

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
ESCUELA DE POSGRADO



**CASO DE ESTUDIO: DESARROLLO DE UNA APLICACIÓN
MÓVIL PARA ANDROID PARA LA MEDICIÓN DEL NIVEL DE
RUIDO INTEGRANDO METODOLOGÍAS ÁGILES Y TÉCNICAS
DE USABILIDAD**

Tesis para optar el grado de Magíster en Informática que presenta
DANIEL VICTORIA DIONICIO

Dirigido por
CLAUDIA ZAPATA DEL RÍO

San Miguel, 2014

Página 1 de 93

1 Resumen

La usabilidad y las metodologías ágiles son dos enfoques orientados al usuario que contribuyen a mejorar el producto desarrollado, pero existen ciertas tensiones para integrarlos como parte de un mismo proceso. En esta tesis, se observa, a través de un caso de estudio, la forma de integrar las prácticas de usabilidad dentro de la metodología ágil más usada, Scrum, en el contexto del desarrollo de una aplicación para medir el ruido. Los resultados del estudio indican que prácticas de usabilidad como el prototipado, encuestas y evaluaciones heurísticas se pueden integrar como tareas dentro de las iteraciones de Scrum mientras que las pruebas de usabilidad deben ser un proceso separado posterior a la ejecución de cada iteración. Asimismo, se observa que la integración de técnicas de usabilidad a lo largo de las iteraciones contribuye con la reducción de la gravedad de los problemas de usabilidad.

2 Abstract

Usability and agile are two user-centered approaches that contribute to improving software development. However, there are certain factors that generate tension when integrating both in the same process. This thesis, explores through a case study the integration of usability practices with the most widely adopted agile framework, Scrum, during the development of a mobile noise meter. The results of the study indicate that certain usability practices such as prototyping, surveys and heuristic evaluation can be integrated as tasks during each Scrum iteration, while usability testing must be a separate process executed after each iteration. The study also found that integrating usability techniques throughout the process contributes to the reduction of the severity of usability problems.

Keywords: Agile, Usability, Scrum, Android, mobile, user-centered design, case-study

Tabla de contenidos

1	Resumen	2
2	Abstract.....	2
3	Introducción	7
3.1	Definición del problema.....	8
3.1.1	Causas	8
3.1.2	Problema:	8
3.1.3	Consecuencias:.....	8
3.2	Objetivo general	9
3.2.1	Objetivos específicos.....	9
3.2.2	Resultados esperados	9
3.2.3	Consideraciones y alcance.....	9
3.3	Marco conceptual.....	10
3.3.1	Usabilidad.....	10
3.3.2	<i>Scrum</i>	13
3.3.3	Entorno móvil.....	15
3.3.4	Medición de la usabilidad en aplicaciones móviles.....	17
3.3.5	Sobre el ruido	18
3.4	Estado del arte: Integración entre usabilidad y metodologías ágiles... 22	
3.5	Metodología de desarrollo.....	25
4	Requisitos y concepción del diseño.....	28
4.1	Mapa de la funcionalidad de las aplicaciones existentes	28
4.2	Evaluación heurística	29
4.2.1	Perfil de los evaluadores	30
4.2.2	Resultado	31
4.3	Prototipado en papel	35
4.3.1	Objetivos de la prueba.....	36
4.3.2	Resultados.....	36
5	Validación de patrones de diseño.....	40
5.1	Lista de patrones de Android y su posible uso.....	40
5.2	Resultados de la prueba	41
6	Validación final del producto.....	44

6.1	Objetivo de la prueba	45
6.2	Preguntas de investigación	45
6.3	Características de los participantes.....	46
6.4	Metodología	46
6.5	Estructura de la prueba	47
6.6	Resultados de la prueba	48
7	Discusión de resultados.....	50
8	Conclusiones y Trabajos futuros.....	54
8.1	Conclusiones.....	54
8.2	Trabajos futuros	55
9	Referencias bibliográficas.....	57
	Anexo A: Prototipo en Papel	64
	Anexo B.....	65
	Introducción a la prueba.....	65
	Prueba <i>Background</i>	66
	Prueba Pre-Test (Entrevista)	67
	Test 68	
	Material referencial.....	69
	Escala de Decibeles	69
	Límite permitido según el Ministerio de Ambiente.	69
	Mapa de San Isidro zonificado.....	70
	Post-Test.....	70
	Consentimiento informado	71
	Anexo C	72
	Matriz de tareas	72
	Tarea 1: Compartir una molestia al lado de tu hogar.....	72
	Tarea 2: Denunciar un abuso	73
	Tarea 3: Medir el ruido.....	74
	Cuestionario Post-Test.....	74
	Anexo D: Encuesta.....	75

Anexo E: <i>Sprint Backlog</i>	77
Anexo F: <i>Burndown chart</i>	82
Anexo G: Script de prueba final.....	83
Tareas.....	83
Post-Test.....	84
Anexo H: <i>Usability Backlog</i>	86

Tabla de Ilustraciones

Ilustración 1: Mapa de proceso de Diseño Centrado en el Usuario. Fuente: Usability.gov	11
Ilustración 2: Curvas de igual intensidad. Fuente: ISO 226:3003 citado por Wikipedia.....	19
Ilustración 3: Prototipo inicial de la aplicación	39
Ilustración 4: Flujo Scrum adaptado. Fuente: Elaboración propia	50
Ilustración 5: Problemas al final de cada iteración	52
Ilustración 6: Prototipo inicial de la aplicación en papel	64
Ilustración 7 Escala de decibeles	69
Ilustración 8 Mapa de San Isidro como referencia	70
Ilustración 9: Historias de usuario vs días – Sprint 0.....	82
Ilustración 10: Historias de usuario vs días - Sprint 1.....	82
Ilustración 11: Historias de usuario vs días - Sprint 2.....	82

Tabla de Tablas

Tabla 1: Modelo de Usabilidad Propuesto por Harrison et al comparado con los modelos existentes de usabilidad.....	16
Tabla 2: Máximo nivel permitido de dB(A).....	20
Tabla 3: Proceso Scrum para un desarrollador	26
Tabla 4: Elementos de Scrum incluídos	27
Tabla 5 Sprint Backlog del Sprint 0	28
Tabla 6 Mapa de funcionalidades de las aplicaciones existentes	29
Tabla 7: Expertos que participaron en la evaluación heurística	31
Tabla 8 Problemas identificados por evaluación heurística.....	34

Tabla 9 Lista de patrones Android.....	41
Tabla 10: Características de los participantes.....	46
Tabla 11: Evolución de problemas de usabilidad	49
Tabla 12: Elementos del product backlog después del product backlog grooming	51
Tabla 13 Zonas de referencia según normativa peruana	69
Tabla 14: Sprint backlog por sprint.....	81



3 Introducción

En el 2001, se inicia la consolidación del movimiento ágil en el campo del desarrollo de software, a través de la creación del Manifiesto Ágil, el cual propone:

*“Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan.” [1]*

Las metodologías ágiles se enfocan en entregar valor al cliente y en la creación de un producto tangible, por encima de crear documentación extensa y seguir un proceso predefinido solo por el hecho de tener un proceso. [2]

Por otro lado, la ingeniería usabilidad es una disciplina que se enfoca en técnicas para desarrollar productos que permitan al usuario hacer lo que él desea, en la forma en que lo desea o espera hacerlo, sin impedimentos, dudas o preguntas. [3]

El foco de la usabilidad está en utilizar técnicas que permitan desarrollar interfaces y sistemas más usables a través de un proceso centrado en el usuario. [4]

El presente trabajo busca observar, a través de un caso de estudio, la integración de ambos enfoques. Si bien el centro de ambos es el usuario, existen tensiones al integrarlos: Las metodologías ágiles se enfocan en tener siempre un producto entregable al final de cada iteración mientras que para implementar técnicas de usabilidad es necesario realizar cierta investigación inicial del usuario y trabajo preparatorio [5] [4]

El caso de estudio explora la problemática de la integración en el contexto del desarrollo de aplicaciones móviles. Este uno de los sectores más crecientes con usuarios que pasan un promedio de 2 horas 20 minutos al día usando aplicaciones móviles [6].

Para el caso de estudio, se utiliza *Scrum* que es un *framework* de desarrollo utilizado por el 73% de proyectos ágiles [7]. En *Scrum*, una persona con el rol de “*Product Owner*” (dueño del producto) trabaja de cerca con el equipo de

desarrollo para priorizar, identificar la funcionalidad del sistema y formar un “*Product Backlog*”, que es un listado de todas las funcionales y tareas necesarias para construir el sistema. El trabajo se da en iteraciones de tiempo fijo llamadas *sprints*. Al inicio de cada iteración, se escogen las funcionalidades a desarrollar del *Product Backlog*. Dentro de un *sprint*, generalmente no está permitido cambiar las funcionalidades escogidas, pero sí refinar su especificación. Al final de cada iteración se debe contar con un producto potencialmente entregable. [8] [9]

3.1 Definición del problema

A continuación se plantea el problema a estudiar a través de un árbol de problemas.

3.1.1 Causas

- En el contexto móvil, el usuario presta menos atención a la aplicación por la concurrencia de actividades que realiza. [10] [11]
- Existe una variedad de formas de interactuar con los dispositivos móviles [12] [13]
- Las interfaces de usuario son limitadas en cuanto a la cantidad de elementos que puedan mostrar [12] [14] [15]
- No existe una propuesta que indique como integrar las metodologías ágiles con las técnicas de usabilidad para los casos en los que no se disponga de un equipo dedicado de usabilidad [16]
- Hay una dificultad para integrar los procesos de usabilidad con las metodologías ágiles. La investigación actual indica que respetar los valores ágiles depende de la actitud de cada equipo [17]

3.1.2 Problema:

La usabilidad de las aplicaciones móviles es menor a la de las aplicaciones de escritorio y no es un uniforme.

3.1.3 Consecuencias:

- Hay una baja intención de uso luego de que un usuario experimenta un problema de usabilidad. [18]
- La memorabilidad de las aplicaciones móviles es menor a la de las aplicaciones de escritorio. [12]

- La usabilidad de una misma aplicación varía de un dispositivo a otro generando experiencias negativas en el usuario. [12]

3.2 Objetivo general

Determinar a través de un caso de estudio el impacto en la usabilidad de una aplicación móvil de incorporar las buenas prácticas de la plataforma móvil y de diseño orientado a usuario dentro de un proceso de desarrollo ágil usando *Scrum*.

3.2.1 Objetivos específicos

- Validar los beneficios de incluir un *sprint* inicial previo al inicio de desarrollo de la aplicación (*Sprint 0*) dentro de un proceso de desarrollo que incluye técnicas de usabilidad.
- Validar si los patrones de diseño de la plataforma mitigan los problemas de usabilidad de las aplicaciones móviles.
- Validar que técnicas de usabilidad se pueden integrar dentro de las iteraciones normales de *Scrum* y en qué partes del proceso en el contexto de un solo desarrollador.

3.2.2 Resultados esperados

- Propuesta de diagrama de flujo de *Scrum* adaptado a un único desarrollador e integrando prácticas de usabilidad.
- Lista de técnicas de usabilidad documentado con las dificultades y beneficios de usarlo dentro de un contexto ágil.
- Lista de patrones de diseño de la plataforma con los problemas de usabilidad que generaron y/o evitaron.

3.2.3 Consideraciones y alcance

El alcance del presente trabajo se centra en la plataforma Android, que tiene alrededor de 84% de participación del mercado de *smartphones* [19].

La aplicación a desarrollar será una aplicación móvil para medir el ruido de uso general. Durante el desarrollo del caso se identificó el posible uso industrial pero no se consideró dentro del alcance.

3.3 Marco conceptual

A continuación se presentan los conceptos de usabilidad y metodologías ágiles que se utilizarán para el desarrollo del caso.

3.3.1 Usabilidad

Usabilidad: Según [20], la definición de usabilidad es: la efectividad, eficiencia y satisfacción con la que determinados usuarios alcanzan las metas especificadas en ambientes particulares.

Diseño centro en el usuario: [3] define el diseño centrado en usuario como las técnicas, procesos, métodos y procedimiento para diseñar productos y sistemas usables así como la filosofía que pone al usuario en el centro de este proceso de diseño.

3.3.1.1 Proceso y técnicas de usabilidad

En [21], Nielsen indica que la ingeniería de usabilidad no es una actividad única que se realiza antes del lanzamiento del producto sino un conjunto de actividades que se realizan a lo largo del ciclo de vida del producto.

Nielsen plantea las siguientes etapas como parte del modelo de ciclo de vida de la ingeniería de usabilidad:

1. Conocer al usuario.
2. Análisis competitivo.
3. Establecer metas de usabilidad.
4. Diseño paralelo: Desarrollar múltiples versiones de una interfaz en paralelo, comparar los resultados y tomar los mejores aspectos de cada una.
5. Diseño participativo: Involucrar al usuario como parte de desarrollo y permite que los desarrolladores tengan un grupo de usuarios disponible al cuál hacer consultas.
6. Diseño coordinado de toda la interfaz del usuario: Coordinar y centralizar los aspectos de la interfaz para mantener la consistencia a lo largo de la aplicación.
7. Aplicar guías y realizar análisis heurístico.
8. Prototipado.
9. Pruebas empíricas.

10. Diseño iterativo: La aplicación debe ser desarrollada en ciclos de prueba y corrección en base a los resultados de las pruebas empíricas.

11. Recolectar retroalimentación del uso en campo

El ciclo de vida planteado es similar al planteado por [22] como ciclo de vida del desarrollo centrado en el usuario

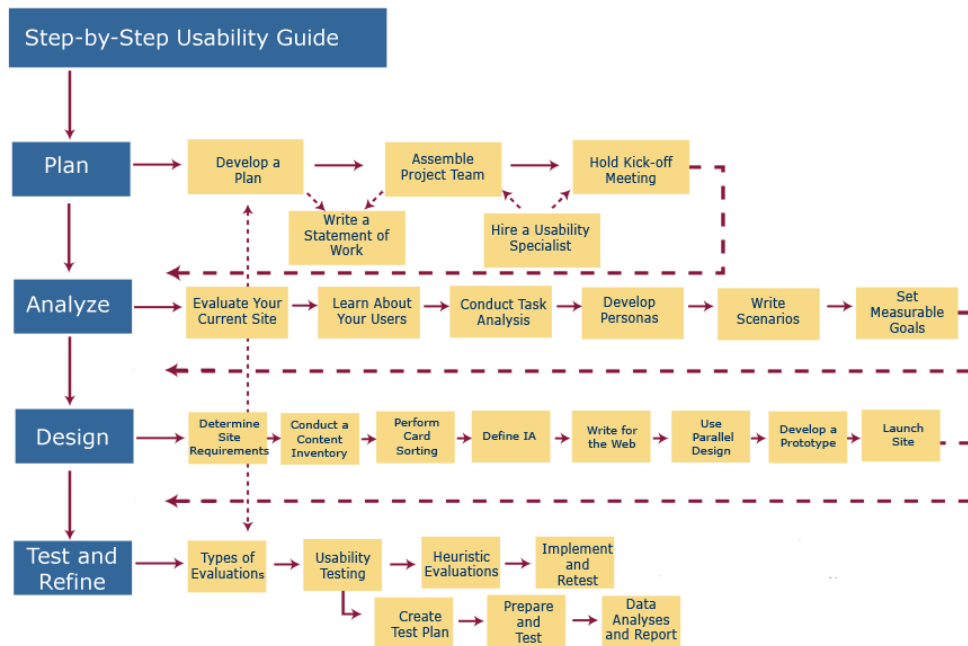


Ilustración 1: Mapa de proceso de Diseño Centrado en el Usuario. Fuente: Usability.gov

En los siguientes puntos, se presenta los aspectos comunes más importantes de los dos mapas de procesos planteados.

3.3.1.1.1 Conocer al usuario

En ambos mapas de proceso, la primera parte del análisis consistente conocer al usuario. Esto se puede lograr a través de la investigación etnográfica [3] que toma técnicas de la antropología para poder obtener datos de los usuarios a través de [3] [23] :

- Grupos focales: Discusiones moderadas de grupos de usuarios.
- Encuestas
- Entrevistas contextuales: Entrevistas realizadas con los usuarios mientras desarrollan sus tareas.
- Entrevistas individuales.

La información recolectada puede ser a su vez condensada en “personas” que son representaciones de clústeres de usuarios que exhiben patrones de comportamiento similares [24].

3.3.1.1.2 Pruebas de usabilidad

Generalmente, una prueba de usabilidad consiste en hacer que un grupo de participantes realice una serie de tareas típicas con una aplicación o un prototipo de la misma ante un grupo de observadores que toma notas. El grupo de participantes debe ser representativo de los usuarios finales de la aplicación. [25]

Según [26], las pruebas con usuarios reales son el método de usabilidad fundamental dado que permiten obtener retroalimentación directa del usuario.

En [3], se plantea que dichas pruebas se pueden realizar en distintos puntos del ciclo de vida del desarrollo del producto:

1. Durante el prototipado, diseño y pruebas: Estudio exploratorio o formativo, Prueba de evaluación
2. Durante las pruebas finales antes del lanzamiento de la aplicación: Prueba de verificación o validación
3. En todo el ciclo de vida para compararse con productos competidores: Prueba de comparación

3.3.1.1.3 Prototipado

El prototipado es un método para diseñar, crear, probar y comunicar interfaces de usuario de manera independiente a la plataforma en la que se desarrolla.

Un prototipo es una representación visual en papel de cómo se podría ver el sistema final, la cual puede usarse para facilitar al usuario visualizar el concepto de aplicación y así elucidar mejor los requerimientos de la misma [27].

Los prototipos también pueden usarse para una variación de las pruebas de usabilidad donde el usuario realiza tareas realistas interactuando con una versión de papel de la interfaz. Esta es manipulada por otra persona que actúa como la computadora [28].

Esta técnica permite que se realicen pruebas de usabilidad de manera temprana en el ciclo de desarrollo, lo cual permite entender mejor al usuario y realizar cambios con un menor costo al de programar la aplicación [21].

3.3.1.1.4 Evaluación heurística

Una evaluación heurística permite que un grupo de expertos compare la interfaz de una aplicación con una serie de principios aceptados de usabilidad e identifique los incumplimientos que se presentan frente a estas heurísticas. [29]

Las heurísticas más usadas son las definidas por Jakob Nielsen [26]:

- Visibilidad del estado del sistema.
- Correspondencia entre el sistema y el mundo real.
- Libertad y control del usuario.
- Consistencia y estándares.
- Prevención de errores.
- Minimizar la carga de memoria del usuario.
- Flexibilidad y eficiencia.
- Diseño estético y minimalista.
- Ayudar al usuario al reconocer, diagnosticar y recuperarse de errores.
- Ayuda y documentación.

3.3.1.1.5 Diseño iterativo

En ambos documentos, hay un foco importante en el diseño iterativo que permite aprender a lo largo del camino y transformar el producto para que se adapte mejor a la expectativa del usuario.

3.3.2 Scrum

Las metodologías ágiles son un grupo de métodos de desarrollo de software que enfatizan [30]:

- La colaboración entre el equipo de programación y los expertos de negocio
- La comunicación frente a frente.
- La entrega rápida de valor para el negocio.
- Equipos auto-organizados.
- La respuesta rápida al cambio.

A continuación se describe la metodología *Scrum* según [9].

Scrum es un enfoque ágil para el desarrollo de producto y servicio. El trabajo se realiza en *sprints* que son periodos de tiempo fijos de entre 1 a 4 semanas durante los cuales un equipo multidisciplinario realiza todo el trabajo. Al final de cada uno periodos el equipo deberá haber construido un producto potencialmente entregable.

En *Scrum*, las historias de usuario que debe implementar una aplicación se colocan en un *product backlog* que consiste en una lista priorizada de funcionalidades. Las actividades que se realizan en *Scrum* y que permiten avanzar el *product backlog* son:

- **Product backlog grooming:** Actividad en la cual se prioriza, agrega o retira nueva funcionalidad del *product backlog*.
- **Sprint planning:** Fase de planeamiento previa a cada iteración donde se define que funcionalidades del *product backlog* se construirán.
- **Sprint:** Una iteración durante la que se construye el producto.
- **Daily scrum:** Reunión diaria del equipo con una duración de 15 minutos, durante la cual se revisa la mejor forma de avanzar hacia la solución, se resuelven bloqueantes y se sincroniza el equipo.
- **Sprint execution:** Actividad de construcción del equipo de desarrollo.
- **Sprint review:** Actividad en la que se inspecciona y adapta el producto en construcción al final de cada iteración.
- **Sprint retrospective:** Actividad en la que se revisa y adapta el proceso de construcción al final de cada iteración.

Los roles en esta metodología son:

- **Product Owner:** Es la persona responsable del liderazgo del producto y definir qué funcionalidades se deben construir, en qué orden se deben construir y priorizar.
- **Scrum Master:** Es la persona que provee de liderazgo al equipo, actúa como coach de los principios de *Scrum*, ayuda a resolver bloqueantes y a que el equipo alcance su mayor desempeño.

- **Equipo de Desarrollo:** Es el equipo de personas auto-organizado que es responsable de diseñar, construir y probar el producto deseado.

3.3.3 Entorno móvil

El mercado actual de los dispositivos móviles está dominado principalmente por los sistemas operativos Android y iOS [31], [19], [32]. Cada uno plantea a sus desarrolladores una serie de guías de diseño de aplicaciones [33], [34].

El hecho de ser plataformas nuevas implica que los desarrolladores experimenten con las interfaces de usuario, y en muchos casos, no se toman en consideración las mejores prácticas de usabilidad desarrolladas a lo largo del tiempo [13].

Según [35], la usabilidad de las aplicaciones de software desarrolladas para dispositivos móviles se ve afectada por distintos factores que no se encuentran presentes en los computadores de escritorio tradicional como son:

- El contexto móvil,
- El tamaño del dispositivo
- La variedad de resoluciones de los dispositivos
- La capacidad de procesamiento
- Los métodos de ingreso de datos.

En [12], se propone un modelo de usabilidad que incluye como atributo medible la carga cognitiva. El estudio indica que el usuario se encuentra realizando muchas tareas en paralelo mientras utiliza una aplicación móvil e identifica 3 factores que deben ser analizados durante las mediciones de usabilidad (usuario, contexto y tarea).

ISO	Nielsen	Modelo propuesto
Efectividad	Efectividad	Efectividad
Eficiencia	Eficiencia	Eficiencia
Satisfacción	Satisfacción	Satisfacción
	Facilidad de aprendizaje	Facilidad de aprendizaje
	Recordación	Recordación
	Errores	Errores
		Carga cognitiva

Tabla 1: Modelo de Usabilidad Propuesto por Harrison et al comparado con los modelos existentes de usabilidad.

Adicionalmente, la sesión promedio de uso de aplicación móviles es de 72 segundos [36], lo cual refuerza la importancia de la carga cognitiva y del contexto multitarea donde se desarrolla la interacción con los dispositivos móviles.

De acuerdo con [10], la atención prestada a las aplicaciones móviles se ve afectaba seriamente por el contexto en el que se usa: las tareas del contexto compiten por los recursos de atención del usuario. Esto puede reducir el tiempo de atención de 16 segundos, que es un valor obtenido en laboratorio, a pocos segundos en situaciones móviles complicadas.

En [37], el estudio indica que la comprensión lectora en un dispositivo con la pantalla del tamaño de un iPhone es 48% de la comprensión alcanzada en una computadora de escritorio.

Todos estos factores influyen en que el ratio de éxito de una aplicación móvil sea menor que el de una aplicación de escritorio.

De acuerdo con [38], la mayoría de desarrollos móviles se realizan utilizando metodologías ágiles incluyendo *Scrum* y desarrollo orientado a pruebas, inclusive desarrolladores individuales adoptan prácticas de *Scrum*.

3.3.4 Medición de la usabilidad en aplicaciones móviles

La usabilidad en aplicaciones móviles se mide mediante 3 técnicas principalmente. [14]:

- Experimentos controlados
- Estudios de campo
- Mediciones *hands-on*.

De acuerdo con la revisión de literatura realizada en [12], cerca del 60% de estudios considerados utilizó experimentos controlados mientras que un 27% utilizó estudios de campos.

3.3.4.1 Experimentos controlados

Los experimentos controlados son aquellos que se llevan a cabo en un laboratorio de usabilidad donde los usuarios son estudiados mientras interactúan con un dispositivo móvil. La principal ventaja es que se puede monitorear directamente al usuario, mientras que la principal desventaja es que el usuario es aislado de su entorno. Esto genera que algunos factores como el contexto móvil, la conectividad [14] no sean tomados en consideración.

Para los experimentos controlados tanto [39] [40] recomiendan que los usuarios utilicen sus propios dispositivos móviles incluyendo la red de sus operadores móviles dado que esto permitirá identificar más problemas de usabilidad que si se utiliza un solo dispositivo estándar.

En [40], se plantea que no existe la misma cantidad de herramientas para monitorear a un usuario móvil en un entorno de laboratorio como si lo existe para las computadoras de escritorio. Este mismo punto también es presentando como un vacío en [41].

Los autores en [41] presentan una revisión del estado del arte en la cual se indica que la gran mayoría de problemas de usabilidad son detectables por los experimentos controlados. Si bien se excluyen factores como el contexto móvil, el tiempo de evaluación durante experimentos de campos se ve reducido considerablemente, lo cual niega el beneficio de incluir estos factores.

3.3.4.2 Estudios de campo

Los estudios de campo son un método general para recolectar información acerca del usuario, sus necesidades y los requerimientos de producto. Esta recolección involucra observación y entrevistas [14].

De acuerdo con [42], las principales ventajas de realizar un estudio de campo son la capacidad de recolectar grandes cantidades de datos en el contexto del usuario y la alta “validez ecológica”, mientras que las desventajas son los sesgos desconocidos, y la dificultad para generalizar y validar externamente; es decir, un bajo nivel de control.

Tal como concluye [42], ambos tipos de estudios son necesarios, lo importante es saber cuándo aplicar cada uno de ellos.

3.3.5 Sobre el ruido

La contaminación sonora está asociada con varios problemas de salud: reducción de la capacidad auditiva, alteración del descanso y sueño; efectos negativos en la salud mental y en el desempeño [43].

Para poder medir la contaminación sonora, se utilizan los decibeles de la presión de sonido o dB SPL (*Sound Pressure Level* por sus siglas en inglés).

El oído humano, que tiene un rango de audición entre 20Hz y 20kHz, percibe con menor intensidad sonidos de baja frecuencia y con mayor intensidad aquellos de frecuencias superiores. Por este motivo, es necesario aplicar ciertos ajustes al cálculo de decibeles para que sean una verdadera representación del impacto que tienen en las personas [44].

Para poder representar la forma subjetiva en la que se perciben los sonidos, se ha ideado una técnica en la cual se iguala la intensidad de un sonido de 1 kHz a la intensidad de un sonido a una frecuencia dada. La intensidad de ambos sonidos es la cantidad de decibel SPL del sonido de 1000 Hz.

Por ejemplo, si un sonido de 20Hz produce 60dB pero es igual de intenso que un sonido de 20dB de 1Khz, ambos tendrán la misma intensidad. Esta intensidad se mide en una unidad llamada “*phon*” y para este caso ambos sonidos generan 20 *phons*. [45]

A continuación se muestran las curvas de igual intensidad para 20, 40, 60, 80 y 100 phon.

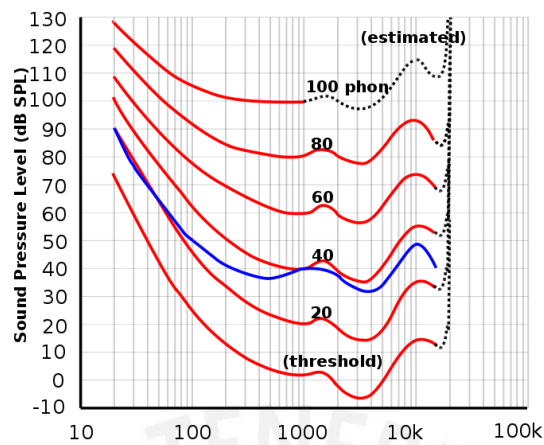


Ilustración 2: Curvas de igual intensidad. Fuente: ISO 226:3003 citado por Wikipedia

En función a estas curvas de igual intensidad se han creado curvas de ponderación para poder ajustar el cálculo de los decibeles para la medición del ruido. Las más frecuentes son [46]:

- Ponderación A: Es muy similar a la curva de 30 *phons*, cuando se usa los decibeles se expresan como dB (A).
- Ponderación B: Se usa para niveles intermedios y está basado en la curva de 70 *phons*.
- Ponderación C: Aproxima la respuesta del oído humano a 100 *phons*.
- Ponderación D: Se usa para la medición del ruido producido por los aviones.

La ponderación más usada dado a que se correlaciona con el daño auditivo producido por los ruidos es la Ponderación A [45]. Esta es la métrica usada por el estándar peruano.

3.3.5.1 Normativa peruana

En el caso de Perú, aplican los siguientes estándares dependiendo de la zonificación [47]. Esta es determinada y actualizada por cada uno de los gobiernos locales y distritales.

Zonas de Aplicación	Horario Diurno (07:01 a.m. - 22:00 p.m.)	Horario Nocturno (22:01 p.m. - 07:00 a.m.)
Zona de Protección Especial	50	40
Zona Residencial	60	50
Zona Comercial	70	60
Zona Industrial	80	70

Tabla 2: Máximo nivel permitido de dB(A)

3.3.5.2 Fórmulas de cálculo

Para poder realizar el cálculo de los decibeles usando un micrófono digital es necesario realizar los siguientes pasos [48]:

1. Captura N muestras durante un periodo de tiempo
2. Aplicar una función ventana a la muestra (como Hamming o Blackman Harris)
3. Calcular la transformada de Fourier
4. Aplicar la ponderación A.
5. Realizar el cálculo de decibeles
6. Aplicar una constante de calibración al resultado.

El primer paso es captura N muestras del sonido medido, para ello es necesario seleccionar una frecuencia de muestreo y un periodo de tiempo. Las ventanas más usadas en los sonómetros profesionales son:

- *Fast*: 125 ms
- *Slow*: 1s

Por ejemplo, si se utiliza una ventana de 1s y una frecuencia de muestreo de 8kHz, se tendría 8000 mil muestras. Para el paso 4 es necesario poder separar la señal en las diversas frecuencias. De acuerdo con el teorema de Nyquist, la mayor frecuencia que se puede distinguir es la mitad de la frecuencia de muestreo. Si quisiéramos tener el máximo del rango del oído humano tendríamos que muestrear a 40kHz para poder distinguir hasta 20kHz.

La transformada de Fourier permite convertir las muestras que se han capturado en cada uno de los puntos en el tiempo en los valores de intensidad que produce cada una de las frecuencias. La fórmula asume que se ha tomado un periodo completo de la señal, sin embargo, podría haberse tomado una fracción, lo cual puede generar que algunas frecuencias que no forman parte de la señal aparezcan en el resultado, a esto se le denomina manchado espectral. Por ello, es necesario aplicar una función ventana que minimice este impacto.

Una de las funciones ventana más usadas es la de Hamming. Cada uno de los datos (1 hasta N) es multiplicado por el valor de función ventana que tiene la siguiente forma.

$$w(n) = \alpha - \beta \cos\left(\frac{2\pi n}{N-1}\right), \alpha = 0.54, \beta = 0.46$$

Ecuación 1: Función ventana Hamming

Sobre los datos ajustados se aplica la transformada de Fourier

$$X_k \stackrel{\text{def}}{=} \sum_{n=0}^{N-1} x_n e^{-i2\pi kn/N}$$

Ecuación 2: Formula de la Transformada de Fourier

La mitad de los valores resultantes contiene las frecuencias de la señal y la cantidad de Hz entre cada par de valores resultantes se denomina resolución de frecuencia y se puede calcular de la siguiente forma:

$$\Delta f = \frac{1}{T} = \frac{f_s}{N}$$

Ecuación 3: Resolución de frecuencia

Sobre cada una de las intensidades de las frecuencias es necesario multiplicarla por una fórmula que la ajuste a la ponderación A. [49]

$\alpha_A(f)$

$$= \frac{(3.5041384 \times 10^{16})f^8}{(20.598997^2 + f^2)^2 \times (107.65265^2 + f^2) \times (737.86223^2 + f^2) \times (12194.217^2 + f^2)^2}$$

Ecuación 4: Factor para la ponderación A

Una vez obtenida la transformada de Fourier multiplicada por los pesos de la ponderación A se utiliza la siguiente fórmula para calcular la energía promedio de la señal.

$$\tilde{\varepsilon}_x \approx \frac{2}{N\Delta t} \sum_{k=0}^{N/2} |X_A[k]|^2$$

Ecuación 5: Energía de la señal

En función a la energía de la señal, se calcula la cantidad de decibeles A. El valor C es un valor de decibeles que se obtiene al calibrar el sonómetro y que se suma al valor productor por la fórmula.

$$\text{Nivel de intensidad (dBA)} = 10 \log_{10}(\tilde{\varepsilon}_x) + C$$

Ecuación 6: Fórmula del cálculo de decibeles

3.4 Estado del arte: Integración entre usabilidad y metodologías ágiles

En esta sección, se detalla el estado del arte de la integración de las prácticas ágiles con la usabilidad a través de la presentación del contenido de dos revisiones sistemáticas: una sobre las prácticas de usabilidad utilizadas en metodologías ágiles [50] y otra sobre la integración del trabajo de usabilidad en las prácticas ágiles de desarrollo de sistema [51]

En [50] se identificó que las técnicas más usadas son:

- Prototipado (40% de los estudios), esta también se encontró en casos adicionales revisados [52] [53] [54] [55].
- Indagación individual: (37% de los estudios): Es un método que se realiza comúnmente mediante encuestas, cuestionarios y/o entrevistas que permiten recolectar información de un único usuario.
- Pruebas formales de usabilidad (25% de los estudios) [52]
- Evaluaciones heurísticas (18% de los estudios).

En lo que respecta a la integración del trabajo de usabilidad con ágil, [51] identifica 7 prácticas y las críticas realizadas a cada una:

Prácticas para la ejecución del trabajo

1. *Trabajo inicial (Little Design Up-front):*

Parte del trabajo de diseño se realiza al inicio del proyecto lo cual permite tener un mayor entendimiento a lo largo del ciclo de vida del proyecto. Depende de que no existan restricciones de tiempo y/o presupuesto además de que no garantiza un sistema usable.

2. *Prototipos de baja fidelidad:*

Construir prototipos de baja fidelidad permite tener un mayor entendimiento del usuario sin incurrir en mayores costos. Sin embargo, en algunos casos, el prototipo es muy abstracto como para guiar el diseño del sistema y existe una capacidad limitada de ser evaluado por el usuario. En algunos casos, puede resultar en diseños que no se pueden implementar.

3. *Realizar pruebas entre iteraciones*

Realizar pruebas de usabilidad entre las iteraciones permite que el sistema sea refinado, reduce el re-trabajo y aumenta el compromiso de los usuarios con el sistema, además que solo los usuarios mismos pueden articular sus propias necesidades; sin embargo, se pueden introducir errores de usabilidad en las siguientes iteraciones. Puede generar que los usuarios se familiaricen demasiado con el sistema y que no lo evalúen de manera objetiva.

4. *Trabajo paralelo de desarrolladores y diseñadores.*

Los diseñadores realizan el trabajo una iteración por delante del equipo de desarrollo [56] lo cual permite que ellos tengan retroalimentación en menor tiempo.

Tal como evidencian [16] [54] [53], trabajar en paralelo genera un esfuerzo adicional de integración y en muchos casos se segrega al equipo de diseño del equipo de desarrollo al punto de manejar las tareas en un tableros separado.

Prácticas para la organización del trabajo

5. Hacer que el equipo de diseñadores sea parte del equipo de desarrollo.

Incluye a un experto de usabilidad en el equipo para asegurar que los aspectos de usabilidad sean importantes en el proyecto, implica que deba existir este rol dentro de la organización.

6. Los diseñadores de usabilidad deben integrarse totalmente al equipo de desarrollo

Permite cumplir con el principio de auto organización de los equipos ágiles sin embargo tanto los expertos de usabilidad como los desarrolladores pueden perder objetividad.

7. Los usuarios finales o representantes de ellos deben estar involucrados en el ciclo de vida del proyecto.

Los usuarios se pueden sesgar a medida que siguen participando en el proceso de desarrollo.

Adicionalmente, se identificaron las siguientes prácticas que no se encuentran en la revisión sistemática, en una revisión de la literatura:

- **Diseño en niveles paralelos:** El diseño se realiza en tres niveles paralelos: diseños para la iteración actual o la siguiente, rediseño de los resultados de previas iteraciones y visión global de diseño. [16]
- **Secuencia fase de diseño iterativo y fase de desarrollo iterativo.** Una fase del desarrollo realiza el diseño en múltiples iteraciones y la siguiente fase realiza la codificación de la aplicación, sin embargo no hay bucle de retroalimentación de todo el proceso. [16]
- **Inclusión de un *Product Owner* de usabilidad.** Para el caso particular de *Scrum* [57] plantea tener un *Product Owner* que dé mantenimiento a la visión de experiencia de usuario en lugar de a un *Product backlog* tradicional

La integración de ágil y usabilidad presenta retos adicionales como:

- El foco de ágil se encuentra en entregar código funcionando. Esto puede generar que se priorice implementar funcionalidades por encima de realizar tareas de usabilidad, dado que estas últimas pueden no tener un beneficio inmediato [53].
- La diferencia en el lenguaje utilizado por los diseñadores y los desarrolladores presenta un reto adicional. Muchos de los artefactos producidos son utilizados solo por el equipo de diseño, particularmente cuando los equipos no están situados físicamente en el mismo lugar o no tienen tiempo para participar a tiempo completo en el proyecto [58]

A pesar de estas dificultades, la tasa de defectos es menor si el equipo tiene conocimiento de UX y mucho menor si hay un experto de usabilidad participando e incluso haciendo diseño paralelos [16]

Finalmente, todos los artículos revisados indican como punto clave la colaboración cercana entre el equipo de desarrollo y diseño.

3.5 Metodología de desarrollo

En el presente trabajo, se utilizará la metodología *Scrum* descrita y se integrará como parte de la revisión de cada iteración pruebas de usabilidad que proveerán de retroalimentación para el proceso de *product backlog grooming*.

Tal como plantea [59], es necesario integrar aspectos de usabilidad dentro del proceso *Scrum* e incluir elementos como personas que permitan enriquecer las historias que se plantean en el *product backlog*.

No se considerará tener contar un segundo *product owner* para usabilidad sino que se utilizará la adaptación de *Scrum* a un solo desarrollador que plantea [60] como se observa en la Tabla 3.

Parte del trabajo será identificar en que puntos dentro de las iteraciones se podrán incluir las técnicas de usabilidad, se considerarán las 4 principales técnicas indicadas en el estado del arte.

Característica	Metodología <i>Scrum</i>	Metodología adaptada a un desarrollador
Duración de <i>Sprint</i>	<i>Sprints</i> de 1 a 4 semanas	<u><i>Sprints</i></u> de 1 a 4 semanas
Participantes	<ul style="list-style-type: none"> • <i>Scrum Team</i> • <i>Scrum Master</i> • <i>Product Owner</i> 	<ul style="list-style-type: none"> • Desarrollador • Cliente
<i>Product Backlog</i>	<ul style="list-style-type: none"> • Lista de requerimientos priorizados • Puntos usado para determinar el tamaño de las historias de usuario individuales 	<ul style="list-style-type: none"> • Lista de requerimientos priorizados • Uso opcional de puntos
<i>Sprint backlog</i>	<ul style="list-style-type: none"> • Lista de historias a ser completadas en el siguiente <i>sprint</i> • Duración fija para cada <i>sprint</i> • Cada historia debe ser partida en tareas 	<ul style="list-style-type: none"> • Lista de historias a ser completadas en el siguiente <i>sprint</i> • Duración fija para cada <i>sprint</i> • Es opcional partir las historias
<i>Sprint tracking</i>	<ul style="list-style-type: none"> • <i>Daily Scrum</i> • Gráfico <i>Burndown</i> para medir el progreso 	<ul style="list-style-type: none"> • No hay <i>Daily Scrum</i> • Gráfico <i>Burndown</i> para medir el progreso
<i>Sprint review</i>	<ul style="list-style-type: none"> • El trabajo hecho durante el <i>sprint</i> se presenta y revisa • Se recoge <i>feedback</i> a ser considerado en el siguiente <i>sprint</i> 	<ul style="list-style-type: none"> • El trabajo hecho durante el <i>sprint</i> se presenta y revisa • Se recoge <i>feedback</i> a ser considerado en el siguiente <i>sprint</i> • La retrospectiva del <i>sprint</i> se realiza inmediatamente después
<i>Sprint Retrospective</i>	<ul style="list-style-type: none"> • El <i>sprint</i> completado se revisa para ver si hay cambios o mejoras al proceso 	No hay reunión separada de retrospectiva

Tabla 3: Proceso *Scrum* para un desarrollador

Esta adaptación incluye casi todos las actividades y artefactos de *Scrum* a la vez que adapta los roles al contexto de un sólo desarrollador. La Tabla 4

detalla que elementos de *Scrum* se han incluido, usando la clasificación de Scrum Alliance [61].

Roles	
<i>Product Owner</i>	Sí
<i>Scrum Master</i>	No
Equipo de desarrollo	Sí
Artefactos	
Incremento de producto	Sí
<i>Product Backlog</i>	Sí
<i>Sprint Backlog</i>	Sí
Actividades	
<i>Product Backlog grooming</i>	Sí
<i>Sprint planning</i>	Sí
<i>Daily Scrum</i>	No
<i>Sprint retrospective</i>	No
<i>Sprint review</i>	Sí

Tabla 4: Elementos de Scrum incluidos

El caso se desarrolló en 3 *Sprints* de 10 días cada uno. El Anexo E: *Sprint Backlog* contiene la evolución del *product backlog* a lo largo de las iteraciones y el contiene el *burndown chart*, durante la discusión del caso se presentará únicamente el *Sprint Backlog* correspondiente a cada iteración.

De acuerdo con [62], un *Backlog* típico consiste de los siguientes tipos de tareas:

- Funcionalidades
- *Bugs*
- Trabajo Técnico
- Adquisición de conocimiento

A esta clasificación se le ha adicionado el rubro de usabilidad y es la que será usada a lo largo del caso.

4 Requisitos y concepción del diseño

De acuerdo a la bibliografía revisada, se utilizó un *Sprint 0* en el cuál se incluyeron dos grupos de tareas:

- La codificación de la funcionalidad que no estaba asociada a la interfaz de usuario de la aplicación.
- La investigación de usuario para poder identificar y definir los requisitos de la aplicación.

La Tabla 5 presenta el *sprint Backlog* para el *Sprint 0*

Tarea	Tipo
1 Prototipo de la aplicación en papel	Funcionalidades
2 Evaluación heurística de aplicaciones existentes para detectar problemas principales	Usabilidad
3 Investigar comentarios de otras aplicaciones	Adquisición de conocimiento
4 Realizar una prueba del primer prototipo con usuarios	Usabilidad
5 Codificar medición de decibeles básica en la aplicación	Trabajo técnico

Tabla 5 *Sprint Backlog* del *Sprint 0*

A continuación se presentará los resultados de cada uno de los puntos de la investigación de usuario.

4.1 Mapa de la funcionalidad de las aplicaciones existentes

Dado el número gran número de aplicaciones disponibles en Google Play, se consideraron los siguientes criterios para las aplicaciones a incluir en la búsqueda.

- La aplicación debía haber sido actualizada en el último año (2014)
- La aplicación debía contar con más de 500 mil descargas

Los términos de búsqueda utilizados fueron: Decibel, Sound Meter, Noise Meter. Se encontraron 3 aplicaciones que cumplían con los requisitos antes indicados y se produjo un mapa de la funcionalidad de ellas. Adicionalmente,

se incluyó en el análisis NoiseWatch que es una aplicación desarrollada por la Comunidad Europea como parte de una iniciativa para reducir el ruido.

Función \ Aplicación	Sound Meter	Noise Watch	Noise Meter	Decibel
Número de descargas	Más de 10 MM	Más de 10 mil	Más de 500 mil	Más de 500 mil
Calificación de los usuarios	4.2	3.9	4.2	3.6
Medir decibeles dB(A)	Sí	Sí	Sí	Sí
Tabla de referencia de ruidos comunes	Sí	No	Sí	No
Línea de tiempo	Sí	No	Sí	No
Calibración	Sí	No	Sí	Sí
Captura de pantalla	Sí	No	No	Sí
Máximo, mínimo y promedio	Sí	No	Sí	Sí
Manual	Sí	No	Sí	Sí
Geolocalización	No	Sí	No	No
Reportar mediciones	No	Sí	No	No
Captura de Logs	Sí	No	Sí	No

Tabla 6 Mapa de funcionalidades de las aplicaciones existentes

De las cuatro aplicaciones presentadas se realizó una evaluación heurística de la primera con el objetivo inicial de identificar posibles requisitos de la aplicación a desarrollar. Se escogió la primera dado que tenía la calificación más alta así como el mayor número de descargas.

4.2 Evaluación heurística

La evaluación se realizó utilizando los principios de usabilidad de Nielsen como base y el procedimiento de evaluación fue el siguiente:

- Cada uno de los evaluadores identificó de manera individual los problemas de la aplicación
- Los problemas fueron consolidados para tener un listado único de problema junto con el principio de usabilidad que incumplían.

Se empleó una evaluación heurística dado que esta es realizada por expertos y permite tener una retroalimentación rápida y de bajo costo al inicio del desarrollo [63]. El objetivo de esta evaluación fue doble:

- Identificar posibles funcionalidades faltantes de la aplicación que pudieran incrementar el *product backlog* y ser validadas posteriormente a través de un prototipo con los usuarios.
- Relevar posibles quiebres de las aplicaciones actuales que sirvieran como guía para evitar problemas de usabilidad en el desarrollo.

Dado que el objetivo de la evaluación no era solucionar los problemas de la aplicación evaluada y que el número de problemas identificados fue relativamente pequeño, no se realizó una priorización o calificación de los problemas identificados. Sin embargo, tal como se muestra en el Anexo H: *Usability Backlog*, se hizo seguimiento de los problemas identificados para no repetirlos en la aplicación o plantear soluciones en base a los principios de usabilidad o patrones de la plataforma.

4.2.1 Perfil de los evaluadores

Los evaluadores que participaron en la evaluación cumplen las siguientes características:

- Eran Ingenieros de Sistemas o Informática con experiencia previa en evaluaciones heurísticas
- Contaban con un dispositivo móvil Android de uso habitual en el cual realizar la evaluación.

Se utilizaron cinco evaluadores que de acuerdo con [64], permite identificar el 80% de los problemas.

N° de evaluador	Profesión	Evaluaciones previas
1	Ingeniero Informático	2
2	Ingeniero de Sistemas	2
3	Ingeniero de Sistemas	2
4	Ingeniero de Sistemas	2
5	Ingeniera de Sistemas	2

Tabla 7: Expertos que participaron en la evaluación heurística

4.2.2 Resultado

ID	Definición del problema	Comentarios/explicaciones sobre el problema	Ejemplos de ocurrencia del problema	Código	Principio(s) de usabilidad incumplido(s)
P1	El sistema muestra cuatro valores distintos de decibeles al usuario además de una línea del tiempo de los decibeles	La aplicación muestra múltiples datos de decibeles que podrían no ser relevantes para el usuario.	En la pantalla principal	H8	Diseño estético y minimalista
P2	La aplicación muestra un valor llamado Sound Pressure ratio que podría no ser familiar al usuario	El valor Sound Pressure ratio corresponde a un valor interno de la plataforma que podría no ser fácilmente entendido por los usuarios	En la pantalla de calibración	H2	Coincidencia entre sistema y el mundo real

ID	Definición del problema	Comentarios/explicaciones sobre el problema	Ejemplos de ocurrencia del problema	Código	Principio(s) de usabilidad incumplido(s)
P3	No es posible acceder a los botones del menú	En la pantalla del menú los botones de captura de pantalla y menú no son accesibles a pesar de ser visibles en este punto. Es necesario presionar el botón cancelar u Ok para poder volver a acceder a los botones	En la pantalla de calibración	H7	Flexibilidad y eficacia del uso
P4	El ajuste a los valores de dB debe ingresarse en incrementos de una unidad	La pantalla de calibración no permite ingresar rápidamente el valor de ajuste sino que es necesario incrementar o reducir el valor actual en 1 unidad	En la pantalla de calibración	H7	Flexibilidad y eficacia del uso
P5	Mensaje de error temporal al iniciar la aplicación	Si no es posible abrir el micrófono del dispositivo solo se muestra un mensaje temporal que indica que no se puede abrir el micrófono, luego de ello la aplicación permanece en blanco	En la pantalla principal al iniciar la aplicación	H9	Ayuda al usuario para reconocer, diagnosticar y recuperarse de errores
P6	Ruta del archivo de captura de pantalla	La aplicación muestra el nombre del archivo de captura de pantalla guardado sin embargo no muestra o indica donde lo ha guardado	Al capturar un pantalla	H1	Visibilidad del sistema
P7	El <i>action bar</i> no rota con la aplicación en el modo " <i>upside-down</i> "	Al rotar la aplicación, el encabezado de la aplicación, incluso la publicidad permanece en la misma orientación	Al habilitar el modo <i>upside down</i>	H4	Consistencia y estándares

ID	Definición del problema	Comentarios/explicaciones sobre el problema	Ejemplos de ocurrencia del problema	Código	Principio(s) de usabilidad incumplido(s)
P8	No debería haber un botón menú	Por convenciones de la plataforma, las acciones del menú (calibrar, manual, configuración, acerca de) debería estar en un <i>action overflow</i> representar por 3 puntos en lugar de bajo un botón menú	El menú	H4	Consistencia y estándares
P9	El botón manual debería llamarse ayuda	Por convenciones de plataforma, la opción manual debería llamarse ayuda y encontrarse al final de la lista	El manual	H4	Consistencia y estándares
P10	El manual solo incluye la función principal	El manual explica solo las funciones básicas de la aplicación, no incluye: 1. Como usar el botón T. 2. Como calibrar la aplicación	El manual	H10	Ayuda y documentación
P11	No se pide confirmación al usuario antes de restablecer valores.	Cuando se mantiene presionado el centro de la rueda, los valores se restablecen sin solicitar confirmación de la operación. Puede ser un problema en el caso de que se presione la pantalla inadvertidamente.	Mantener pulsado el centro de la rueda. Las lecturas se restablecen sin pedir confirmación al usuario.	H3	Control y libertad del usuario
P12	No hay confirmación de haberse realizado el proceso de restablecer	Cuando se mantiene presionado el centro de la rueda, los valores se restablecen y no muestran ningún mensaje que indique que se ha efectuado el proceso. Puede ser un	Mantener pulsado el centro de la rueda. Luego de restablecerse las lecturas no	H1	Visibilidad del sistema

ID	Definición del problema	Comentarios/explicaciones sobre el problema	Ejemplos de ocurrencia del problema	Código	Principio(s) de usabilidad incumplido(s)
	valores.	problema en el caso de que se presione la pantalla inadvertidamente.	se muestra ningún mensaje de haberse realizado la operación.		
P13	El icono disponible para cambiar la presentación de los datos no es muy explicativa	La metáfora usada para cambiar la presentación de datos no se asocia a esta funcionalidad. Se asocia más a cambiar Tipo de Letra.	Icono para cambiar la presentación de datos	H2	Coincidencia entre sistema y el mundo real
P14	La característica de calibración no es intuitiva	La opción Calibrar no indica valores referenciales realizar la calibración.	Presionar la opción Menú en la pantalla principal y seleccionar la opción Calibrar	H10	Ayuda y documentación

Tabla 8 Problemas identificados por evaluación heurística

Del listado de problemas identificados, se lograron identificar recomendaciones de cómo implementar las funcionalidades incluidas en el *product backlog*, si bien no se identificaron funciones adicionales que debía incluir la aplicación. Estas recomendaciones se plantearan como criterios de aceptación de las historias de usuario incluidas en el *product backlog*.

Uno de los puntos claves identificados fue la dificultad de entender y realizar la calibración de la aplicación, esto también se identificó en los comentarios de la aplicación en los cuales la percepción pasaba de ser muy buena para el caso de dispositivos en los cuales la aplicación funcionaba correctamente a muy mala para casos donde la calibración de la aplicación no era correcta.

4.3 Prototipado en papel

En función a la evaluación heurística y a la matriz de funcionalidades de las aplicaciones, se preparó una serie de historias de usuario tentativas a ser validadas con un conjunto potencial de usuarios.

Las historias iniciales fueron:

1. Como usuario casual debería poder medir la cantidad de decibeles que generan los ruidos ambientales
2. Como usuario casual tengo una referencia conocida (ejemplo) con la cual comparar el ruido que se está produciendo
3. Como usuario casual tengo claro las limitaciones del dispositivo móvil que estoy usando para medir decibeles
4. Como usuario casual puedo compartir los resultados de mi medición en redes sociales con imágenes y/o videos contextuales
5. Como usuario casual debería poder guardar mis mediciones para enviarlas a las autoridades lo que permitirá reportar ruidos molestos en mi comunidad

En función a estas historias se construyó un prototipo en papel (ver Anexo A) y se realizó una prueba con 4 usuarios para validar las historias de manera cualitativa. El script utilizado se encuentra en el

Anexo B.

Tal como se plantea, [52], se incluyó en los siguientes *sprints* más usuarios lo cual permitirá dar mayor validez a los resultados obtenidos, si bien en una fase inicial solo se incluyeron 4 usuarios para tener *feedback* rápido sobre el prototipo.

A continuación se presenta el objetivo y los resultados de la prueba.

4.3.1 Objetivos de la prueba

Los objetivos de la prueba fueron

- Identificar funcionalidades no levantadas hasta el momento
- Validar la intención de uso y la necesidad de una aplicación para medir el ruido en el contexto de vida de los usuarios.

4.3.2 Resultados

Con la finalidad de proteger la identidad de los usuarios, nos referiremos a ellos por las iniciales U1, U2, U3 y U4. La prueba se desarrolló en 3 fases, siguiendo la estructura planteada por [3]:

1. Una entrevista previa
2. El desarrollo de 2 tareas utilizando el prototipo en papel
3. Un cuestionario post-test.

Sobre la entrevista previa

- Con excepción del usuario U4, todos los usuarios indicaron que en su distrito existían problemas de contaminación sonora.
- Con excepción del usuario U4, ninguno de los usuarios tenía experiencia usando un sonómetro físico o una aplicación para medir decibeles.
- Los usuarios U3 y U4 indicaron haber reportado abusos en redes sociales antes, mientras que el usuario U1 indicó su intención de hacerlo en el caso la circunstancia lo ameritara.

Sobre las tareas:

- El usuario U1 identificó que la llamada de denuncia podría no ser útil dependiendo del horario de atención

- El usuario U3 utilizó el botón denunciar para compartir a Facebook dado que este era más llamativo, está no era la funcionalidad original del botón, sin embargo durante la prueba se alteró el prototipo para reflejar esta intención y se mantuvo de esa forma con los siguientes usuarios.

Sobre las pruebas post test:

- Todos los usuarios concordaron en que los escenarios eran realistas sin embargo tanto el U2 como el U4 indicaron que la herramienta podría tener más uso en un contexto industrial o empresarial.
- Los usuarios U2 y U3 indicaron la necesidad de guardar o poder enviar evidencias para sustentar una denuncia.
- El usuario U1 y U4 indicaron la importancia de mostrar los límites, tanto los normativos como aquellos relacionados a la salud de las personas.
- El usuario U3 fue el único que consultó si había variabilidad al momento de medir el número decibeles de un equipo a otro.

Se actualizaron las funcionalidades incluidas en el *product backlog* para reflejar la retroalimentación de los usuarios. Se incluyeron criterios de aceptación para las historias existentes.

A continuación se presenta las funcionalidades actualizadas en el *backlog*.

Backlog actualizado

1. Como usuario casual debería poder medir la cantidad de decibeles que generan los ruidos ambientales.
2. Como usuario casual tengo una referencia conocida (ejemplo) con la cual comparar el ruido que se está produciendo

Criterios de aceptación:

- Es posible visualizar la referencia como parte de una escala para poder poner en contexto la referencia y entenderla mejor
- Debo conocer el nivel de ruido permitido de acuerdo a la zonificación del lugar en el que me encuentro.

3. Como usuario casual puedo compartir los resultados de mi medición en redes sociales con imágenes y/o videos contextuales

Criterio de aceptación:

- Esta funcionalidad no debe confundir con la opción de denunciar o deben ser una única función integrada.

4. Como usuario casual debería poder guardar mis mediciones para enviarlas a las autoridades lo que permitirá reportar ruidos molestos en mi comunidad

Criterio de aceptación:

- La medición debe incluir algún tipo de evidencia que permita justificar mi denuncia o reporte.

5. Como usuario casual puedo reportar problemas de ruido a la autoridad

Criterio de aceptación:

- La aplicación deberá validar el horario de atención de cada autoridad.

6. Como usuario debería serme fácil calibrar la aplicación o no tener que calibrarla.

Criterio de aceptación:

Debe ser automático dado que el usuario no espera realizar esta tarea.

Esto concluyó en el desarrollo de un primer prototipo de la aplicación.

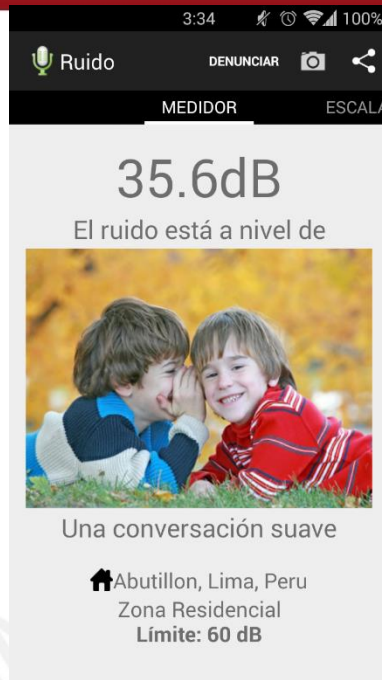


Ilustración 3: Prototipo inicial de la aplicación

5 Validación de patrones de diseño

En este capítulo se presenta los resultados del *Sprint 1* y la prueba realizada al final de este *sprint*.

Para esta se realizó una prueba de usabilidad con una primera versión de la aplicación.

En esta prueba participaron 4 usuarios: se incluyó dos usuarios que participaron en la prueba anterior para verificar si se habían resuelto los problemas previamente identificados y se incluyó dos usuarios nuevos que podrían identificar aspectos no considerados en la evaluación anterior.

Objetivos de la prueba

- Validar si los patrones de diseño de la plataforma evitaban o generaban algún problema de usabilidad.
- Explorar los posibles problemas de usabilidad de los elementos de la interfaz
- Identificar funcionalidad faltante a incluir en una siguiente iteración
- Compara la aplicación realizando un A/B con la aplicación (Sound Meter) que se utilizó en la evaluación heurística.

5.1 Lista de patrones de Android y su posible uso

Se preparó un listado basado en los patrones de Android presentados en [33]

Android Design Patterns	Aplica	Donde	Prueba
Action Bar	Sí	Botón para tomar fotos y grabar Botón para denunciar	Prueba 1
Sharing	Sí	Compartir en redes sociales	Prueba 1
Navigation			
Spinners	No	No, se utiliza <i>navigation drawer</i>	
Navigation Drawer	Sí	Para mostrar los paneles de la grabación	Fuera de alcance de la iteración
Details	No	No hay una relación master y <i>detail</i>	

Android Design Patterns	Aplica	Donde	Prueba
Tabs	No	No porque se están utilizando <i>Navigation drawer</i> y al recomendación es no mezclar	
Full Screen	No	No es una aplicación inmersiva	
Confirmation and Acknowledging			
Confirm	Sí	Los usuarios no deberían llamar sin conocer el impacto	
Acknowledge	Sí	Luego de tomar una foto o una imagen	Prueba 1
Notifications	No	No ninguna historia de usuario se cubre con notificación	
Swipe	Sí	El tipo de visualización cambia con el <i>swipe</i>	Prueba 1 / 2
Widgets	No	No, no cubre una historia de usuario.	
Settings	No	Se ha optado por no tener una configuración.	

Tabla 9 Lista de patrones Android

El contenido de la prueba se encuentra en Anexo C

5.2 Resultados de la prueba

Al igual que en la prueba anterior se utilizan las denominaciones U1 a U4 para representar a los usuarios, están son distintas a las utilizadas en la prueba anterior.

Patrones de Android probados:

- *Swipe*: Ninguno de los usuarios tuvo problemas para encontrar la escala utilizando *Swipe*.
- *Action Bar*: El usuario U3 no identificó el botón denunciar, esto resalta la importancia de utilizar un icono en el *action bar* a diferencia del texto que usó. El mismo usuario también tuvo problemas para diferenciar la cámara dado que esta se confundía con las pestañas de medición y escala.

- *Acknowledge*: Los usuarios U1 y U2 dudaron antes de realizar la llamada dado que creían que realmente estaban llamando a la municipalidad. Indicaron también que quizás no atendiera a todas horas. Este patrón permitió evitar que la aplicación tomara una decisión de la cual los usuarios no estaban tan seguros.
- *Sharing*: Los usuarios U2 y U4 mostraron frustración al usar el botón compartir dado que era lento al mostrar la lista de aplicaciones y al reaccionar después. Incluso el usuario U2 intentó presionar el botón múltiples veces. La implementación de esta función se realizó según la guía de desarrollo, sin embargo aquí podría colocarse alguna retroalimentación no estándar para indicar la espera.

Aplicación

- Ninguno de los usuarios notó la zonificación.
- Ninguno de los usuarios identificó la escala en la aplicación Sound Meter.
- Los usuarios U2 y U3, no utilizaron la funcionalidad de cámara en primera instancia por lo cual es importante que la aplicación comparta contenido significativo con el botón share por defecto.

En función a los resultados de esta prueba se ajustó las funcionalidades a desarrollar *backlog* para el *Sprint 2*. A continuación se muestran aquellas funcionalidades que tuvieron cambios.

1. Como usuario casual puedo reportar problemas de ruido a la autoridad

Criterio de aceptación:

- La aplicación deberá validar el horario de atención de cada autoridad.
 - La aplicación debería darme diversas opciones para reportar los problemas.
2. Como usuario casual puedo compartir los resultados de mi medición en redes sociales con imágenes y/o videos contextuales

Criterio de aceptación:

- Esta funcionalidad no debe confundir con la opción de denunciar o deben ser una única función integrada.
- La aplicación debe mostrar si está realizando determinado procesamiento al momento de compartir contenido.



6 Validación final del producto

Como parte del *Sprint* final de validación, se realizó una encuesta para validar algunos de los problemas que se habían identificado. También se validaron algunos supuestos que se habían considerado, por ejemplo el hecho que los usuarios reportarían sus problemas a la municipalidad. El contenido de la encuesta se encuentra en el Anexo D: Encuesta.

La encuesta se realizó a 60 personas lo cual da un margen de error de 10.6% con un nivel de confianza de 90%. El nivel de confianza de 90% se encuentra dentro de los estándares comúnmente usados por los investigadores [65] y se consideró un margen de error alto dado que el carácter de la encuesta es exploratorio.

Los resultados más relevantes de la encuesta fueron los siguientes:

- 70% considera que existen problemas de ruido en su distrito.
- 78% asocia dichos problemas al tráfico
- 60% considera que podría reducir dichos problemas reportándolos.
- Solo 27% ha hecho una denuncia antes
- El 77% de los participantes acudiría a Serenazgo antes que a la municipalidad o a la policía.
- 52% de los participantes comparte denuncias o realiza denuncias de abusos en redes sociales.
- 80% comparte dicho contenido en Facebook.

En función a estos resultados se agregaron los siguientes criterios de aceptación a los ítems del *Product Backlog*.

1. Como usuario casual puedo reportar problemas de ruido a la autoridad

Criterio de aceptación:

- La aplicación deberá mostrar prominentemente Serenazgo como una opción de denuncia.
2. Como usuario casual puedo compartir los resultados de mi medición en redes sociales con imágenes y/o videos contextuales

Criterio de aceptación:

- La aplicación deberá mostrar prominentemente Facebook como una opción para compartir contenido.

Posteriormente, se terminó con la construcción de la aplicación y se diseñó la prueba final del software. A continuación se presentan los objetivos y resultados de la prueba final.

Los puntos de comparación (*benchmarks*) de la prueba se han planteado en función a los resultados de las pruebas anteriores y de la encuesta.

6.1 Objetivo de la prueba

- Verificar que el 60% o más (datos según la encuesta) de los participantes consideran que la aplicación les serviría para reportar el problema de ruido a las autoridades.
- Verificar que el 90% (ningún participante tuvo problemas grandes durante las pruebas iniciales por eso el número) de los participantes pudieron completar los escenarios sin mayor inconveniente (Cumplen con los criterios de culminación de las tareas)
- Verificar los problemas identificados en iteraciones pasadas se encuentran corregidos.
- Identificar posibles problemas de usabilidad no detectados en pruebas anteriores.

6.2 Preguntas de investigación

- ¿Qué tan fácilmente, puede los usuarios realizar una grabación y compartirla en redes sociales?
- ¿Los participantes tienen alguna dificultad al diferenciar o identificar alguna funcionalidad dentro de la interfaz?
- ¿Los usuarios encuentran la escala referencial y la utilizan para entender la interfaz principal?
- ¿Identifican correctamente los usuarios la zonificación y el límite máximo de decibeles de la zona?
- ¿Los participantes esperan correctamente a que se habrá el menú de compartir?
- ¿Qué dudas tienen los usuarios al utilizar la aplicación?

- ¿Consideran los usuarios que el contenido de la grabación serviría de evidencia para realizar una denuncia?
- ¿Los usuarios identifican correctamente la aplicación en la pantalla de inicio de Android?

6.3 Características de los participantes

Para la prueba, se busca tener una distribución igual de ambos géneros. La prueba se realizará con 12 participantes en total, 2 de los cuales serán participantes de respaldo. Para el diseño se está siguiendo la guía en [3]

Tal como se indicó en el estado del arte, volver a probar con los mismos usuarios podría causar sesgos por efectos del aprendizaje por lo cual todos los usuarios para la fase final serán nuevos.

Características	Numero deseado de participantes
Tipo	
Regular	10
Respaldo	2
Edad	
20-35	Todos
Experiencia con instrumentos para medir el ruido	
Ninguna	10
Con experiencia previa	2
Género	
Hombres	6
Mujeres	6

Tabla 10: Características de los participantes

6.4 Metodología

Durante la prueba cada participante realizará 3 tareas con la aplicación móvil de manera aleatorizada para mitigar el aprendizaje de realizar una tarea antes de otra.

6.5 Estructura de la prueba

Introducción (10 minutos)

- Se tomará 5 minutos para presentar al participante información relacionada al ruido e indicar la mecánica de la prueba, se utilizará la introducción a la prueba presentada en el Anexo B.
- En este punto se le presentará el consentimiento informado.
- Se entregará un cuestionario para clasificar al participante.
- Introducción (10 minutos)

Entrevista previa (5 minutos)

- En este punto se realizará la prueba pre-test presentada en el Anexo B.

Tareas (20 minutos)

- Los participantes realizarán las tareas presentadas en el Anexo G: Script de prueba final. Todos los participantes empezarán desde la pantalla de inicio de Android, sin haber ingresado a la aplicación.

Debriefing (10 minutos)

- Se realizará un cuestionario de preguntas abiertas para levantar retroalimentación acerca de la aplicación (Ver Anexo G: Script de prueba final)

Medición

Desempeño

- Durante las pruebas se completará un *checklist* por cada usuario de los criterios de aceptación de cada una de las tareas.

Preferencia

- Para cada participante se recolectarán los resultados de la entrevista post-test.

6.6 Resultados de la prueba

Todos los usuarios con excepción de U1, lograron identificar correctamente los elementos de la interfaz y concluir con las tareas. Los resultados respecto a los objetivos planteados inicialmente fueron:

- El 70% de los participantes considera que la aplicación podría utilizarse de esta forma. Este resultado se encuentra dentro del margen de error de la encuesta.
- El 90% de los usuarios concluyó con éxito las tareas. Solo un usuario (U1) tuvo problemas en ejecutar las tareas planteadas.
- Durante la prueba se identificaron nuevos problemas de usabilidad, del mismo modo algunos problemas identificados iteraciones pasadas persistieron pero con menor intensidad, tal como se muestra en la Tabla 11

#	Problema de usabilidad	<i>Sprint 0</i>	<i>Sprint 1</i>	<i>Sprint 2</i>	Estado final
1	La aplicación debería tener una escala de referencia	Nueva funcionalidad	No presente	Interfaz	2 usuarios (u10 y u5) esperaban más contenido en esta opción
2	Debería tener zonificación. La zonificación debería ser correctamente visible	Nueva funcionalidad	Interfaz	Interfaz	2 usuarios (u4-u5) no identificaron la zonificación
3	La aplicación debería guardar evidencia de denuncia no solo llamar.	Nueva funcionalidad	Nueva funcionalidad	Interfaz	3 usuarios (u5-u7) indicaron que faltaba información mientras que u4 no la visualizó

#	Problema de usabilidad	Sprint 0	Sprint 1	Sprint 2	Estado final
4	La aplicación no valida el horario de atención de cada autoridad	Nueva funcionalidad	Nueva funcionalidad	No presente	Ok, cubierto por las otras opciones y porque se cambió a serenazgo que atiende 24 horas
5	La funcionalidad de denuncia y compartir no debería confundirse	Interfaz	No presente	No presente	Ok, todos los usuarios que completaron la tarea, con excepción de u1
6	La aplicación debería dar varias opciones para reportar problemas	No presente	Nueva funcionalidad	No presente	Ok, corregido, los usuarios consideraron los medios como suficiente
7	La aplicación debe mostrar si está realizando determinado procesamiento al compartir	No presente	Interfaz	No presente	Ok, ningún usuario presiono el botón compartir más de una vez
8	Problemas para compartir	No presente	No presente	Interfaz	6 usuarios (u2, u4 y u7-u10) tuvieron problemas con el click largo
9	Los iconos de la aplicación no se identifican correctamente	No presente	No presente	Interfaz	2 usuarios (u9 y u10) tuvieron problemas con la iconografía
#	La aplicación debería armar el post en lugar de solo redirigir a la página.	No presente	No presente	Nueva funcionalidad	2 usuarios esperaban que la funcionalidad armara un post en redes sociales (u5 y u9)
#	Movimiento de la escala	No presente	No presente	Interfaz	1 usuario encontró incómodo el movimiento de la escala (u5)

Tabla 11: Evolución de problemas de usabilidad

7 Discusión de resultados

El flujo *Scrum* planteado que se obtuvo como ejecución del caso de estudio se presenta en la Ilustración 4.

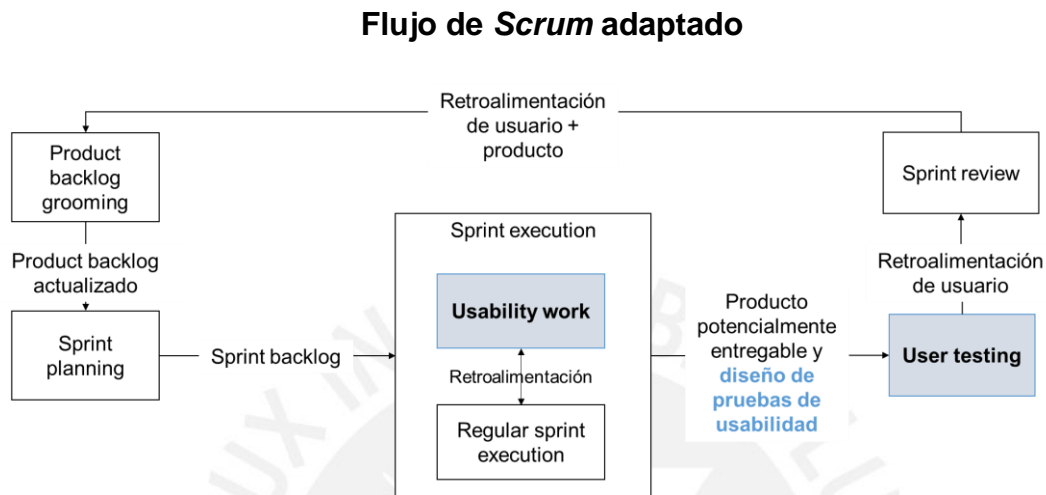


Ilustración 4: Flujo *Scrum* adaptado. Fuente: Elaboración propia

A diferencia del flujo de *Scrum* tradicional, se ha incluido como parte de la ejecución del *sprint* tareas de usabilidad, las cuales dan retroalimentación a la ejecución regular del *sprint*. En este punto, es importante hacer las siguientes distinciones

- Durante el *sprint* 0, no existió retroalimentación entre las tareas de usabilidad y las tareas de construcción regulares dado que éstas últimas correspondían al código *backend*, el cual no se ve afectado por la interfaz de usuario.
- Las pruebas de usabilidad se han colocado como un proceso separado posterior dado que ejecutarlas implica congelar la funcionalidad del sistema durante la ejecución de las mismas [3]. Durante el caso de estudio, estas formaron parte del *sprint backlog* y se colocaron también en el *burndown chart*, sin embargo, siempre se ejecutaron al final de la iteración, lo cual indica que deben colocarse como un proceso separado.

Las técnicas que sí se lograron integrar como parte de los *sprints* fueron: el desarrollo de prototipos, las evaluaciones heurísticas, la indagación individual

(encuestas). Estas permitieron reforzar la especificación de las historias de usuario o tienen poca dependencia de un estado particular de la aplicación que se está desarrollando y permiten incluirse como retroalimentación para el siguiente *product backlog grooming*.

El *product backlog grooming* se realiza tanto con retroalimentación directa del usuario final y de la inspección del producto desarrollado. Por ejemplo, esto se puede apreciar en los dos siguientes ítems. El ítem 9 es resultado de la inspección del producto que, post-*sprint* 0, presentaba un defecto de funcionamiento, mientras que al punto 10 se le agregaron criterios de aceptación para reflejar las necesidades del usuario identificadas en las pruebas con el prototipo.

Ítem	Descripción	Tipo
8	(Ítem agregado Post-Sprint 0) Corrección del bug que evita el cerrado del micrófono	Bugs
9	Como usuario casual tengo una referencia conocida (ejemplo) con la cual comparar el ruido que se está produciendo Criterio de aceptación (Post-Sprint 0): Es posible visualizar la referencia como parte de una escala para poder poner en contexto la referencia y entenderla mejor (Post-Sprint 0): Debo conocer el nivel de ruido permitido de acuerdo a la zonificación del lugar en el que me encuentro.	Funcionalidades

Tabla 12: Elementos del *product backlog* después del *product backlog grooming*

Tal como se presenta en la Ilustración 5, el número de problemas críticos que requirieron para su solución codificación de funcionalidades completamente nuevas se redujo con cada iteración, lo cual se logró al tener retroalimentación en cada una. Los problemas pasaron, por ejemplo, de funcionalidades totalmente nuevas, como la creación de una forma de grabar video que incluya evidencia del nivel de ruido en el *sprint* 0, a problemas con un menor impacto que se detectaron en el *sprint* 2 como problemas en la iconografía. Estos

últimos si bien tienen un impacto en la usabilidad no impedían significativamente el uso de la aplicación.

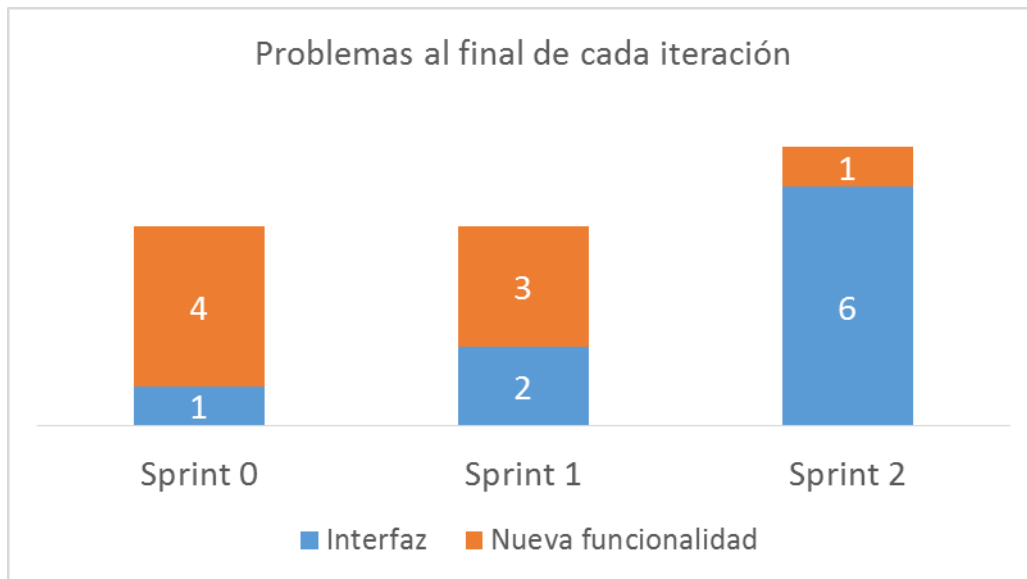


Ilustración 5: Problemas al final de cada iteración

El crecimiento del número de problemas se explica por 3 factores:

- Las correcciones aplicadas no corrigieron los problemas identificados en las iteraciones anteriores.
- Las correcciones que introdujeron nueva funcionalidad, introdujeron a su vez problemas de usabilidad asociados a esta funcionalidad.
- El número de usuarios que participaron en la última prueba de usabilidad fue mayor y permitió que se identificaran problemas no explorados en iteraciones anteriores.

Esto no desmerece el hecho de integrar prácticas de usabilidad descontada tal como plantean [66] [52] [3], sin embargo si muestran las limitaciones de realizar pruebas con pocos usuarios tal como indica [67] y haría necesario que cada cierto número de iteraciones se realicen pruebas con un mayor número de usuarios.

Un aspecto importante a tomar en cuenta son los problemas no resueltos de un *sprint* a otro. Para este punto, es importante incluir alguna técnica y tareas respectivas en el *sprint* que permitan obtener retroalimentación directa de los usuarios antes de la siguiente ronda de pruebas de usabilidad sobre los problemas en corrección. Podrían explorarse técnicas tales como el prototipado

rápido o pruebas de usabilidad reducidas con 1 o 2 usuarios que sí puedan integrarse en la mitad de un sprint.

Por otra parte, tal como se observa en el *sprint backlog*, parte del tiempo del *sprint* se utilizó para diseñar las pruebas de usabilidad y las funcionalidades incluidas en las pruebas fueron creciendo en cada iteración. Sería importante poder construir las pruebas de incrementalmente de tal manera que se utilicen de manera similar a pruebas unitarias pero integrando a los usuarios. Esto podría reducir el tiempo de diseño de las pruebas y aseguraría que se están volviendo a inspeccionar los quiebres encontrados en iteraciones anteriores.

Una última observación sobre el proceso ejecutado es que existe trabajo de usabilidad como la realización de encuestas que si bien es integrable como parte de un *sprint* y entrega retroalimentación durante el mismo, podría tener un mayor impacto y alcance si se realizará de manera paralela o distribuida a lo largo de múltiples *sprints*.

Finalmente, en lo que concierne a los patrones de la plataforma (Android), con la excepción del patrón de clic largo que se introdujo en la última iteración [68], los usuarios no tuvieron problemas para utilizar el resto de patrones de la plataforma utilizados en la aplicación. Tal como se presentó en el Capítulo 0, los patrones son identificados por sus usuarios en casi todo los casos y los problemas que presentan con ellos son de carácter cosmético.

8 Conclusiones y Trabajos futuros

En esta sección, se presentan las conclusiones del caso de estudio realizado y los trabajos futuros que de este se desprenden.

8.1 Conclusiones

La ejecución de un *Sprint 0* permitió identificar y especificar las necesidades de los usuarios y sirvió de base para el trabajo en los siguiente *Sprints*. Tal como se presentó en el Capítulo 4, la evaluación heurística reforzó la idea de que era necesario analizar los patrones de la plataforma. Tanto esta evaluación como la ejecución de una prueba con un prototipo en papel permitieron especificar las historias de usuario a través de criterios de aceptación. En paralelo, haber construido parte de la funcionalidad que no se veía afectada por las pruebas permitió mantener el espíritu de ágil de tener un producto entregable siempre funcionando y reducir el tiempo de codificación de los siguientes *sprints*.

Los patrones de la plataforma ayudaron a crear una sensación de familiaridad en el usuario y a que identifiquen correctamente la funcionalidad de la aplicación. Durante la validación de patrones, se observó que casi la totalidad de los usuarios tanto en el *Sprint 1* como en la *Sprint 2* utilizaron correctamente los patrones. Tal como se presentó en esta validación, los problemas que sí ocurrieron con los patrones podrían deberse a la forma de reflejarlos en la interfaz de usuario o a una incorrecta elección del patrón, más que a un problema del patrón mismo.

En lo que respecta a las técnicas de usabilidad, estas son integrables en dos puntos del flujo *Scrum*: durante la ejecución de los *sprints* y antes del *sprint review* como un proceso de validación adicional.

Las técnicas que pueden integrarse como tareas dentro del *sprint backlog* y por ende dentro de la ejecución de un *sprint* son:

- Prototipos.
- Evaluación heurística.
- Indagación individual.
- El diseño de las pruebas de usabilidad.

En el caso de la indagación individual y evaluación heurística, esto se debe a que estas técnicas permiten refinar el *sprint backlog*. Las cuatro técnicas, comparten la característica de no depender de haber terminado el resto de tareas del *sprint*.

Por otro lado, las pruebas de usabilidad, que sirven como validación del proceso de construcción, deben incluirse al culminar cada iteración, dado que dependen de tener un producto potencialmente entregable y potencialmente evaluable lo cual no ocurre hasta el final de cada *sprint*. Esto es válido inclusive para el *Sprint 0*, donde el producto potencialmente entregable de cara al usuario es el prototipo en papel.

Finalmente, se identificó una oportunidad para construir de manera iterativa las pruebas de usabilidad. A medida que se avanza con la construcción de la aplicación, se incrementa el alcance de la pruebas, a la vez que es necesario validar y asegurar que aspectos evaluados en iteraciones anteriores sigan siendo correctos o hayan sido corregidos según sea el caso. Por este motivo, sería conveniente poder refinar e incrementar el alcance de una o varias pruebas de usabilidad que crezcan con la construcción de la aplicación, en lugar de construir pruebas desde cero en cada iteración. El producto potencialmente entregable que sale de la ejecución del *sprint* debería por ende incluir las pruebas de usuario asociada.

8.2 Trabajos futuros

En esta sección, se presenta aspectos pendientes de explorar que se identificaron como parte del desarrollo de la tesis y podrían ser motivo de futuras investigación

El primer punto identificado es la necesidad de realizar experimentos controlados para medir el impacto cuantitativo de la integración de las técnicas ágiles con usabilidad y validar los hallazgos del presente estudio.

Por otra parte, dado que el caso presenta un contexto particular de un único desarrollador y una aplicación móvil que tiene un alcance pequeño, sería importante realizar más casos de estudio donde se utilicen aplicaciones de mayor alcance y se apliquen a equipos de desarrollo.

Finalmente, un punto pendiente por explorar sería la forma de integrar técnicas de usabilidad con otros *frameworks* ágiles crecientes como Kanban.



9 Referencias bibliográficas

- [1] Agile Alliance, "Agile Manifesto," [Online]. Available: <http://agilemanifesto.org/history.html>. [Accessed 08 12 2014].
- [2] Jim Highsmith, "History: The Agile Manifesto," Agile Alliance, 2001. [Online]. Available: <http://agilemanifesto.org/history.html>. [Accessed 08 12 2014].
- [3] J. Rubin, D. Chisnell and J. Spool, Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests, Indianapolis: Wiley Publishing, Inc., 2008.
- [4] S. M. Butt, W. F. W. Ahmad and L. Rahim, "Handling tradeoffs between agile and usability methods," in *Computer and Information Sciences (ICCOINS), 2014 International Conference on*, 2014.
- [5] O. Sohaib and K. Khan, "Integrating usability engineering and agile software development: A literature review," in *Computer Design and Applications (ICCD), 2010 International Conference on*, 2010.
- [6] S. Khalaf, "Apps Solidify Leadership Six Years into the Mobile Revolution," Flurry Analytics, [Online]. Available: <http://www.flurry.com/bid/109749/Apps-Solidify-Leadership-Six-Years-into-the-Mobile-Revolution>. [Accessed 08 12 2014].
- [7] VersionOne, "8th Annual State of Agile Development Survey," VersionOne, 2014.
- [8] M. McLaughlin, "Agile Methodologies for Software Development," VersionOne, [Online]. Available: <http://www.versionone.com/agile-101/agile-development-methodologies-scrum-kanban-lean-xp/>. [Accessed 08 12 2014].
- [9] K. Rubin, Essential Scrum, Michigan: Pearson Education, 2013.

- [10] A. Oulasvirta, S. Tamminen, V. Roto and J. Kuorelahti, "Interaction in 4-Second Bursts: The Fragmented Nature of Attentional Resources in Mobile HCI," [Online]. Available: <https://research.nokia.com/files/InteractionIn4SecondBursts.pdf>. [Accessed 28 septiembre 2014].
- [11] B. Schildbach and E. Rukzio, "Investigating selection and reading performance on a mobile phone while walking," in *Proceedings of the 12th international conference on Human computer interaction with mobile devices and services*, 2010.
- [12] R. Harrison, D. Flood and D. Duce, "Usability of mobile applications: literature review and rationale for a new usability model," *Journal of Interaction Science*, 2013.
- [13] D. A. Norman and J. Nielsen, "Gestural interfaces: a step backward in usability," vol. 17, no. 5, 2010.
- [14] F. Nayebe, J.-M. Desharnais and A. Abran, "The State Of The Art Of Mobile Application Usability Evaluation".
- [15] D. Raptis, N. Tselios, J. Kjeldskov and M. B. Skov, "Does size matter?: investigating the impact of mobile phone screen size on users' perceived usability, effectiveness and efficiency.," in *Proceedings of the 15th international conference on Human-computer interaction with mobile devices and services*, 2013.
- [16] L. Schwartz, "Agile-User Experience Design: With or Without a Usability Expert in the Team?," 2013.
- [17] L. Schwartz, "Agile-user Experience Design: an Agile and User-Centered Process," in *ICSEA 2013: The Eighth International Conference on Software Engineering Advances*, 2013.
- [18] Compuware, "Mobile Apps: What consumers Really Need and Want," Compuware, 2012.

- [19] IDC, "IDC," [Online]. Available: <http://www.idc.com/prodserv/smartphone-os-market-share.jsp>. [Accessed 21 Septiembre 2014].
- [20] ISO, "ISO 9241-1," 1992.
- [21] J. Nielsen, Usability Engineering, 1993.
- [22] U.S. Department of Health & Human Services, "User-Centered Design Process Map," 06 marzo 2014. [Online]. Available: <http://www.usability.gov/how-to-and-tools/resources/ucd-map.html>. [Accessed 28 septiembre 2014].
- [23] U.S. Department of Health and Human Services, "User Research Basics," 06 marzo 2014. [Online]. [Accessed 28 septiembre 2014].
- [24] K. O'Connor, "Personas: The Foundation of a Great User Experience," 25 march 2011. [Online]. Available: <http://uxmag.com/articles/personas-the-foundation-of-a-great-user-experience>. [Accessed 28 septiembre 2014].
- [25] usability.gov, "Usability Testing," usability.gov, [Online]. Available: <http://www.usability.gov/how-to-and-tools/methods/usability-testing.html>. [Accessed 08 12 2014].
- [26] J. Nielsen, "10 Usability Heuristics for User Interface Design," 01 enero 1995. [Online]. Available: <http://www.nngroup.com/articles/ten-usability-heuristics/>. [Accessed 28 septiembre 2014].
- [27] J. Vijayan and G. Raju, "A New approach to Requirements Elicitation Using Paper Prototype," *International Journal of Advanced Science and Technology*, vol. 28, pp. 9-16, 2011.
- [28] C. Snyder, Paper Prototyping: The Fast and Easy Way to Design and Refine User Interfaces, Interactive Technologies, 2003.
- [29] U.S. Department of Health and Human Services, "Heuristic Evaluations and Expert Reviews," [Online]. Available: <http://www.usability.gov/how->

- to-and-tools/methods/heuristic-evaluation.html. [Accessed 28 septiembre 2014].
- [30] Agile Alliance, "What is Agile?," [Online]. Available: <http://www.agilealliance.org/the-alliance/what-is-agile/>. [Accessed 02 octubre 2014].
- [31] Gartner, "Gartner," 13 Febrero 2014. [Online]. Available: <http://www.gartner.com/newsroom/id/2665715>. [Accessed 14 Septiembre 2014].
- [32] Kantar World Panel, "Kantar World Panel," [Online]. Available: <http://www.kantarworldpanel.com/global/smartphone-os-market-share/>. [Accessed 14 September 2014].
- [33] Google, "Design | Android Developers," 2014. [Online]. Available: <https://developer.android.com/design/index.html>. [Accessed 26 septiembre 2014].
- [34] Apple Inc., "iOS Human Interface Guidelines," [Online]. Available: <https://developer.apple.com/library/ios/documentation/userexperience/conceptual/mobilehig/>. [Accessed 26 septiembre 2014].
- [35] D. Zhang and B. Adipat, "Challenges, methodologies, and issues in the usability of testing mobile applications," 2005.
- [36] R. Budiu, "Nielsen Norman Group," 13 abril 2014. [Online]. Available: <http://www.nngroup.com/articles/scaling-user-interfaces/>. [Accessed 28 septiembre 2014].
- [37] R. I. Singh, M. Sumeeth and J. Miller, "Evaluating the Readability of Privacy Policies in Mobile Environments," *International Journal of Mobile Human Computer Interaction*, 2011.
- [38] A. I. Wasserman, "Software engineering issues for mobile application development," in *Proceedings of the FSE/SDP workshop on Future of software engineering research*, 2010.

- [39] U.S. Department of Health and Human Services, "Mobile Usability Testing," 06 marzo 2014. [Online]. Available: <http://www.usability.gov/how-to-and-tools/methods/mobile-device-testing.html>. [Accessed 28 septiembre 2014].
- [40] T. Lang, "Eight Lessons in Mobile Usability Testing," 10 abril 2013. [Online]. Available: <http://uxmag.com/articles/eight-lessons-in-mobile-usability-testing>. [Accessed 29 septiembre 2014].
- [41] C. Cuadrat Seix, M. S. Veloso and J. J. R. Soler, "Towards the Validation of a Method for Quantitative Mobile Usability Testing Based on Desktop Eyetracking," 2012.
- [42] J. Kjeldskov and M. B. Skov, "Was it Worth the Hassle? Ten Years of Mobile HCI Research Discussions on Lab and Field Evaluations," 2014.
- [43] World Health Organization, "Guidelines for Community Noise," 1999. [Online]. Available: <http://www.who.int/docstore/peh/noise/guidelines2.html>.
- [44] B. Moore, An Introduction to the Psychology of Hearing, Boston: Leiden, 2013.
- [45] C. H. Hansen, Fundamental of Acoustics, World Health Organization.
- [46] N. Z. Soren Bech, Perceptual Audio Evaluacion - Theory, Method and Application, Chichester: John Wiley & Sons Ltd, 2006.
- [47] Presidencia del Consejo de Ministros de Perú, "Reglamento de Estándares Nacionales de Calidad Ambiental para el Ruido," Presidencia del Consejo de Ministros, Lima, 2003.
- [48] D. R. Lanman, "Design of a Sound Level Mete," 29 noviembre 2005. [Online]. Available: http://alumni.media.mit.edu/~dlanman/courses/decibel_meter.pdf. [Accessed 29 septiembre 2014].

- [49] "Frequency Weighting Spectrum," [Online]. Available: <http://www.cross-spectrum.com/audio/weighting.html>. [Accessed 09 September 2014].
- [50] C. Salvador, "Una revisión sistemática de usabilidad en metodologías ágiles," 2013.
- [51] A. Wale-Kolade, P. A. Nielsen and T. Päivärinta, "Usability Work in Agile Systems Development Practice: A Systematic Review," in *Building Sustainable Information Systems*, Springer, 2013, pp. 569-582.
- [52] J. Nielsen, "Discount Usability: 20 Years," Nielsen Norman Group, [Online]. Available: <http://www.nngroup.com/articles/discount-usability-20-years/>. [Accessed 18 octubre 2014].
- [53] A. Dahl, "Agile/UX Integration:: how user experience-related practices and processes are integrated with Agile development processes in real-world projects," 2012.
- [54] T. S. a. S. M. S. Da Silva, F. Maurer, T. Hellmann and others, "User experience design and agile development: From theory to practice," vol. 5, 2012.
- [55] O. Sohaib and K. Khan, "Integrating usability engineering and agile software development: A literature review," in *Computer design and applications (ICCD), 2010 international conference on*, 2010.
- [56] G. Jurca, T. D. Hellmann and F. Maurer, "Integrating Agile and User-Centered Design".
- [57] M. Singh, "U-SCRUM: An agile methodology for promoting usability," in *Agile, 2008. AGILE'08. Conference*, 2008.
- [58] J. Persson, "Communication through boundary objects in distributed agile teams: An integration of user-centered design and agile development," 2014.
- [59] M. Singh, "U-SCRUM: An Agile Methodology for Promoting Usability," in

- Agile, 2008. AGILE '08. Conference, 2008.*
- [60] A. Srirangarajan, "The Scrum Process for Independent Programmers," 2009.
- [61] Scrum Alliance, "Core Scrum," Scrum Alliance, [Online]. Available: <https://www.scrumalliance.org/why-scrum/core-scrum-values-roles>. [Accessed 9 12 2014].
- [62] M. Cohn, "Scrum Product Backlog," Mountain Goat Software, [Online]. Available: <http://www.mountaingoatsoftware.com/agile/scrum/product-backlog>. [Accessed 23 noviembre 2014].
- [63] Usability.gov, "Heuristic Evaluation," [Online]. Available: <http://www.usability.gov/how-to-and-tools/methods/heuristic-evaluation.html>. [Accessed 2014 12 09].
- [64] J. Nielsen, "Why You Only Need to Test with 5 Users," Nielsen Norman Group, 19 03 2000. [Online]. Available: <http://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/>. [Accessed 09 12 2014].
- [65] SurveyMonkey, "Sample Size Calculator," Survey Monkey, [Online]. Available: <https://www.surveymonkey.com/blog/en/sample-size-calculator/>. [Accessed 2014 12 09].
- [66] O. Sohaib and K. Khan, "Incorporating Discount Usability in Extreme Programming," *International Journal on Software Engineering and its Applications*, vol. 5, no. 1, pp. 51-61, 2011.
- [67] A. Woolrych and G. Cockton, "Why and when five test users aren't enough," in *Proceedings of IHM-HCI 2001 conference*, Toulouse, 2001.
- [68] Google, "Design | Action Bar," [Online]. Available: <http://developer.android.com/design/patterns/actionbar.html>. [Accessed 15 12 2014].