “Especificar la palabra clave”.

E.4 Cuestionario de satisfacción de la aplicación

Para medir el grado de satisfacción de la aplicación, se ha usado el cuestionario SUS y a continuación se puede ver las calificaciones obtenidas por cada uno de los participantes, teniendo en cuenta que la puntuación va del 0 a 100.

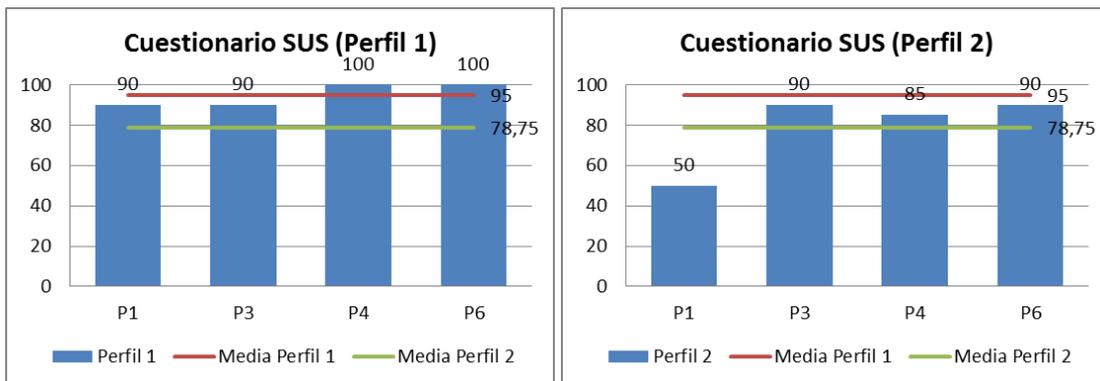


Ilustración 50. Cuestionario SUS por perfiles.

Tras el análisis del cuestionario se puede concluir:

- Que los usuarios están contentos con la aplicación con un 86,875 de media en el cuestionario SUS.
- A pesar de que el perfil 2, tuvo bastantes más problemas, su media es de 78,75, con lo que queda constancia de su grado de satisfacción.
- La aplicación es fácil de usar y se consiguen las funcionalidades en pocos pasos.
- A los usuarios les parece interesante y la usarían de forma frecuente.