



FACULTAD DE INGENIERÍA

CARRERA DE INGENIERÍA DE SISTEMAS COMPUTACIONALES

“ANÁLISIS DE LA EFICIENCIA DE SOFTWARE
BAJO ISO/IEC 9126-2 EN UN APLICATIVO MÓVIL
LITE PARA OPERACIONES MINERAS”

Tesis para optar el título profesional de:

Ingeniero de Sistemas Computacionales

Autores:

Bach. Anderson Kevin Bazán Urteaga

Bach. Oscar Geny Saavedra Guarniz

Asesora:

Dra. Patricia Janet Uceda Martos

Cajamarca - Perú

2020

DEDICATORIA

A mis tres madres Berminda, Emelina y Silvia quienes siempre estuvieron a mi lado para guiarme, apoyarme y que gracias a ellas he podido sobresalir y cumplir todas las metas que me he propuesto.

A mis hermanos que son el plus que necesito para no caer en momentos difíciles y a toda mi familia que siempre están ahí con su alegría característica apoyándonos entre todos, esto es por ustedes Urteagas.

Anderson Kevin Bazán Urteaga

A mis padres Salomón e Imelda y por supuesto a todos mis hermanos que me han apoyado durante todos estos años.

Oscar Geny Saavedra Guarniz

AGRADECIMIENTO

A la Dra. Patricia Janet Uceda Martos por su amistad, compromiso y paciencia en todos estos años de formación profesional, quien siempre estuvo apoyándonos y alentándonos a seguir adelante.

Así también, un agradecimiento muy especial a toda mi familia por el apoyo brindado en todos estos años, gracias, son los máximo Urteagas.

Anderson Kevin Bazán Urteaga

Deseo expresar mi agradecimiento a la Dra. Patricia Uceda por la dedicación y apoyo brindado en este proyecto.

Gracias a mis padres y hermanos por estar siempre presentes y por el apoyo brindado, son un aliento para seguir.

Oscar Geny Saavedra Guarniz

TABLA DE CONTENIDOS

DEDICATORIA	2
AGRADECIMIENTO	3
TABLA DE CONTENIDOS	4
ÍNDICE DE FIGURAS	7
RESUMEN	13
CAPÍTULO I. INTRODUCCIÓN.....	14
1.1. Realidad problemática	14
1.2. Formulación del problema	23
1.3. Objetivos	23
1.3.1. Objetivo general	23
1.3.2. Objetivos específicos.....	23
1.4. Hipótesis	23
1.4.1. Hipótesis general	23
1.4.2. Hipótesis específicas	23
CAPÍTULO II. METODOLOGÍA	25
2.1. Tipo de investigación.....	25
2.2. Población y muestra (Materiales, instrumentos y métodos).....	25
2.2.1. Unidad de estudio.....	25
2.2.2. Población.....	25
2.2.3. Muestra.....	25
2.2.4. Materiales, instrumentos y métodos.....	26
2.3. Técnicas e instrumentos de recolección y análisis de datos	26
2.4. Procedimiento	27
2.4.1. Recolección de información.....	27
2.4.2. Análisis de información	28
CAPÍTULO III. RESULTADOS	30
3.1. Objetivo 1: Identificar las actividades de campo y las tecnologías móviles que utilizan en el área de SSOMA.....	30
3.2. Objetivo 2: Desarrollar e implementar un aplicativo móvil lite que cumpla con una de las actividades de campo del área de SSOMA.....	31
3.3. Objetivo 3: Evaluar la eficiencia de software del aplicativo móvil lite considerando los criterios de conformidad de la norma ISO/IEC 9126 luego de su implementación.....	33
3.3.1. Tiempo medio de respuesta.....	34

3.3.2.	Rendimiento de la interfaz de usuario	34
3.3.3.	Utilización máxima de memoria	36
3.3.4.	Conformidad de la eficiencia	38
CAPÍTULO IV. DISCUSIÓN Y CONCLUSIONES.....		39
4.1.	Discusión.....	39
4.2.	Conclusiones	41
REFERENCIAS		43
ANEXOS		46
Anexo N° 1. Ficha de campo para registrar actos o condiciones observadas.....		46
Anexo N° 2. Formatos en Excel para el registro de inspecciones		47
Anexo N° 3. Flujo del proceso para registrar actos o condiciones observadas		48
Anexo N° 4. Operacionalización de Variables de Estudio		55
Anexo N° 5. Encuesta para identificar el problema.....		63
Anexo N° 6. Firebase Performance Monitoring para recolectar y analizar información ..		73
Anexo N° 7. Metodología de desarrollo		82
Anexo N° 8. Manual de usuario.....		115
Anexo N° 9. Vincular aplicativo hacia Firebase Performance Monitoring		132
Anexo N° 10. Resultados de la cobertura de red según la encuesta		137
Anexo N° 11. Resultados de herramientas de tecnología móvil según la encuesta.....		138
Anexo N° 12. Resultados ayuda de herramientas móviles según la encuesta		140
Anexo N° 13. Desarrollo de la aplicación en Android Studio.....		141
Anexo N° 14. Configurar la Biblioteca de Persistencia de datos ROOM		157
Anexo N° 15. WEB API YANAPAY		169
Anexo N° 16. Base de datos Yanapay		176
Anexo N° 17. Servidor Amazon LightSail		180
Anexo N° 18. Resultados tiempo medio de respuesta		184
Anexo N° 19. Resultados rendimiento de la interfaz de usuario		189
Anexo N° 20. Resultados utilización máxima de memoria		192

ÍNDICE DE TABLAS

Tabla 1 Actividades de campo.....	30
Tabla 2 Resultados de las actividades de campo	30
Tabla 3 Criterios de Conformidad para Tiempo Medio de Respuesta	34
Tabla 4 Resultados del tiempo medio respuesta.....	34
Tabla 5 Resultados del rendimiento de la interfaz de usuario	35
Tabla 6 Criterios de conformidad para utilización de memoria	36
Tabla 7 Resultados de la utilización máxima de memoria	37
Tabla 8 Conformidad de la Eficiencia	38

ÍNDICE DE FIGURAS

Figura 1: Arquitectura del proyecto móvil	32
Figura 2: Ficha de campo	46
Figura 3: Formato Excel para reporte de inspecciones.....	47
Figura 4: Formato Excel para reporte de actos y condiciones.....	47
Figura 5: Diagrama del proceso para reportar actos y condiciones.....	50
Figura 6: Gráfico de barras	69
Figura 7: Gráfico circular	69
Figura 8: Cuadro de recolección de tiempos de duración	75
Figura 9: Cuadro de recolección de procesamiento lento.....	76
Figura 10: Cuadro de recolección de tiempos de respuesta.....	76
Figura 11: Gráfico de anillo.....	77
Figura 12: Gráfico de área de tiempo	78
Figura 13: Gráfico de área de porcentaje.....	78
Figura 14: Gráfico de área de memoria	78
Figura 15: Modelo del dominio	85
Figura 16: Diagrama de caso de uso supervisor	87
Figura 17: Diagrama de caso de uso personal observado.....	87
Figura 18: Prototipo ver observaciones	88
Figura 19: Prototipo ver observaciones pendientes	89
Figura 20: Prototipo ver observaciones cerradas.....	90
Figura 21: Prototipo buscar observaciones.....	91
Figura 22: Prototipo buscar observaciones pendientes.....	92
Figura 23: Prototipo buscar observaciones cerradas	93
Figura 24: Prototipo registrar observación	94
Figura 25: Prototipo adjuntar evidencia	95
Figura 26: Prototipo registrar solución.....	96
Figura 27: Prototipo visualizar detalle observación	97
Figura 28: Diagrama de paquetes	97

Figura 29: Diagrama de robustez ver observaciones.....	109
Figura 30: Diagrama de robustez ver observaciones pendientes.....	110
Figura 31: Diagrama de robustez ver observaciones cerradas	110
Figura 32: Diagrama de robustez buscar observaciones	110
Figura 33: Diagrama de robustez buscar observaciones pendientes	111
Figura 34: Diagrama de robustez buscar observaciones cerradas	111
Figura 35: Diagrama de robustez registrar observación.....	112
Figura 36: Diagrama de robustez adjuntar evidencia	112
Figura 37: Diagrama de robustez registrar solución.....	113
Figura 38: Diagrama de robustez visualizar detalle observación	113
Figura 39: Diagrama de clases.....	114
Figura 40: Vista presentación	117
Figura 41: Vista iniciar sesión	118
Figura 42: Vista pantalla de inicio.....	119
Figura 43: Vista opción observaciones.....	119
Figura 44: Vista opción agregar observación	120
Figura 45: Vista registrar observación.....	120
Figura 46: Vista opción guardar	121
Figura 47: Vista lista de observaciones	121
Figura 48: Vista seleccionar observación.....	122
Figura 49: Vista opción adjuntar evidencia.....	123
Figura 50: Vista captura de fotografía.....	124
Figura 51: Vista buscar archivo de evidencia.....	124
Figura 52: Vista estado de red	125
Figura 53: Vista opción ingresar	126
Figura 54: Mensaje modo offline	126
Figura 55: Vista opción de configuración	127
Figura 56: Vista configuración	127
Figura 57: Vista botón offline	128
Figura 58: Vista opción sincronizar.....	128

Figura 59: Vista opción reportar.....	130
Figura 60: Vista mensaje de correo	131
Figura 61: Página de inicio de Firebase.....	134
Figura 62: Añadir proyecto Yanapay en consola de Firebase	134
Figura 63: Dependencia para adicionar el SDK	135
Figura 64: Complemento para Performance Monitoring	135
Figura 65: Complemento y dependencia para Performance Monitoring	136
Figura 66: Proyecto Yanapay creado en Firebase	136
Figura 67: Herramienta Performance vinculada al aplicativo	136
Figura 68: Resultados de la cobertura de red dentro de la operación minera.....	137
Figura 69: Resultados de las zonas donde la cobertura de red es menor.....	137
Figura 70: Resultados que indican si se utilizan tecnologías móviles en las actividades de campo.....	138
Figura 71: Resultados que indican si cuentan con dispositivos móviles del tipo Smartphone	138
Figura 72: Resultados del sistema operativo con el que cuenta su dispositivo móvil.....	139
Figura 73: Resultados que indican si el dispositivo móvil sería de ayuda para realizar actividades de campo.....	140
Figura 74: Resultados que indican si se utilizaría una aplicación móvil que ayude agilizar las actividades de campo	140
Figura 75: Arquitectura MVP.....	143
Figura 76: Paquete con las clases View.....	144
Figura 77: Vista Login.....	144
Figura 78: Vista Inicio.....	145
Figura 79: Vista Observaciones.....	145
Figura 80: Vista registrar acto/condición	146
Figura 81: Vista evidencias	146
Figura 82: Vista configuración	147
Figura 83: Paquete con las clases Presenter	148
Figura 84: Paquete con las clases Model.....	149
Figura 85: Clases Interactor que valida de donde se obtiene la información	149

Figura 86: SDK Retrofit y Gson.....	150
Figura 87: Clase HelperView para iniciar Retrofit.....	150
Figura 88: Etiquetas y métodos de Retrofit.....	151
Figura 89: SDK Okhttp3.....	151
Figura 90: Clases HelperView para iniciar Okhttp3	151
Figura 91: SDK Room.....	152
Figura 92: Clase AppDatabase para crear la base de datos local	152
Figura 93: Etiquetas y métodos de Room.....	152
Figura 94: SDK Firebase Storage.....	152
Figura 95: Adjuntar información a Firebase Storage	153
Figura 96: Descargar información de Firebase Storage	153
Figura 97: SDK Firebase Performance.....	153
Figura 98: Etiquetas de Firebase Performance	153
Figura 99: SDK RecyclerView.....	154
Figura 100: RecyclerView implementado en una vista.....	154
Figura 101: RecyclerView implementado por medio del patrón ViewHolder.....	154
Figura 102: SDK Saripaar	155
Figura 103: Etiquetas de validación Saripaar	155
Figura 104: SDK SearchableSpinner.....	155
Figura 105: SearchableSpinner implementado para mostrar los tipos de observaciones ..	156
Figura 106: SDK SweetAlert.....	156
Figura 107: SweetAlert implementada para mostrar una alerta de progreso	156
Figura 108: Dependencias para incorporar Room.....	159
Figura 109: Clases para las tablas de la base datos local.....	160
Figura 110: Clase ActoCondiciónRoomEntity.....	160
Figura 111: Clase EvidenciaRoomEntity	161
Figura 112: Clase NivelRiesgoRoomEntity	161
Figura 113: Clase ObservacionRoomEntity	162
Figura 114: Clase PersonaObsRoomEntity	162
Figura 115: Clase PersonaRoomEntity.....	163

Figura 116: Clase TipoOcurrenciaRoomEntity	163
Figura 117: Clase ZonaRoomEntity	164
Figura 118: Interfaces para procesar la información de la base datos local	164
Figura 119: Interfaz ActoCondicionTipoDAO.....	165
Figura 120: Interfaz EvidenciaDAO.....	165
Figura 121: Interfaz NivelRiesgoDAO.....	166
Figura 122: Interfaz ObservacionDAO	166
Figura 123: Interfaz PersonaDAO	166
Figura 124: Interfaz TipoOcurrenciaDAO	167
Figura 125: Interfaz ZonaDAO	167
Figura 126: Clase abstracta AppDatabase	167
Figura 127: Interfaces invocadas	168
Figura 128: Base de datos local en memoria caché.....	168
Figura 129: Arquitectura de N capas	171
Figura 130: Arquitectura Yanapay	172
Figura 131: Estructura en Visual Studio.....	173
Figura 132: Diagrama de componentes Yanapay.....	175
Figura 133: Tablas para la seguridad del aplicativo	178
Figura 134: Tablas para el procesamiento de información.....	179
Figura 135: Servidor creado en Amazon LightSail	182
Figura 136: Base de datos Yanapay	182
Figura 137: Servicio Web API desplegado	183
Figura 138: Web API publicado	183
Figura 139: Tiempos de respuesta en iniciar sesión	184
Figura 140: Tiempos de respuesta de listar observaciones.....	184
Figura 141: Tiempos de respuesta en listar categorías	185
Figura 142: Tiempos de respuesta en listar zonas	185
Figura 143: Tiempos de respuesta en listar tipos de ocurrencia.....	186
Figura 144: Tiempos de respuesta en listar niveles de riesgo	186
Figura 145: Tiempos de respuesta en listar personas	187

Figura 146: Tiempos de respuesta en guardar una observación	187
Figura 147: Tiempos de respuesta en guardar una evidencia	188
Figura 148: Rendimiento en vista presentación	189
Figura 149: Rendimiento en vista login	189
Figura 150: Rendimiento en vista menú.....	190
Figura 151: Rendimiento en vista registrar observación	190
Figura 152: Rendimiento en vista registrar evidencia	191
Figura 153: Memoria utilizada en iniciar sesión	192
Figura 154: Memoria utilizada en listar observaciones.....	192
Figura 155: Memoria utilizada en listar categorías	193
Figura 156: Memoria utilizada en listar zonas	193
Figura 157: Memoria utilizada en listar tipos de ocurrencia	194
Figura 158: Memoria utilizada en listar niveles de riesgo.....	194
Figura 159: Memoria utilizada en listar personas	195
Figura 160: Memoria utilizada en guardar una observación	195
Figura 161: Memoria utilizada en guardar una evidencia	196
Figura 162: Memoria utilizada en cargar una evidencia a Firebase Storage.....	196
Figura 163: Memoria utilizada en descargar una evidencia desde Firebase Storage	197

RESUMEN

El presente trabajo de investigación buscó determinar si en un aplicativo móvil lite, el grado de conformidad de la eficiencia de software bajo la norma ISO/IEC 9126-2 es aceptable en inmediaciones de una operación minera; es así que para poder orientar el aplicativo y cumplir con el objetivo planteado se identificó una área y un proceso de campo el cual se realice constantemente en las operaciones mineras, para ello se aplicó una encuesta la cual ayudó a recolectar información necesaria que permitió identificar el proceso al cual fue orientado el aplicativo móvil lite.

Como segundo paso, se desarrolló e implementó el aplicativo utilizando las tecnologías más adecuadas que posibilitaron su funcionamiento más óptimo considerando las características geográficas y de conectividad en inmediaciones de las operaciones mineras, luego de recopilar y analizar la información se obtuvo un 96% de conformidad de la eficiencia con respecto al rendimiento del aplicativo.

Finalmente, luego que se obtuvieron y analizaron los resultados, los indicadores ayudaron a tener otra visión sobre los aplicativos móviles lite, llegando a la conclusión que por medio de ellos también se pueden realizar diferentes tipos de transacciones sin necesidad de demandar grandes cantidades de recursos en los dispositivos móviles.

Palabras clave: dispositivo móvil, aplicativo móvil lite, rendimiento, operaciones mineras.

CAPÍTULO I. INTRODUCCIÓN

1.1. Realidad problemática

En la actualidad, según el Ministerio de Energía y Minas (2020), en su boletín estadístico “Perú, un país minero que supera las adversidades”, existen diversas operaciones mineras operando en el Perú, quienes extraen y procesan sus minerales llegando a producir productos concentrados y metálicos, estas operaciones por lo general están situadas en lugares alejados de las ciudades y cuentan con diferentes áreas que usualmente no se ubican en una zona cercana a la sede principal de la operación. Por ello, en su mayoría, cada responsable y/o colaborador del área cuenta con un dispositivo móvil de gama media o alta, dependiendo el puesto que ocupe, para mantener una comunicación activa y efectiva entre todos los involucrados de la operación minera.

Cabe resaltar que estas operaciones mineras cuentan con toda la infraestructura asociada requerida, como centrales hidroeléctricas, subestaciones, talleres, almacenes, canchas de relaves, planta de tratamiento de aguas ácidas, viviendas y oficinas administrativas; todas estas, conectadas entre sí por un operador de telefonía y una red administrativa, donde por lo general, la cobertura de estas redes no tienen la potencia deseada en algunos lugares alejados de la operación minera por causa de la geografía accidentada donde están desplegadas, generando intermitencia en su señal e incomodidad de los colaboradores cuando estos se disponen a entablar una comunicación o coordinación con otras personas de la operación.

Las operaciones mineras, por desarrollar labores de alto riesgo tanto para la salud como para el medio ambiente, disponen del área de Seguridad y Salud Ocupacional y Medio Ambiente en adelante SSOMA, la cual se encarga de validar las actividades realizadas por el personal, tanto en oficinas como en campo; dentro de las labores que realiza esta área se encuentra la actividad de observar y reportar actos o condiciones sub estándar que ocurran en cualquiera de las áreas de la operación minera.

Para poder realizar esta actividad, cada colaborador del área de SSOMA antes de salir a campo imprime varias fichas (ver anexo N° 1), en estas fichas se detalla información necesaria para reportar actos o condiciones sub estándar, luego de imprimirlas el personal de SSOMA se traslada por las diferentes ubicaciones donde los operarios se encuentren

realizando labores de campo, con la finalidad de poder monitorear las actividades que estos realizan, al encontrarse ubicados en alguna área específica ellos pueden observar si el personal que está laborando incumple con alguna norma o regla de seguridad mientras ejecutan sus actividades diarias; cada vez que logran identificar un acto o condición sub estándar toman una foto con su dispositivo móvil e informan al personal involucrado indicándoles la manera correcta de como ejecutar esa actividad, posterior a ello llenan la ficha que imprimieron antes de salir a campo con información del acto o condición que lograron identificar; al finalizar sus visitas a campo proceden a volver a las oficinas principales de la operación, encontrándose ya en las oficinas, cada personal de SSOMA descarga las imágenes tomadas con su dispositivo móvil a su laptop o computadora, luego de ello transcriben los datos registrados en las fichas impresas y adjuntan las imágenes descargadas como evidencia de lo reportado a un sistema web o formato en Excel (ver anexo N° 2) al cuál tienen acceso únicamente dentro de la red administrativa; finalmente luego de transcribir y adjuntar todos los datos recolectados en campo envían un correo a cada personal involucrado en el acto o condición observada, adjuntando las fotografías como evidencia de lo reportado (ver anexo N° 3).

Considerando que cada personal del área de SSOMA cuenta con un dispositivo móvil el cual les permite tomar fotografías de cada observación reportada, según Aranaz (2009), en un proyecto publicado de “Desarrollo de Aplicaciones para Dispositivos Móviles sobre la Plataforma Android de Google”, se menciona que en los últimos años los dispositivos móviles han tenido una evolución notable con respecto a la capacidad de procesar y almacenar datos gracias a sus sistemas operativos que permiten realizar tareas para gestionar la memoria y controlar el hardware que precisen, estos pueden ser de gran ayuda no solamente para realizar llamadas o escribir y enviar mensajes de texto si no que por sus componentes mejorados pueden ser utilizados para el procesamiento de información mientras estén conectados o no a servicios de internet e incluso pueden ser utilizados para capturar fotografías o vídeos, adicional a ello indica que de acuerdo al tipo o categoría de dispositivo móvil es necesario diseñar sistemas operativos que se adapten específicamente a sus características como las de restricción de memoria y procesamiento, consumo mínimo de energía o gran estabilidad de su funcionamiento.

Teniendo en cuenta que en la actualidad por medio de los dispositivos móviles y gracias a sus componentes mejorados se puede realizar procesamiento de información mientras

estos se encuentren conectados o no a un servicio de internet, se opta por utilizar estos gadgets para realizar el registro de datos del personal de SSOMA mientras ellos realizan sus actividades de supervisión en campo, adicional a ello también se debe considerar que no todos disponen de un dispositivo móvil de gama alta es por ello que sus componentes no siempre van hacer del mismo tipo, pues según Moya (2017), en el artículo “Qué son las apps lite y por qué deberíamos usarlas” publicado en Tekcrispy, refiere que de acuerdo a las características de los dispositivos móviles, no todos soportan aplicaciones móviles que utilicen muchos recursos del sistema operativo, es por ello que se opta por desarrollar las versiones lite o básicas de las aplicaciones originales; estas versiones son consideradas para ser instaladas y utilizadas en dispositivos móviles de gama media o media baja ya que no requieren muchos recursos para su funcionamiento pero que cumplen con la funcionalidad de la aplicación original, además se menciona que estas aplicaciones al ser más sencillas y ligeras poseen funciones muy simples en las que no se visualizan grandes características como: interfaces muy elaboradas o reproducción de animaciones, todo esto con la finalidad de no utilizar muchos recursos de los dispositivos móviles.

Según Moya quien indica que un aplicativo móvil lite puede cumplir con las mismas funcionalidades de una aplicación móvil original y considerando lo indicado por Aranaz quien indica que las aplicaciones móviles pueden realizar procesamiento de información, se opta por que estas aplicaciones deben ser elaboradas siguiendo estándares de calidad con el objetivo de cumplir su propósito de la manera más óptima, garantizando de esta manera la satisfacción de los requerimientos solicitados por los usuarios, pues según Medina (2017), en un artículo publicado sobre “Modelos y Estándares de Calidad Aplicados al Sistema de Información”, indica que la norma ISO/IEC 9126 es utilizada para especificar y evaluar la calidad del software considerando diferentes criterios los cuales están asociados con el desarrollo, el uso, la evaluación, la eficiencia, el mantenimiento, el aseguramiento de la calidad y la auditoria de software.

Entre los antecedentes internacionales, en Madrid, Siabato (2008), en su artículo “Métricas aplicadas a los modelos de calidad: caso de uso en los SIG, realizada en la Universidad Politécnica de Madrid”, señala que el software y los ordenadores, son empleados cada vez más para una gran variedad de aplicaciones, es por este motivo que se considera relevante la selección y desarrollo de productos software de alta calidad, sabiendo que su correcto desarrollo e implementación pueden influir en el éxito o fracaso

de los procesos o tareas para los cuales han sido diseñados. Es así que precisar y evaluar la calidad del software es de vital importancia para las empresas, ya que pueden asegurar el óptimo funcionamiento de sus procesos, esto se puede obtener identificando las características de calidad apropiadas y teniendo claro el uso o propósito para el cual se va desarrollar el producto software.

En Colombia, Guío (2013), en su investigación “Evaluación de las Características de un Sistema de Información con base en la Norma ISO/IC 9126-1”, hace mención que durante los últimos años se ha venido evidenciado un notable avance en la elaboración o desarrollo de tecnologías de la información para las organizaciones indicando que los sistemas de información aportan beneficios en base a la eficiencia y eficacia a las organizaciones que los adoptan, sin embargo, la gran mayoría son desarrollados e implementados sin considerar técnicas o recursos que permitan validar su capacidad de funcionamiento, obteniendo resultados adversos a los esperados; es por ello que se hace imprescindible contar con técnicas o métodos que permitan la comprobación de la calidad de los productos de software que estén orientados a garantizar la satisfacción de las necesidades de los usuarios. También se indica que la calidad de software o de los sistemas de información ha sido una inquietud constante desde los inicios de la informática. En este sentido, se han ideado modelos que facilitan la evaluación de las condiciones idóneas de un producto de software; desde el punto de vista del uso, se considera como software exitoso aquel que satisface las necesidades y expectativas del usuario, incluyendo la calidad como una utilidad general para las personas que interactúan con él.

Asimismo, en Colombia, Bohórquez (2017), en su investigación “Propuesta Metodológica de una aplicación móvil para la gestión de la investigación: uso en diferentes niveles de agregación”, señala que el manejo de herramientas tecnológicas es una realidad latente e ineludible que ingresa a formar parte de esta época, de las actividades y procesos cotidianos en aspectos como la educación e investigación y particularmente en la administración de información de toda índole (personal, académica o laboral). Para esto, las herramientas preferidas resultan siendo los smartphones de uso cada vez más general y constante. Este fenómeno suscita la reflexión acerca de requerir gestionar la información en tiempo real y el manejo de este tipo de herramientas se hace indispensable como respuesta a la necesidad creciente de estar informados y contar con

un acceso ágil desde cualquier punto del planeta. De manera que, se percibe el aumento del uso del teléfono móvil no solo con el objetivo de usarlo con fines de comunicación y cortar distancias, sino que hay en este la posibilidad de utilizarlo con fines de gestionar la información y el conocimiento por medio de aplicaciones que se desarrollen de acuerdo a las necesidades de los usuarios o empresas.

En relación a los tres antecedentes presentados, estos señalan que en la actualidad los dispositivos móviles han tenido una fuerte evolución, permitiendo no solamente realizar llamadas o enviar mensajes de texto, sino que también permiten realizar diferentes tipos de tareas ayudando así al procesamiento de distintos tipos de información, todo esto por medio de aplicaciones que ayuden a impactar de manera positiva los procesos de las empresas u organizaciones; lo cual para obedecer a este impacto de forma positiva es necesario desarrollar un aplicativo que cumpla con ciertos estándares de calidad considerando los lineamientos de la norma ISO 9126, teniendo claro lo detallado se opta por desarrollar un aplicativo móvil lite que ayude al proceso de reportar actos o condiciones sub estándar en las operaciones mineras, cumpliendo con el criterio de la eficiencia que indica la norma ISO 9126.

Continuando con los antecedentes, en Perú, Bendezú y Figueroa (2017), en su investigación “Evaluación de la Eficiencia, según la norma ISO 9126, de un sistema web desarrollado e implementado en el área de ventas y servicios de la empresa Intecsh”, tienen como objetivo cuantificar la eficiencia basándose en la norma ISO 9126 a un sistema web desarrollado e implementado en un área de ventas, no sin antes haber elaborado un análisis del proceso que debió cumplir el sistema web y diseñando una arquitectura eficiente que luego se implementó para la programación de dicho aplicativo; luego de haber desarrollado el aplicativo, este es implementado para posteriormente medir su eficiencia basándose en diferentes métricas de monitoreo web, consiguiendo como resultado final que de los tres indicadores evaluados, dos de ellos, alcanzaron tener una calidad aceptable, logrando el 67% de eficiencia en el sistema web creado. Concluyendo de esta manera que la eficiencia de un aplicativo o producto software es importante para lograr el funcionamiento más óptimo de este, por cuanto, las métricas orientan acciones a considerar en el desarrollo e implementación de un sistema web final.

Considerando que la eficiencia de un producto software es importante para lograr un correcto desempeño entre el sistema y la cantidad de recursos que utiliza y teniendo claro lo indicado por Moya (2017) que no todos los dispositivos móviles soportan las mismas aplicaciones ya que cada dispositivo tiene diferentes tipos de recursos, se utiliza esta característica de la norma ISO 9126 para evaluar el aplicativo móvil lite en operaciones mineras donde no todos los usuarios cuentan con la misma gama de celulares, así mismo se utiliza el grado de conformidad que obtuvieron para realizar una comparación con la calificación que recibe el aplicativo móvil lite.

En el ámbito sobre tecnologías móviles, en Perú, Calsina y Calcina (2017), en su investigación “Sistema de localización basado en dispositivos móviles para el control y monitoreo del personal en el campamento de la empresa minera Vanessasac en el primer trimestre del 2016”, hace hincapié que actualmente vivimos en una sociedad globalizada en la cual la obtención de información por parte de las personas, empresas, entidades públicas y privadas son un factor clave para que estos puedan desarrollar sus tareas o actividades de manera más eficiente. Es por ello que sienten la necesidad de obtener información fiable y concisa en todo momento, independientemente del lugar o la zona donde se ubique el solicitante. También hacen mención que la minería en el Perú es una de las actividades económicas más importantes que impulsan el desarrollo del país. No obstante, esta actividad sigue ubicándose como la de mayor incidencia de accidentes y fatalidades, es por ello que la finalidad de diversos investigadores es poder controlar y atacar el grave problema de la seguridad en la minería; es así que su objetivo principal es determinar la influencia de un sistema de localización en el control y monitoreo del personal en un campamento minero, para ello consideran implementar un aplicativo móvil que evalúe el proceso de control y monitoreo utilizando la función de localización basado en dispositivos móviles. Determinando así, que la utilización masiva de dispositivos móviles con sistema operativo Android en las diferentes actividades que realizan las personas diariamente se adaptan a las actividades que realizan los usuarios en cualquier momento; por lo que el desarrollo e implementación de aplicaciones móviles también son de interés para los diversos investigadores de este rubro, adicional a ello concluyen que la implementación del sistema de monitoreo orientado en dispositivos móviles con sistema operativo Android mejora satisfactoriamente el control y monitoreo del personal en la empresa minera Vanessasac.

Según lo investigado por Calsina y Calcina quienes indican que las actividades en el sector minero siguen siendo las de mayor incidencia de accidentes y fatalidades, la aplicación móvil lite es orientada para apoyar en los trabajos o actividades que realiza el personal de SSOMA con respecto a la seguridad de los trabajadores en las operaciones mineras, contribuyendo de esta manera con una de las áreas más importantes en temas de seguridad del sector minero.

Así mismo, en Perú, Vergara (2016), en su investigación “Desarrollo de una aplicación móvil para apoyar las supervisiones a entidades prestadoras de servicios de Salud”, sostiene que un teléfono móvil con capacidades de cómputo avanzadas es un teléfono inteligente, los teléfonos inteligentes incluyen las funcionalidades de asistente personal digital, reproductores multimedia portátiles, cámaras digitales, navegación GPS y pantallas táctiles de alta definición y navegadores web, el acceso a información en alta velocidad mediante sus sistemas operativos, siendo Android uno de los más populares, el cual fue diseñado para permitir la mejor experiencia de usuario posible en un teléfono móvil. Por otro lado concluye que las tareas de supervisión en campo pueden ser reducidas en su totalidad gracias al diseño e implementación de un proceso mediante un dispositivo móvil que ayude con el registro de la información recolectada por los supervisores, evitando así el uso a cero de papel ya que todo se llevaría a cabo a través de tramas y mensajes, sumando a ello que el tiempo de actualización o sincronización de dicha información se puede reducir hasta cero si el proceso de supervisión se realiza en una zona que cuente con acceso a internet, en caso no tenga cobertura o acceso a internet estima que el tiempo de demora es de 6 horas en caso de estar en Lima y hasta 25 horas estando en provincia.

Es por ello que en referencia a la información obtenida por Vergara y considerando que su proceso tiene mucha similitud con el que se desarrolla en el área de SSOMA, se opta por incluir un modo offline que sirve para registrar información en lugares o zonas donde no existe cobertura de red o esta sea demasiado lenta, de esta manera se asegura que el proceso realizado por los trabajadores no se vea afectado por la deficiencia en la cobertura de red, posteriormente esta información es sincronizada y utilizada en modo online.

Entre los antecedentes locales, Flores y Gonzales (2018), en su tesis “Efecto de la implementación del aplicativo Carpooling, bajo la norma ISO 9126, en la economía de

estudiantes universitarios de Cajamarca”, tiene como objetivo principal llegar a conocer cuál es el impacto económico de los estudiantes de la universidad Privada del Norte al implementar un aplicativo Carpooling (carro compartido), no sin antes hacen hincapié que la calidad de software en el ámbito local y nacional no es la adecuada si la logramos comparar con la de los países líderes en desarrollo de software, es por ello que uno de sus objetivos específicos está centrado en medir la calidad de software basándose en las métricas de calidad externa de la Norma ISO 9126-2. Posterior a ello según la característica de la eficiencia del software y considerando los resultados obtenidos mediante las encuestas realizadas, indican que obtuvieron un valor promedio de 1.59 y 3.43 minutos en ejecutar tareas de manera satisfactoria, un 2.73 y 3.27 segundos en reanudar el aplicativo e iniciar sesión luego de un colapso, un 69.9% de encuestados indican que en el aplicativo CaxasPool les demora 15s en registrar un viaje; datos que son de ayuda para poder contrastar y tener como base en los resultados que se obtienen con la solución móvil bajo la misma norma de calidad.

Por otro lado, Valdivia (2017), en su tesis “Impacto del uso de herramientas de Software en la implementación de Software de Calidad”, tiene como objetivo identificar cual es impacto de utilizar herramientas que evalúen la calidad de código a la hora de desarrollar e implementar un producto software de calidad, quien además menciona que este tipo de evaluación se encuentra dentro de la característica de calidad del software mantenibilidad; llegando a determinar que la utilización de herramientas de calidad de código generan un impacto positivo en la implementación de un producto software de calidad y en comparación con los antecedentes que lo ratifican, se indica que utilizar este tipo de herramientas de calidad a lo largo del ciclo de vida del desarrollo de software garantiza la detección de olores de código, errores y vulnerabilidades; siendo estos los tres puntos esenciales para tener asegurado el desarrollo de un software de calidad, aseverando que logra cumplir con el objetivo general de la investigación.

Adicionalmente recomienda, que no solo las empresas desarrolladoras de software, sino todo profesional y estudiante debe utilizar este tipo de herramientas de calidad de software en sus proyectos de desarrollo ya que tienen resultados positivos en la construcción de software de calidad, dada esta recomendación para poder validar el impacto de la eficiencia de software en el aplicativo móvil lite, se opta por utilizar una

herramienta que permita monitorear el rendimiento del aplicativo mientras este es utilizado.

Por el lado de la evaluación de un aplicativo móvil, Rabanal y Zegarra (2019), en su tesis “Evaluación de una aplicación móvil Android desarrollada en Flutter según las especificaciones de eficiencia de la NTP-ISO/IEC TR 9126-2”, evalúan un aplicativo móvil Android desarrollado en Flutter tomando como referencia las especificaciones de eficiencia que indica la norma ISO/IEC 9126-2, para ello toman en cuenta cinco métricas de eficiencia basándose en las sub características de “Tiempo de Respuesta” y “Utilización de recursos” que plantea dicha norma, las métricas utilizadas son las siguientes: “Tiempo medio de respuesta”, “Cantidad promedio de fotogramas por segundo”, “Tiempo máximo de construcción de fotogramas”, “Tiempo máximo de rasterizado de fotogramas” y “Utilización máxima de memoria”.

En su investigación consideran que estas métricas son necesarias para que un aplicativo móvil tenga un rendimiento y funcionamiento satisfactorio, características que son fundamentales para que un aplicativo móvil funcione de manera óptima en sitios alejados de las operaciones mineras; por otro lado, para que puedan calificar los grados de conformidad consultan a profesionales expertos en el tema con la finalidad que ellos evalúen y planteen un umbral de calificación, llegando a considerar que el aplicativo debe obtener una calificación no menor al 93.8% para que este sea considerado como satisfactorio según las métricas de eficiencia seleccionadas; adicional a ello, luego de haber analizado y calificado las cinco métricas establecidas, concluyen que el aplicativo móvil cumple con el 80% de eficiencia, resultado que es obtenido mediante la métrica de “Conformidad de Eficiencia”.

Considerando que las métricas utilizadas en la investigación de Rabanal y Zegarra son avaladas por expertos para evaluar la eficiencia de un aplicativo móvil basándose en la norma ISO/IEC 9126-2, en la presente investigación se consideran las mismas métricas para la evaluación del aplicativo móvil lite calificándolo con los mismos grados de conformidad.

1.2. Formulación del problema

¿Cuál es el grado de conformidad de la eficiencia de software, bajo la norma ISO/IEC 9126-2, de un aplicativo móvil lite que procese información del área de SSOMA en operaciones mineras?

1.3. Objetivos

1.3.1. Objetivo general

Determinar el grado de conformidad de la eficiencia de software, bajo la norma ISO/IEC 9126-2, de un aplicativo móvil lite que procese información del área SSOMA en operaciones mineras.

1.3.2. Objetivos específicos

- Identificar las actividades de campo y las tecnologías móviles que utilizan en el área de SSOMA.
- Desarrollar e implementar un aplicativo móvil lite que cumpla con una de las actividades de campo del área de SSOMA.
- Evaluar la eficiencia de software del aplicativo móvil lite considerando los criterios de conformidad de la norma ISO/IEC 9126-2 luego de su implementación.

1.4. Hipótesis

1.4.1. Hipótesis general

Un aplicativo móvil lite bajo la norma ISO/IEC 9126-2 para el procesamiento de información del área de SSOMA en operaciones mineras, tiene grado de conformidad satisfactorio respecto a la eficiencia de software.

1.4.2. Hipótesis específicas

- Las actuales actividades de campo del área de SSOMA no utilizan tecnologías móviles que ayuden con el registro de su información.

- El desarrollo e implementación del aplicativo móvil lite se realiza de acuerdo a la actividad de campo que realizan con mayor frecuencia los colaboradores del área de SSOMA.
- La eficiencia de software del aplicativo móvil lite tiene un grado de conformidad satisfactorio según los criterios de la norma ISO/IEC 9126-2.

CAPÍTULO II. METODOLOGÍA

2.1. Tipo de investigación

La presente investigación se encarga de comprobar una determinada hipótesis mediante datos estadísticos de la muestra, datos que luego serán resumidos y presentados interpretando los resultados con el fin de comprobar la hipótesis, siendo así que el tipo de investigación utilizada es descriptiva.

Por el lado del diseño, la investigación es no experimental debido a que los estudios son realizados observando situaciones ya existentes tal cual se dan en su contexto natural para luego ser analizados, siendo así que las variables tanto dependiente como independiente no serán manipuladas (Sampieri 2014).

2.2. Población y muestra (Materiales, instrumentos y métodos)

2.2.1. Unidad de estudio

Cada muestra obtenida del procesamiento de información por la herramienta Firebase Performance Monitoring del aplicativo móvil lite.

2.2.2. Población

Se considera como población a todas las muestras obtenidas desde el 20 de abril del 2020 al 15 de junio del 2020 del procesamiento de información por la herramienta Firebase Performance Monitoring del aplicativo móvil lite instalado en los dispositivos móviles de los colaboradores que laboran en operaciones mineras.

2.2.3. Muestra

La muestra utilizada en la presente investigación es del tipo por conveniencia, basado en Sampieri (2014), quien indica que este tipo de muestra está formada por los casos disponibles a los cuales tenemos acceso y considerando lo indicado por Serrano (2017), menciona que mientras mayor sea el número de la muestra mejor será la precisión de los resultados, se considera como tamaño de la muestra todos los casos obtenidos de la población.

2.2.4. Materiales, instrumentos y métodos

Los materiales que han sido utilizados en la investigación son dos laptops, las cuales fueron de ayuda para el desarrollo del aplicativo y la elaboración del presente documento, además de ello se consideran 11 dispositivos móviles pertenecientes a los diferentes colaboradores quienes apoyaron para que la aplicación sea probada. Como instrumentos para el desarrollo del aplicativo se han utilizado los IDE de Visual Studio 2019, Android Studio, Microsoft SQL Server 2016 y la herramienta Firebase de Google, mientras que, para la elaboración del documento únicamente se utilizó Microsoft Word 2016.

En lo que respecta a la metodología de desarrollo se consideró utilizar Iconix (ver anexo N° 7), teniendo en cuenta que EcuRed la considera como una metodología ágil de fácil uso y que además es utilizada para construir sistemas funcionales de mediana complejidad con la participación de los usuarios sin la necesidad de demandar grandes cantidades de documentación.

2.3. Técnicas e instrumentos de recolección y análisis de datos

Para recolectar información y poder cumplir con el primer objetivo, se utilizó como técnica la encuesta, que fue aplicada a los colaboradores del área de SSOMA que laboran dentro de las operaciones mineras mediante un cuestionario online publicado en la plataforma Google Forms (ver anexo N° 5), la elaboración del cuestionario se realizó por medio de preguntas cerradas considerando lo indicado por Sampieri (2014), quien indica que este tipo de preguntas facilitan la codificación de las respuestas preparándolas para su análisis, además de ello facilitaron identificar los indicadores presentados en la operacionalización de la variable independiente (ver anexo N° 4).

Por otra parte, para cumplir con el tercer objetivo y poder recolectar información sobre el rendimiento y uso de recursos de la aplicación móvil lite cuando esta es utilizada luego de su despliegue, se utilizaron los cuadros de registros que brinda la herramienta Firebase Performance Monitoring estructurando los indicadores que fueron medidos de acorde al proceso principal que realiza la aplicación (ver anexo

N° 3), la elección de la herramienta se realizó considerando lo mencionado en la documentación de Google donde indica que permite recopilar datos sobre las características de rendimiento de las aplicaciones web y móviles, adicional a ello se solicitaron opiniones de expertos que avalen lo indicado por Google (ver anexo N° 6).

Así mismo, considerando que las herramientas utilizadas para la recolección de datos presentan la información en cuadros y gráficos estadísticos y basado en lo indicado por Montes (2018), como técnica para el análisis de datos se utilizó el análisis estadístico mediante el método descriptivo, que consiste en describir un conjunto de datos permitiendo conocer a detalle la información presentada por medio de deducciones directas a partir de los datos obtenidos.

2.4. Procedimiento

2.4.1. Recolección de información

Con la finalidad de recopilar información que ayude a identificar el uso de aplicaciones móviles y el proceso que realizan con mayor demanda los colaboradores del área de SSOMA fuera de las oficinas de la operación minera, se ubicó a diferentes personas que laboran en dicha área, a las cuales se les informó sobre el proyecto que se deseaba elaborar, indicándoles que se tenía como finalidad elaborar un aplicativo que les sea de ayuda para desarrollar sus actividades de campo, así mismo se les indico que la información que ellos brinden iba ser utilizada únicamente con fines educativos para la presente investigación; habiendo informado a las personas se elaboró una encuesta online que les fue proporcionada mediante un enlace publicado en la plataforma Google Forms (ver anexo N° 5), formulario que fue respondido por diferentes colaboradores del área.

Seguidamente, luego que las encuestas fueron completadas, se logró conocer cuál es el proceso de mayor demanda que realiza el personal de SSOMA en sus actividades de campo, el cual fue plasmado en un diagrama de procesos (ver anexo N° 3), así mismo, se identificó cuál es el inconveniente que presentan al desarrollar sus actividades en zonas alejadas a las oficinas principales de las operaciones mineras; posterior a ello se

procedió a desarrollar el aplicativo móvil que contribuya con el proceso identificado resolviendo los inconvenientes que tienen en zonas alejadas, además durante el desarrollo se enlazó a la herramienta Firebase Performance Monitoring con la finalidad de recolectar información y datos estadísticos de su rendimiento mientras el aplicativo era ejecutado y usado por el personal de SSOMA en modo online (ver anexo N° 9). La distribución de la aplicación para su instalación se realizó por medio de un enlace compartido en la plataforma Google Drive, finalmente se capacitó y brindó un manual de usuario a los colaboradores para que puedan interactuar correctamente con el aplicativo (ver anexo N° 8).

2.4.2. Análisis de información

En primer lugar, para interpretar la información recopilada por las encuestas se utilizó los cuadros y gráficos estadísticos que brinda la herramienta Google Forms, esta información ha sido interpretada mediante el método descriptivo, permitiendo de esta manera conocer a detalle los resultados por medio de deducciones directas a partir de los datos obtenidos; datos que lograron responder a los propósitos de los indicadores que se presentan en la operacionalización de la variable independiente (ver anexo N° 4).

Para determinar el grado de eficiencia del aplicativo móvil, se utilizaron los cuadros y gráficos estadísticos que brinda la herramienta Firebase Performance Monitoring, estos resultados son presentados de acuerdo a las muestras obtenidas por la misma herramienta e interpretados por medio de dos tipos de análisis cuantitativos. La interpretación para determinar el tiempo medio de respuesta es presentada de acuerdo a la mediana, según Sampieri (2014) es una de las medidas de tendencia central que ayudan a ubicar los valores de una variable dentro de una escala de medición, reflejando la posición intermedia de la distribución de resultados, para determinar el rendimiento de la interfaz de usuario los resultados son interpretados de acuerdo a una distribución de frecuencias mediante un gráfico de frecuencia, según Sampieri (2014) esta distribución es un conjunto de puntuaciones de una variable ordenadas de acuerdo a sus categorías y los gráficos de frecuencia son de gran utilidad para una mejor

interpretación y descripción de la información, por otro lado, para determinar el uso máximo de memoria, la información es obtenida de la sesión registrada por los diferentes dispositivos móviles.

Finalmente, los resultados obtenidos fueron analizados y evaluados tomando en cuenta las métricas establecidas por Rabanal y Zegarra (2019), quienes utilizan: “Tiempo medio de respuesta”, “Cantidad promedio de fotogramas por segundo”, “Tiempo máximo de construcción de fotogramas”, “Tiempo máximo de rasterizado de fotogramas”, “Utilización máxima de memoria” y “Conformidad de eficiencia” basándose en la norma ISO/IEC 9126-2 para evaluar la eficiencia de software y así asegurar el correcto uso de los recursos garantizando un óptimo rendimiento y funcionamiento, el detalle de las métricas se encuentran en la operacionalización de la variable dependiente (ver anexo N° 4).

CAPÍTULO III. RESULTADOS

A continuación, los resultados que se lograron obtener para la presente investigación serán presentados de acuerdo a los objetivos planteados.

3.1. Objetivo 1: Identificar las actividades de campo y las tecnologías móviles que utilizan en el área de SSOMA.

Con la aplicación de la encuesta se obtuvieron resultados que contribuyeron a alcanzar el propósito del presente objetivo considerando los indicadores de la variable independiente (ver anexo N° 4), dichos resultados son presentados y descritos a continuación:

Tabla 1 Actividades de campo

ENCUESTADO	PREGUNTA	RESPUESTA(S)
1	Según las actividades que realiza en el área, podría indicar cual o cuales son las que implica tener que salir a campo seguidamente	Inspecciones, capacitaciones, control de operaciones.
2		Geología
3		Comunicación con el personal
4		Inspecciones, charlas.
5		Inspecciones
6		Inspecciones
7		Supervisión de equipos y herramientas en buen estado; llenado de AST, Charlas de 5 minutos
8		Supervisor
9		Inspección de SSOMA
10		Verificar la calidad de los trabajos
11		Instalación de equipos de comunicación

Tabla 2 Resultados de las actividades de campo

Respuesta	Cantidad
Inspecciones	5
Capacitaciones	1
Control de operaciones	1
Geología	1
Comunicación con el personal	1
Charlas	2
Supervisión	2
Llenado de AST	1
Verificar la calidad de los trabajos	1
Instalación de equipos de comunicación	1

Según la Tabla 2, “Resultados de las actividades de campo”, se puede evidenciar que la actividad de campo que realizan los colaboradores del área de SSOMA con mayor frecuencia es la de inspeccionar las labores o actividades que desarrollan los trabajadores, la cual demanda recopilar información de los actos o condiciones observadas en distintos lugares o zonas dentro de la operación minera, encontrándose ubicados en su gran mayoría en sitios alejados, donde se logró conocer que el 81.8% de encuestados no cuentan con cobertura de red permanente, viéndose afectada con un 54.5% en socavón y almacenes, 27.3% en tajo, 18.2% en oficinas, 9.1% en laboratorio, balanza y otros como obras y comedores satelitales (ver anexo N° 10).

Por el lado de tecnologías móviles se pudo detectar que el 54.5% de encuestados indicaron que sus empresas no cuentan con herramientas móviles que contribuyan en el desenvolvimiento o avance de sus actividades de campo, a pesar que el 100% de encuestados cuentan con un dispositivo móvil del tipo smartphone y todos con sistema operativo Android (ver anexo N° 11).

Finalmente se realizaron dos preguntas con el objetivo de saber si los encuestados estarían dispuestos a utilizar herramientas de tecnología móvil que ayuden con las tareas de campo, gracias a ello se logró obtener como resultado que el 90.9% cree que su dispositivo móvil le sería de gran apoyo para realizar estas labores en la operación minera y el 100% estaría dispuesto a utilizar un aplicativo móvil que agilice estas actividades (ver anexo N° 12).

3.2. Objetivo 2: Desarrollar e implementar un aplicativo móvil lite que cumpla con una de las actividades de campo del área de SSOMA.

Para alcanzar el propósito del presente objetivo, a continuación, se detalla cómo se elaboró el desarrollo del aplicativo móvil lite según los resultados obtenidos en el objetivo 1:

Según la Tabla 2, “Resultados de las actividades de campo”, el aplicativo se orientó para el proceso que involucra la actividad de inspección, ya que esta es una de las actividades de campo que se realiza con mayor frecuencia, para lograr conocer los

requerimientos que demanda dicha actividad se elaboró un documento de análisis que ayude a identificarlos, este documento se preparó mediante la metodología Iconix. (ver anexo N° 7)

Para lograr desarrollar la interfaz de usuario se apoyó en los resultados de la Figura 72 “Resultados del sistema operativo con el que cuenta su dispositivo móvil” que se muestra en el Anexo N° 11, resultados que indican que el 100% de los dispositivos móviles que utilizan las personas encuestadas tienen el sistema operativo Android, es así que el diseño y desarrollo de la interfaz de usuario ha sido realizado con el IDE de Android Studio. (ver anexo N° 13)

Según la Figura 68, “Resultados de la cobertura de red dentro de la operación minera” que se muestra en el Anexo N° 10, se enfoca en el inconveniente que tiene el personal de SSOMA, el cual es la deficiencia de cobertura de red en distintas zonas de la operación minera, es por ello que una de las funcionalidades que se incorporó al aplicativo es el modo offline, el cual permite realizar el registro de información cuando el dispositivo no cuenta con señal de internet; para cumplir con esta funcionalidad y para el desarrollo de la aplicación móvil lite se escogió la siguiente arquitectura:

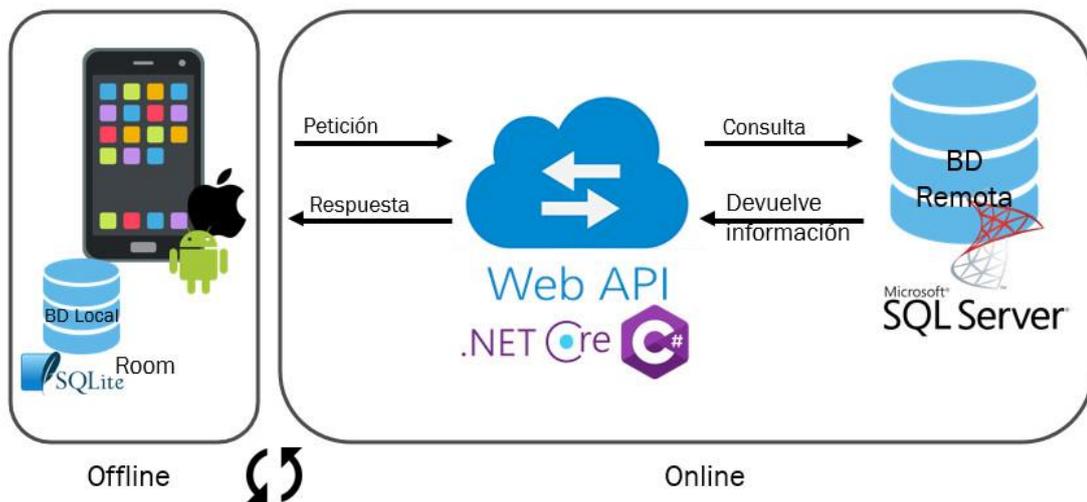


Figura 1: Arquitectura del proyecto móvil

En la Figura 1, “Arquitectura del proyecto móvil”, se puede evidenciar que para cumplir con la funcionalidad del modo offline se utilizó la tecnología SQLite con

la biblioteca de persistencia de datos Room (ver anexo N° 14), la cual permitió registrar la información de manera local para que luego se pudiera sincronizar a una base de datos remota, para esta sincronización de información se utilizó una Web Api que se desarrolló con .Net Core y C# (ver anexo N° 15), esta Web Api es la intermediaria que facilitó la comunicación entre la aplicación móvil y la base de datos remota cuando el aplicativo se encontraba en modo online, la base de datos remota fue creada en Sql Server 2016 (ver anexo N° 16), en ella se almacenaron todos los datos registrados con el aplicativo móvil.

Finalmente, la implementación y despliegue de los servicios que permitieron trabajar en el modo online se ubicaron en un servidor virtual alojado en Amazon Lightsail (ver anexo N° 17), permitiendo realizar consultas hacia la base de datos remota a cualquier momento del día mientras el aplicativo se encontraba conectado a internet. Cabe mencionar que la aplicación desarrollada ha podido ser instalada en dispositivos móviles que cuentan con Android 5.1 Lollipop como versión mínima, con memoria RAM mínima de 1.5 Gb y 1.5 Ghz de velocidad de CPU.

3.3. Objetivo 3: Evaluar la eficiencia de software del aplicativo móvil lite considerando los criterios de conformidad de la norma ISO/IEC 9126 luego de su implementación.

Para cumplir con este objetivo se proporcionó el aplicativo móvil lite al cuál se nombró Yanapay a diferentes usuarios, los cuales estuvieron utilizándolo en el transcurso de sus actividades en operaciones mineras.

Estos resultados son presentados según las métricas de la variable dependiente (ver anexo N° 4) y de acuerdo al modo online en el cual fue utilizada la aplicación, adicional a ello para la evaluación de las métricas se consideraron las tareas que realizan los usuarios mientras se encuentran supervisando sus actividades de campo.

A continuación, se presentan los resultados obtenidos:

3.3.1. Tiempo medio de respuesta

Para identificar el cumplimiento de esta métrica se consideraron los siguientes criterios de conformidad establecidos por Rabanal y Zegarra (2019), siendo las siguientes tareas las que más se asemejaron a la funcionalidad del aplicativo:

Tabla 3 Criterios de Conformidad para Tiempo Medio de Respuesta

Criterio	Tarea	Tiempo medio	Grado conformidad
01	Cargar listas con 20 registros	$T_{\text{medio}} \leq 1.3\text{s}$	Satisfactorio
02	Crear registro	$T_{\text{medio}} \leq 5\text{s}$	Satisfactorio

Fuente: Rabanal y Zegarra (2020)

Tabla 4 Resultados del tiempo medio respuesta

Tarea	Cantidad de muestras	Tiempo medio en segundos	Criterio de evaluación	Grado conformidad
Login o inicio de sesión	143	0.15 s	01	Satisfactorio
Listar observaciones	620	0.058 s	01	Satisfactorio
Listar categorías	156	0.055 s	01	Satisfactorio
Listar zonas	154	0.01 s	01	Satisfactorio
Listar tipo de ocurrencias	143	0.068 s	01	Satisfactorio
Listar nivel de riesgo	151	0.009 s	01	Satisfactorio
Listar personas	147	0.01 s	01	Satisfactorio
Guardar observación	117	0.07 s	02	Satisfactorio
Guardar evidencia	136	0.053 s	02	Satisfactorio

En la Tabla 4, “Resultados del tiempo medio de respuesta”, se puede observar que todas las tareas que fueron analizadas cumplieron con el grado de conformidad “Satisfactorio”, estos tiempos fueron obtenidos de acuerdo al cálculo de la media que brinda la herramienta Firebase Performance Monitoring, así mismo la calificación del grado de conformidad se realizó de acuerdo a la Tabla 3, “Criterios de conformidad para Tiempo Medio de Respuesta”. El detalle de los tiempos obtenidos se puede observar en el Anexo 18.

3.3.2. Rendimiento de la interfaz de usuario

Para lograr analizar y evaluar esta métrica se tomó como base la documentación de fotogramas congelados de Google Developers (2020), donde se menciona

analizar este rendimiento mediante las métricas de procesamiento lento y fotogramas congelados, ambas métricas tienen la misma finalidad que las propuestas por Rabanal y Zegarra (2019), las cuales son garantizar una interacción continua del usuario analizando que el aplicativo deba procesar los fotogramas en menos de 16 ms para lograr una velocidad de 60 FPS (fotogramas por segundo), de lo contrario si el aplicativo tardara en procesar los fotogramas por más de 700 ms estos son bloqueados generando que haya una sensación de lentitud o interrupción en la pantalla.

A continuación, se muestra la información obtenida por la herramienta Firebase Performance Monitoring, la cual recopiló información del procesamiento de interfaz de usuario según las métricas propuestas por Google Developers en las diferentes vistas donde han interactuado los usuarios:

Tabla 5 Resultados del rendimiento de la interfaz de usuario

Vista	N° muestras	Procesamiento lento		Fotogramas congelados		Grado conformidad
		Marcos lentos	Mayor Mmedia	Marcos congelados	Mayor Mmedia	
Vista de presentación	147	77.55 %	Si	5.44 %	No	No satisfactorio
Login o inicio de sesión	186	41.94 %	No	4.3 %	No	Satisfactorio
Menú	780	4.36 %	No	5.64 %	No	Satisfactorio
Registrar observación	546	3.66 %	No	11.17 %	No	Satisfactorio
Registrar evidencia	812	0.62 %	No	0.49 %	No	Satisfactorio

En la Tabla 5, “Resultados del rendimiento de la interfaz de usuario”, se logra visualizar que la vista “Vista de presentación” tuvo un 77.55% de procesamiento lento, lo cual indica que más de la mitad de muestras tuvieron una demora mayor a 16 ms, según la interpretación de Google Developers hubo un procesamiento lento ya que las velocidades de construcción fueron inferiores a los 60 FPS y el 5.44% de muestras tuvieron fotogramas congelados, lo cual indica que menos de la mitad de muestras superaron los 700 ms en procesar un fotograma, según la interpretación de Google Developers indica que la vista no presentó bloqueos ni lentitud en más del 93% de muestras obtenidas; teniendo en cuenta estos resultados se calificó a la vista con el grado de conformidad “No Satisfactorio”, ya que a pesar que no presentaron bloqueos tuvieron un alto porcentaje de procesamiento lento en la construcción de fotogramas.

Por otro lado, en las demás vistas se pudo observar que el porcentaje de procesamiento lento es inferior a la media de las muestras obtenidas, lo cual indica que menos de la mitad de muestras tuvieron una demora mayor a 16 ms, según la interpretación de Google Developers hubo un procesamiento rápido ya que las velocidades de construcción de los fotogramas fueron mayores o iguales a los 60 FPS en más del 87%; así mismo, el porcentaje de fotogramas congelados también se encontraron por debajo de la media de las muestras obtenidas, lo cual indica que menos de la mitad de muestras superaron los 700 ms en procesar un fotograma, según la interpretación de Google Developers indica que la vista no presentó bloqueos ni lentitud en más del 90% de muestras, teniendo en cuenta estos resultados se calificó a las vistas: “Login o inicio de sesión”, “Menú”, “Registrar observación”, “Registrar evidencia” con el grado de conformidad “Satisfactorio”, ya que no presentaron bloqueos y no tuvieron un bajo porcentaje de procesamiento lento en la construcción de fotogramas. El detalle de los porcentajes obtenidos se puede observar en el Anexo 19.

3.3.3. Utilización máxima de memoria

Para evaluar esta métrica se tomó como base la información obtenida de las sesiones de memoria registradas por la herramienta Firebase Performance Monitoring de dos dispositivos móviles que utilizaron el aplicativo, estos dispositivos se consideraron de acuerdo a sus especificaciones siendo uno de ellos un dispositivo con menores recursos que el otro.

Adicionalmente para calificar esta métrica se consideraron los siguientes criterios de conformidad, donde se posicionó al aplicativo en el tipo de aplicación “Estándar”, ya que según Sims (2019) indica que se consideran estándar a las aplicaciones que no cargan muchas imágenes, siendo Yanapay una aplicación lite.

Tabla 6 Criterios de conformidad para utilización de memoria

Tipo Aplicación	Memoria en Montón Máxima	Memoria en Montón Usada	Grado de Conformidad
Estándar	400 Mb	$M_{max} \leq 400 \text{ Mb}$	Satisfactorio

Tipo Aplicación	Memoria en Montón Máxima	Memoria en Montón Usada	Grado de Conformidad
Intensivas en medios	700 Mb	$M_{max} \leq 700 \text{ Mb}$	Satisfactorio
Enormes	1152 Mb	$M_{max} \leq 1152 \text{ Mb}$	Satisfactorio

Fuente: Sims (2019)

Tabla 7 Resultados de la utilización máxima de memoria

Tarea	Memoria en montón máx.	Media de memoria máx.	Memoria en montón usada	Grado conformidad
Login o inicio de sesión	144 MB	72 MB	11.14 MB	Satisfactorio
Listar observaciones	96 MB	48 MB	10.895 MB	Satisfactorio
Listar categorías	240 MB	120 MB	11.55 MB	Satisfactorio
Listar zonas	144 MB	72 MB	8.62 MB	Satisfactorio
Listar tipo de ocurrencias	96 MB	48 MB	8.225 MB	Satisfactorio
Listar nivel de riesgo	144 MB	72 MB	11.135 MB	Satisfactorio
Listar personas	144 MB	72 MB	11.4 MB	Satisfactorio
Guardar observación	144 MB	72 MB	8.54 MB	Satisfactorio
Guardar evidencia	96 MB	48 MB	12.6 MB	Satisfactorio
Cargar evidencia a Firebase Storage	144 MB	72 MB	9.255 MB	Satisfactorio
Descargar evidencia desde Firebase Storage	144 MB	72 MB	11.08 MB	Satisfactorio

En la Tabla 7, “Resultados de la utilización máxima de memoria”, se puede observar que en todas las tareas realizadas la cantidad de memoria en montón usada estuvo por debajo de la media de memoria en montón máxima o asignada, es así, que estas cantidades de memoria en montón usadas fueron calificadas de acuerdo a la Tabla 6, “Criterios de conformidad para utilización de memoria”, logrando obtener el grado de conformidad “Satisfactorio”. El detalle de las cantidades de memoria obtenidas se puede observar en el Anexo 20.

3.3.4. Conformidad de la eficiencia

Luego que se obtuvieron los resultados de las métricas expuestas anteriormente, estas fueron calificadas de acuerdo al criterio de conformidad que se indica en la operacionalización de la variable dependiente (ver anexo N° 4).

Tabla 8 Conformidad de la Eficiencia

Tareas evaluadas	25
Tareas conformes	24
Conformidad	96%

En la tabla 8, “Conformidad de la Eficiencia”, se puede visualizar que hubo un total de 25 tareas analizadas correspondientes a las tres métricas evaluadas de las cuales 24 cumplieron con los grados de conformidad establecidos, utilizando la fórmula $X = A / B * 100$ donde A fue la cantidad de métricas conformes y B el total de métricas evaluadas, se logró obtener el 96% de conformidad de eficiencia; este porcentaje fue calificado de acuerdo al criterio de conformidad que se indica en la operacionalización de la variable dependiente (ver anexo N° 4), logrando obtener el grado de conformidad “Satisfactorio” ya que el porcentaje de conformidad es menor al 93.8% que se establece como criterio aceptable.

CAPÍTULO IV. DISCUSIÓN Y CONCLUSIONES

1.1. Discusión

En las operaciones mineras se desarrollan trabajos de alto riesgo los cuales deben ser controlados por el personal de seguridad para evitar daños hacia las personas o daños materiales, es por ello que estos colaboradores salen a campo a realizar inspecciones sin contar con algún aplicativo que les faciliten el levantamiento de información, es así que se pudo evidenciar que para este rubro este tipo de aplicaciones aún no están siendo explotadas de acorde a las nuevas tecnologías que se dispone en la actualidad, esto es ocasionado a que uno de los impedimentos más resaltantes es la falta de cobertura de red en distintas zonas donde labora el personal minero, ocasionando que algunas empresas sigan utilizando formatos en papel para el registro de información en campo.

La presente investigación estuvo dirigida hacia las tareas de campo más concurrentes que realizan los colaboradores del área SSOMA, pudiendo identificar que en su totalidad las personas que realizan trabajos en campo cuentan con un dispositivo móvil a la mano, siéndoles de gran ayuda para registrar sus evidencias mediante las cámaras fotográficas que tienen incorporadas o para realizar coordinaciones con otros encargados de las diversas áreas o contratas. Es por ello que se consideró el desarrollo de una aplicación para este tipo de plataformas móviles, ya que en la actualidad estos dispositivos permiten realizar diferentes tipos de procesamiento de información de manera más rápida y simplificada, estando de acuerdo con Aranaz (2009), quien indica que los dispositivos móviles tienen la capacidad de procesar y almacenar datos gracias a sus sistemas operativos y componentes mejorados mientras se encuentren conectados o no a servicios de internet, adicionalmente considerando que estos dispositivos pueden almacenar información estén o no conectados a internet se implementó una función de modo offline que ayudó a realizar los registros de información donde la cobertura de red no exista o esta sea demasiado débil, dando la facilidad de poderla sincronizar cuando el dispositivo se encuentre conectado a una red de Wifi o su cobertura de red sea más estable que en campo, aceptando de esta manera lo indicado por Vergara (2016), quien concluye que esta funcionalidad es de gran ayuda para las tareas

de supervisión que se realizan en campo, pudiendo simplificar los tiempos de sincronización a cero en caso los dispositivos cuenten con señal o cobertura de internet.

Por otro lado, se tuvo en consideración que no todos los dispositivos móviles tienen las mismas especificaciones, es así que el aplicativo fue desarrollado teniendo en cuenta los criterios de una aplicación del tipo lite, siendo esta de gran ayuda para procesar información en diferentes zonas de las operaciones mineras sin la obligación de contar con un dispositivo móvil de gama alta, tal y como lo señala Moya (2017), quien indica que dependiendo de las especificaciones o características de los dispositivos móviles no todos soportan aplicaciones que utilicen muchos recursos del sistema operativo, es por ello que se opta por desarrollar las versiones lite o básicas que cumplen con la misma función de las versiones originales pero sin la necesidad de sobrecargarlas de imágenes o animaciones que consuman recursos del dispositivo.

Así mismo con el desarrollo e implementación del aplicativo móvil lite Yanapay, se consideró que siendo un producto software este debió cumplir con algún estándar de calidad que asegure el correcto funcionamiento y garantice la satisfacción de los usuarios o personas que la utilizan, es por ello que se evaluó dicho aplicativo según la norma ISO/IEC 9126-2 tomando en cuenta lo indicado por Guío (2013) y Siabato (2008), quienes señalan que la ISO 9126 es un estándar de mecanismos que posibilitan la verificación de la calidad de los productos de software que estén orientados a garantizar la satisfacción de las necesidades de los usuarios; teniendo en cuenta estos criterios y considerando que el aplicativo estuvo orientado para que se despliegue en cualquier dispositivo móvil con sistema operativo Android y sea utilizado en cualquier zona de las operaciones mineras, la evaluación se realizó bajo las métricas de la eficiencia aceptando lo señalado por Bendezú y Figueroa (2017) quienes indican que estas métricas aseguran la correcta utilización de recursos de los dispositivos donde han sido desplegados, garantizando un correcto rendimiento y funcionamiento.

Finalmente, para lograr calificar al aplicativo según la norma ISO/IEC 9126-2 se tomaron como base los criterios de calificación que proponen Rabanal y

Zegarra (2019), quienes evalúan la eficiencia de un aplicativo móvil desarrollado con el Framework Flutter de Google, obteniendo un 80% de conformidad, mientras que el aplicativo móvil lite Yanapay desarrollado nativamente con Java y Android Studio obtuvo un 96% de cumplimiento, calificándolo con el grado de conformidad “Satisfactorio”; por otro lado luego de analizar la eficiencia del aplicativo se concuerda con BendeZú y Figueroa (2017), quienes indican que la eficiencia de un sistema es importante, por cuanto, las métricas orientan acciones a tomar en cuanto a la programación e implementación de un sistema; así mismo cabe mencionar que para determinar el grado de conformidad de la aplicación, esta fue vinculada a la herramienta de calidad Firebase Performance Monitoring de Google, la cual se encargó de monitorear las actividades que realizan los usuarios registrando información del rendimiento del aplicativo de acuerdo al dispositivo móvil y versión del sistema operativo en el que se encontraba instalado, estando de acuerdo con Valdivia (2017), quien recomienda no solo a las empresas desarrolladoras de software, sino que a todo profesional y estudiante deba utilizar herramientas de calidad de software en sus proyectos de desarrollo ya que tienen resultados positivos en la construcción de software de calidad.

Cabe resaltar que para obtener los resultados que ayudaron a validar la eficiencia del aplicativo Yanapay, este fue alojado en una carpeta compartida de Google Drive y proporcionado a diversas personas que laboran en empresas contratistas que brindan servicios en operaciones mineras, así mismo, la herramienta Firebase Performance Monitoring obtiene información únicamente cuando el dispositivo se encuentra conectado a una red, es así que la información obtenida en los resultados de esta investigación ha sido recolectada cuando el aplicativo funcionaba en el modo online, considerando esto como una limitación para lograr adquirir los datos de rendimiento cuando el dispositivo funcione en modo offline.

1.2. Conclusiones

El aplicativo móvil lite orientado para el procesamiento de información en las actividades de campo que realiza el personal del área de SSOMA en operaciones mineras cumple con las métricas de eficiencia que establece la norma ISO/IEC

9126-2, cumpliendo con el 96% de las métricas propuestas y obteniendo el grado de conformidad “Satisfactorio”, con este resultado se puede contrastar la hipótesis y decir que en efecto la aplicación móvil lite cumple con lo planteado.

Se ha logrado identificar que la actividad de campo más demandante por los colaboradores del área de SSOMA es la de inspeccionar a los trabajadores durante el desarrollo de sus tareas diarias, la cual demanda recolectar información de los actos o condiciones observadas en distintas zonas de la operación minera, las cuales en su gran mayoría se encuentran ubicadas en áreas alejadas donde el 81.8% de encuestados indican que no cuentan con cobertura de red en todo momento siendo almacenes y socavón las áreas más afectadas; así mismo se ha podido saber que el 54.5% de personas no cuentan con aplicaciones o herramientas de tecnología móvil adecuadas por parte del área o empresa para cumplir con sus labores diarias pero por otro lado el 100% de ellos si cuenta con un dispositivo móvil del tipo smartphone el cuál llevan consigo en todo momento.

Se ha logrado desarrollar y desplegar un aplicativo móvil lite el cual cumple con la actividad de inspeccionar a los trabajadores durante el desarrollo de sus tareas diarias, enfocado en registrar actos o condiciones observadas, adicionalmente ha sido incorporada la funcionalidad de un modo offline logrando de esta manera manejar el inconveniente que tiene el personal de SSOMA cuando se encuentran en zonas donde no existe cobertura de red o esta es demasiado débil.

Se ha evaluado que el aplicativo Yanapay ha podido ser utilizado sin ninguna dificultad en los diferentes dispositivos móviles donde ha sido implementado y que cumple con las métricas de eficiencia propuestas bajo la norma ISO/IEC 9126-2, determinando de esta manera que las aplicaciones móvil lite cumplen con el estándar de calidad de eficiencia para que puedan ser implementadas en diferentes tipos de dispositivos móviles sin necesidad de que estos sean dispositivos de gama alta, ya que no consumen o demandan tantos recursos y siguen cumpliendo con el objetivo para el cual han sido desarrolladas.

REFERENCIAS

- Ministerio de Energía y Minas. (2020, enero). PERÚ, UN PAÍS MINERO QUE SUPERA LAS ADVERSIDADES. Recuperado de <http://www.minem.gob.pe/minem/archivos/file/Mineria/PUBLICACIONES/VARIABLES/2020/BEMENE2020.pdf>
- Aranaz Tudela, J. (2009). *Desarrollo de aplicaciones para dispositivos móviles sobre la plataforma Android de Google.*
- Moya, D. (2017, diciembre 22). Qué son las «apps lite» y por qué deberíamos usarlas. Recuperado 6 de mayo de 2020, de TekCrispy website: <https://www.tekcrispy.com/2017/12/22/apps-lite-que-son/>
- Medina Prieto, R. (2017) Modelos y estándares de calidad aplicados al sistema de información. Recuperado 6 de mayo de 2020, de 4.2. La norma ISO/IEC 9126 website: <http://unidad4rociomp.blogspot.com/2017/07/41.html>
- Siabato, W. (2008). *Métricas aplicadas a los modelos de calidad: Caso de uso en los SIG.*
- Guío Ávila, H. A. (2013). Evaluación de las características de un sistema de información con base en la norma ISO/IEC 9126-1. *Signos: Investigación en sistemas de gestión*, 5(2), 33-44.
- Bohórquez Garzón, D. P. (2017). *Propuesta metodológica de una aplicación móvil para la gestión de la investigación: Uso en diferentes niveles de agregación.* Recuperado de <http://repository.javeriana.edu.co/handle/10554/36188>
- Bendezú Cabello, J. C., & Figueroa Ferrer, C. W. (2017). evaluación de la eficiencia, según la norma ISO 9126, de un sistema web desarrollado e implementado en el área de ventas y servicios de la empresa INTECSH. *Universidad de Huánuco.*
- Calsina Paredes, A., & Calcina Paredes, W. (2017). Sistema de localización basado en dispositivos móviles para el control y monitoreo del personal en el campamento de la empresa minera VANESSASAC en el primer trimestre del 2016. *Universidad Nacional del Altiplano.* Recuperado de <http://repositorio.unap.edu.pe/handle/UNAP/6641>
- Vergara Pérez, R. Y. (2016). Desarrollo de una aplicación móvil para apoyar las supervisiones a entidades prestadoras de servicios de salud. *Repositorio de Tesis*

- *UNMSM*. Recuperado de <https://cybertesis.unmsm.edu.pe/handle/20.500.12672/5302>
- Flores Martos, C. Y., & Gonzales Otiniano, J. E. (2018). Efecto de la implementación del aplicativo Carpooling, bajo la norma ISO 9126, en la economía de estudiantes universitarios de Cajamarca. *Universidad Privada del Norte*. Recuperado de <https://repositorio.upn.edu.pe/handle/11537/13038>
- Valdivia Huamán, D. A. (2017). Impacto del uso de herramientas de software en la implementación de software de calidad. *Universidad Privada del Norte*. Recuperado de <https://repositorio.upn.edu.pe/handle/11537/12451>
- Rabanal Rojas, G. E., & Zegarra Zavala, J. J. (2020). Evaluación de una aplicación móvil android desarrollada en flutter según las especificaciones de eficiencia de la NTP-ISO/IEC TR 9126-2. *Universidad Privada del Norte*. Recuperado de <https://repositorio.upn.edu.pe/handle/11537/23596>
- Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, P. (2014). *Metodología de la investigación*. México, D.F.: McGraw-Hill Education.
- Serrano Angulo, J. (2017). Sobre la población y muestra en investigaciones empíricas [Billet]. Recuperado 17 de mayo de 2020, de Aula Magna 2.0 website: <https://cuedespyd.hypotheses.org/2353>
- ICONIX - EcuRed. (2013). Recuperado 12 de junio de 2020, de <https://www.ecured.cu/ICONIX>
- Montes, D. (2018, septiembre 12). Métodos de Análisis Estadístico. Recuperado 13 de junio de 2020, de Proyectos Gestión Conocimiento website: <https://www.pgconocimiento.com/metodos-de-analisis-estadistico/>
- Fotogramas congelados | Desarrolladores de Android. (2020). Recuperado 21 de mayo de 2020, de Android Developers website: <https://developer.android.com/topic/performance/vitals/frozen?hl=es-419>
- Sims, G. (2019, enero 28). How much RAM does your phone REALLY need in 2019?. Recuperado 6 de mayo de 2020, de Android Authority website: <https://www.androidauthority.com/how-much-ram-do-you-need-in-smartphone-2019-944920/>
- Castillero Mimenza, O. (2017, agosto 24). Tipos de gráficas: Las diversas maneras de representar datos. Recuperado 19 de junio de 2020, de <https://psicologiyamente.com/miscelanea/tipos-de-graficas>

- Biblioteca de persistencias Room | Desarrolladores de Android. (2020). Recuperado 6 de mayo de 2020, de Android Developers website: <https://developer.android.com/topic/libraries/architecture/room?hl=es-419>
- Modelo Vista Presentador (MVP) en Android. (2016). Recuperado 6 de mayo de 2020, de Develapps website: <http://www.develapps.com/es/noticias/modelo-vista-presentador-mvp-en-android>
- Sube archivos en Android. (2020). Recuperado 6 de mayo de 2020, de Firebase website: <https://firebase.google.com/docs/storage/android/upload-files?hl=es>
- Descarga archivos en Android. (2020). Recuperado 6 de mayo de 2020, de Firebase website: <https://firebase.google.com/docs/storage/android/download-files?hl=es>
- Controla el rendimiento técnico de tu app con Android vitals—Ayuda de Play Console. (2020). Recuperado 21 de mayo de 2020, de <https://support.google.com/googleplay/android-developer/answer/7385505?hl=es-419>
- Inspect GPU rendering speed and overdraw | Desarrolladores de Android. (2020). Recuperado 21 de mayo de 2020, de Android Developers website: <https://developer.android.com/topic/performance/rendering/inspect-gpu-rendering?hl=es-419>
- Procesamiento lento | Desarrolladores de Android. (2020). Recuperado 21 de mayo de 2020, de Android Developers website: <https://developer.android.com/topic/performance/vitals/render?hl=es-419>
- Realiza análisis con el procesamiento de perfiles de la GPU. (2020). Recuperado 21 de mayo de 2020, de Android Developers website: <https://developer.android.com/topic/performance/rendering/profile-gpu?hl=es-419>
- Tipos de gráficos de Hojas de cálculo de Google—Ayuda de Editores de Docs. (2020). Recuperado 19 de junio de 2020, de <https://support.google.com/docs/answer/190718?hl=es-419>
- Documentación. (2020). Recuperado 19 de junio de 2020, de Firebase website: <https://firebase.google.com/docs?hl=es>
- Firebase Performance Monitoring. (2020). Recuperado 19 de junio de 2020, de Firebase website: <https://firebase.google.com/docs/perf-mon?hl=es>

ANEXOS

Anexo N° 1. Ficha de campo para registrar actos o condiciones observadas

REPORTE DE ACTITUDES , CONDICIONES E INCIDENTES		SMEB E07.01-F01 V. 01
SEGURO	M. AMBIENTE	SALUD
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
REPORTADO POR: _____		
LUGAR DEL REPORTE: _____		
FECHA: _____		HORA: _____
TURNO: _____	GERENCIA _____	
ACTO SUBESTANDAR <input type="checkbox"/>	CONDICION SUBESTANDAR <input type="checkbox"/>	INCIDENTE <input type="checkbox"/>
ACTO POSITIVO <input type="checkbox"/>		
GRAVEDAD POTENCIAL (colocar el Valor - Matriz IPERC)		
ALTO <input type="checkbox"/>	MEDIO <input type="checkbox"/>	BAJO <input type="checkbox"/>
QUÉ OBSERVÓ (DESCRIBIR)		
_____ _____ _____ _____		
ACTIVIDAD DE RIESGO RELACIONADA A LA OBSERVACION (VER LISTADO EN PARTE POSTERIOR)		

MARCAR EL NUMERO DE LISTA DE TIPOS DE OCURRENCIA		<input type="checkbox"/>
¿QUÉ MEDIDA TOMO AL RESPECTO?		
_____ _____ _____ _____		
PLAZO DE CUMPLIMIENTO :	<input type="text"/>	<input type="text"/>
_____ FIRMA REPORTANTE		

Figura 2: Ficha de campo

Anexo N° 2. Formatos en Excel para el registro de inspecciones

A		B		C		D		E		F		G		H		I		J		K		L		M		N		O		P		Q		R		S		T		U		V		W		X		Y		Z		AA		AB		AC		AD		AE		AF		AG		AH		AI		AJ	
				SISTEMA INTEGRADO DE GESTION																				CÓDIGO: PRO-SSO-11																																															
				ANEXO 2:																				VERSIÓN: 01																																															
				FORMATO ELECTRÓNICO PARA REPORTE DE INSPECCIONES																				FECHA: 25/12/2017																																															
				Página 01 de 01																																																																			
		AREA INSPECCIONADA:		PRODUCCIÓN																																																																			
		REALIZADA POR:																																																																					
		FECHA DE INSPECCION:		martes, 11 de Febrero de 2020																																																																			
N°		Observación o Recomendación		Acción		Responsable		Fecha de Cierre		Fotos Antes		Fotos Después																																																											
1		Se observó en el área de cocina caliente (Producción) el piso mojado, pudiendo ocasionar una caída de los colaboradores.		Retroalimentar al personal de cocina caliente sobre la importancia del secado del piso y señalización respectiva.		Walter Toribio		11/02/2020																																																															
2		Se observó la balanza del área de postres (Producción) una balanza sin el color de cinta trimestral correspondiente.		Capacitar al personal de Postres sobre la importancia de colocar la cinta trimestral a los equipos y/o herramientas.		Walter Toribio		11/02/2020																																																															

Figura 3: Formato Excel para reporte de inspecciones

Nro	Tipo SSO / A.M.	Fecha	Mes	Inspector(es)	Empresa	Descripción de la observación (Peligro/Riesgo/Consecuencia / Aspecto/Incidencia Ambiental)	Foto hallazgo	Clasificación por tipo	Nivel de riesgo	Medidas Correctivas/Inmediata y Definitiva	Resp
1	Seguridad y Salud Ocupacional	25/11/2019	enero			Suficiencia Deficiente de Gabinete de comunicaciones de Edificio A - Piso 1		Sistema de advertencia/ aviso inadecuado	Bajo	Se programó señalización y etiquetado de gabinete, para evitar que el personal que labora en esta oficina se golpee la cabeza con el gabinete.	F
2	Seguridad y Salud Ocupacional	25/11/2019	enero	F		Falta de Orden de cableado de Gabinete de Comunicaciones de Edificio A - Piso 1		Orden y Limpieza deficientes	Bajo	Se programó ordenamiento, limpieza y reemplazo de cableado.	R
3	Seguridad y Salud Ocupacional	25/11/2019	enero			Suficiencia Deficiente de Gabinete de comunicaciones de Edificio A - Piso 2		Sistema de advertencia/ aviso inadecuado	Bajo	Se programó señalización y etiquetado de gabinete, para evitar que el personal que labora en esta oficina se golpee la cabeza con el gabinete.	F
4	Seguridad y Salud Ocupacional	25/11/2019	enero	ago		Falta de Orden de cableado de Gabinete de Comunicaciones de Edificio A - Piso 2		Orden y Limpieza deficientes	Bajo	Se programó ordenamiento, limpieza y reemplazo de cableado.	F
5	Seguridad y Salud Ocupacional	*****	febrero	po		3 luminarias de los 5 existentes en el Datacenter de Pasco están quemadas e inoperativas.		Iluminación deficiente o excesiva	Bajo	Se solicitó al área de Servicios Generales el reemplazo inmediato de las luminarias inoperativas.	I
6	Seguridad y Salud Ocupacional	*****	febrero	po		Presado de un CPU de servidor en el pasadizo del Datacenter.		Contorno, piso, superficies, inadecuado	Bajo	Se realizó la coordinación para trasladar este equipo al Contenedor de servidor en el P12, liberando el pasadizo del Datacenter, evitando de esta manera que el personal tropiece y se haga daño.	I

Figura 4: Formato Excel para reporte de actos y condiciones

Anexo N° 3. Flujo del proceso para registrar actos o condiciones observadas

PROCESO PARA REGISTRAR ACTOS O CONDICIONES OBSERVADAS

ALCANCE

Este documento muestra un diagrama de flujo en el cual se puede apreciar el proceso que desempeña un ingeniero de seguridad para reportar actos y condiciones cuando realiza labores de supervisión en campo.

El diagrama de flujo del proceso ha sido orientado de acuerdo a la experiencia en trabajos realizados en inmediaciones de una operación minería, las cuales involucraban hacer inspecciones de seguridad en diferentes áreas y zonas de la operación

Asu vez este documento muestra una ficha de validación por parte de expertos en seguridad y salud ocupacional y supervisores de proyectos, los cuales validaron el proceso realizado en operaciones mineras.

1. DIAGRAMA DE FLUJO

El proceso plasmado en el siguiente diagrama de flujo está orientado a la tarea de reportar actos o condiciones observadas en zonas alejadas o que se encuentren fuera de las oficinas principales de la operación minera

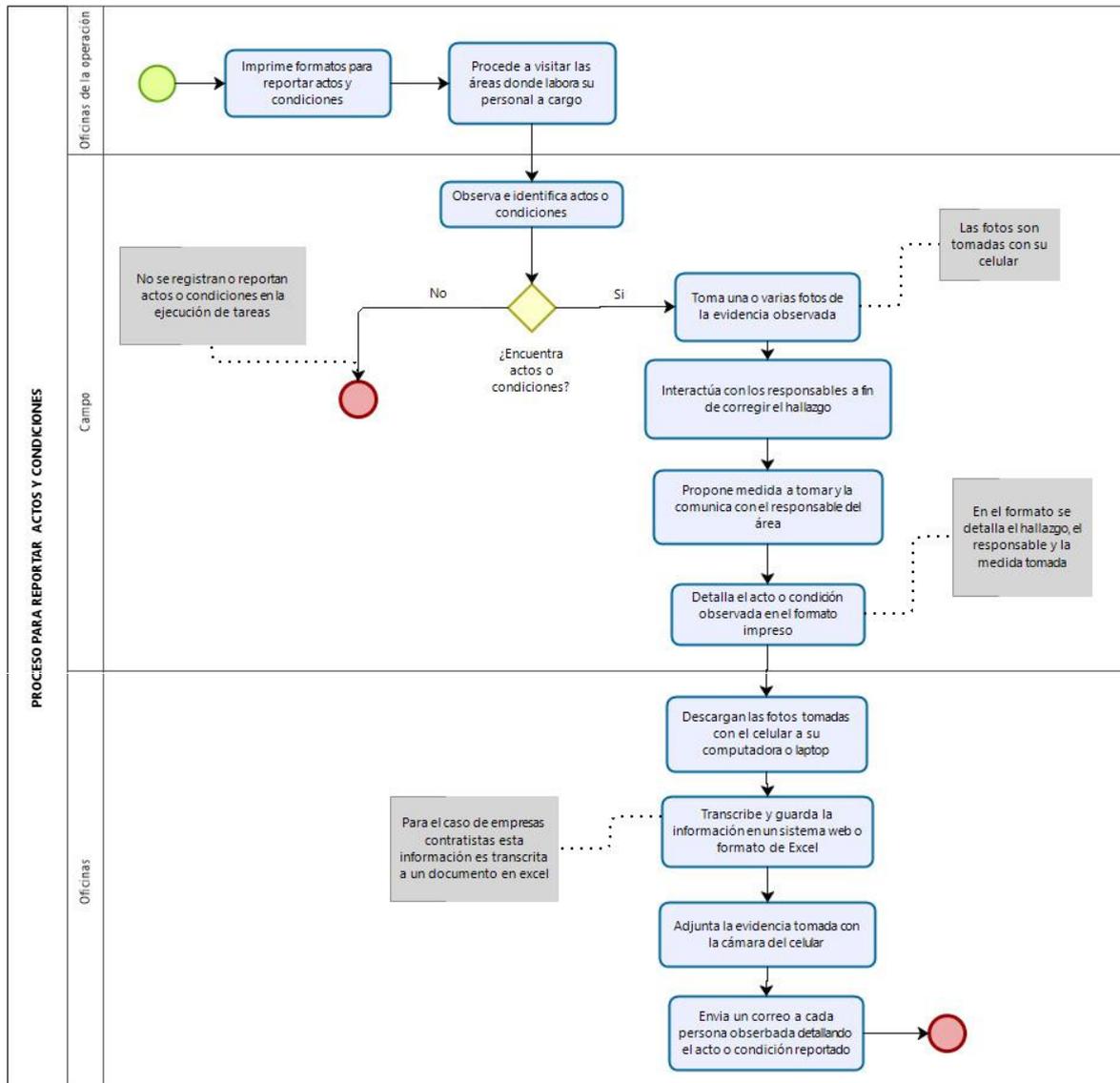


Figura 5: Diagrama del proceso para reportar actos y condiciones

2. VALIDACIÓN DE EXPERTOS



UNIVERSIDAD
PRIVADA DEL NORTE

FICHA PARA VALIDACION DEL PROCESO

I. REFERENCIA

1.1. Experto: NONTOL COTRINA ERICK DANIEL

1.2. Especialidad: INGENIERO INDUSTRIAL

1.3. Cargo actual: COORDINADOR SSOMA

1.4. Grado académico: UNIVERSITARIO

1.5. Institución: SEGUROS ALIMENTARIOS CAROLINOS

1.6. Nombre del proceso: PROCESO PARA REPORTAR ACTOS Y ACCIONES

1.7. Lugar y fecha: 01 JULIO 2020

II. TABLA DE VALORACION

N°	EVIDENCIAS	VALORACION					
		5	4	3	2	1	0
1	Se identifica claramente el proceso principal	✓					
2	Las tareas del diagrama coinciden con las actividades	✓					
3	Las tareas tienen secuencia lógica	✓					
4	Se puede comprender fácilmente	✓					
5	Favorece al entendimiento del proceso	✓					
	Total	100%					

Coefficiente de valoración porcentual: $c = 100\%$

III. OBSERVACIONES Y/O RECOMENDACIONES

CONSIDERAR AL MOMENTO DE LA INSPECCION
DESCOMPOSICION CLARA DEL HALLAZGO Y MEDIDA
INMEDIATAS Y FUTURAS PARA EVITAR
LA RECURSIVIDAD



Firma y sello del Experto

ERICK DANIEL NONTOL COTRINA
Ingeniero Industrial
Reg. CIP. N° 715615



FICHA PARA VALIDACION DEL PROCESO

I. REFERENCIA

- 1.1. Experto: *Tito Arturo Lozano Córdova*.....
- 1.2. Especialidad: Seguridad y Salud en el Trabajo.....
- 1.3. Cargo actual: Supervisor HSE Proyecto Marcobre.....
- 1.4. Grado académico: Universitario
- 1.5. Institución: Universidad Privada del Norte.....
- 1.6. Nombre del proceso: Registro de actos o condiciones observadas
- 1.7. Lugar y fecha: Cajamarca, 20 Jun-2020

II. TABLA DE VALORACION

N°	EVIDENCIAS	VALORACION					
		5	4	3	2	1	0
1	Se identifica claramente el proceso principal	x					
2	Las tareas del diagrama coinciden con las actividades		x				
3	Las tareas tienen secuencia lógica	x					
4	Se puede comprender fácilmente	x					
5	Favorece al entendimiento del proceso		x				
	Total	15	8				

Coefficiente de valoración porcentual: $c = 92\%$

III. OBSERVACIONES Y/O RECOMENDACIONES

Incluir medida o acción realizada durante la ejecución de la observación de acto y/o condiciones en la ejecución de tareas.

.....

.....

.....

.....



Firma y sello del Experto



FICHA PARA VALIDACION DEL PROCESO

- I. REFERENCIA**
- 1.1. Experto: Rafael Edinson Escobar Portal
 - 1.2. Especialidad: Gestión de Proyectos
 - 1.3. Cargo actual: Supervisor de Operaciones
 - 1.4. Grado académico: Msc. de Sistemas - Titulado
 - 1.5. Institución: Mine Sense Solutions
 - 1.6. Nombre del proceso: Proceso para registrar datos o condiciones observadas
 - 1.7. Lugar y fecha: Cajamarca, 20 de junio 2020

II. TABLA DE VALORACION

N°	EVIDENCIAS	VALORACION					
		5	4	3	2	1	0
1	Se identifica claramente el proceso principal	X					
2	Las tareas del diagrama coinciden con las actividades	X					
3	Las tareas tienen secuencia lógica			X			
4	Se puede comprender fácilmente	X					
5	Favorece al entendimiento del proceso		X				
Total							

Coefficiente de valoración porcentual: c =

- III. OBSERVACIONES Y/O RECOMENDACIONES**
- Se debería considerar en el flujo que algunas inspecciones son planificadas.
-
-
-

Firma y sello del Experto

Rafael E. Escobar Portal
SUPERVISOR DE OPERACIONES
Mine Sense Solutions Services S.A.C.



FICHA PARA VALIDACION DEL PROCESO

I. REFERENCIA

- 1.1. Experto: Luis Alberto Pereira Araujo
 1.2. Especialidad: Gestión de Proyectos
 1.3. Cargo actual: Supervisor de Proyectos
 1.4. Grado académico: Ing. de Sistemas - Colegiado
 1.5. Institución: DXC
 1.6. Nombre del proceso: Proceso para registrar actos o condiciones observado
 1.7. Lugar y fecha: Cajamarca 23 de junio del 2020

II. TABLA DE VALORACION

N°	EVIDENCIAS	VALORACION					
		5	4	3	2	1	0
1	Se identifica claramente el proceso principal	X					
2	Las tareas del diagrama coinciden con las actividades	X					
3	Las tareas tienen secuencia lógica			X			
4	Se puede comprender fácilmente	X					
5	Favorece al entendimiento del proceso			X			
	Total						

Coefficiente de valoración porcentual: $c = \dots\dots\dots$

III. OBSERVACIONES Y/O RECOMENDACIONES

.....


 Luis Alberto Pereira Araujo
 Ingeniero de Sistemas
 Reg.CIP. N° 199915

Firma y sello del Experto

Anexo N° 4. Operacionalización de Variables de Estudio

Operacionalización de variable dependiente

VARIABLE	DEFINICIÓN	DIMENSIONES	INDICADORES	MÉTRICAS	FÓRMULA	PROPÓSITO	INSTRUMENTO DE MEDICIÓN
Grado de conformidad de la eficiencia de software bajo la norma ISO/IEC 9126-2	Aprobación de la eficiencia medida a través de atributos como el consumo de tiempo y la utilización de recursos de un sistema de computación, incluyendo el software durante las pruebas o las operaciones (NTP-ISO/IEC TR 9126-2).	Eficiencia	Comportamiento en el tiempo	Tiempo medio de respuesta	<p>Con base en la NTP ISO/IEC 9126-2 para medir el tiempo de respuesta se debe calcular de la siguiente manera:</p> $T_{\text{medio}} = \frac{\sum(T_i)}{N}$ <p>Donde:</p> <p>Ti: tiempo de ejecución de la tarea para la observación i</p> <p>N: número de observaciones</p> <p><i>Interpretación:</i></p>	Determinar el tiempo medio que toma ejecutar una tarea específica.	Firestore Performance Monitoring

VARIABLE	DEFINICIÓN	DIMENSIONES	INDICADORES	MÉTRICAS	FÓRMULA	PROPÓSITO	INSTRUMENTO DE MEDICIÓN
					- Para cargar lista con veinte productos $T_{\text{medio}} \leq 1.3s$ es Satisfactorio		
					- Para crear venta con dos productos $T_{\text{medio}} \leq 5s$ es Satisfactorio		
					- Para cargar lista con veinte ventas $T_{\text{medio}} \leq 1.6s$ es Satisfactorio		
					- Para cargar reporte a partir de cien ventas $T_{\text{medio}} \leq 1.3s$ es Satisfactorio		
				Tiempo máximo de rasterizado	Con base en la NTP ISO/IEC 9126-2 para calcular tiempo máximo de	Determinar el tiempo máximo que	Firestore Performance Monitoring

VARIABLE	DEFINICIÓN	DIMENSIONES	INDICADORES	MÉTRICAS	FÓRMULA	PROPÓSITO	INSTRUMENTO DE MEDICIÓN
				de fotogramas	rasterizado de fotogramas debe aplicar la siguiente fórmula: Tmax = MAX(Tmax(i)) <i>Donde:</i> Tmax(i): tiempo máximo de rasterizado de fotogramas para la observación i. <i>Interpretación:</i> Tmax < 16ms es Satisfactorio	toma rasterizar los fotogramas.	
				Tiempo máximo de construcción de fotogramas	Con base en la NTP ISO/IEC 9126-2 para calcular el tiempo máximo de construcción de fotogramas se	Determinar el tiempo máximo que toma construir los fotogramas	Firestore Performance Monitoring

VARIABLE	DEFINICIÓN	DIMENSIONES	INDICADORES	MÉTRICAS	FÓRMULA	PROPÓSITO	INSTRUMENTO DE MEDICIÓN
					<p>debe aplicar la siguiente fórmula:</p> $T_{max} = \text{MAX}(T_{max}(i))$ <p><i>Donde:</i></p> <p>$T_{max}(i)$: tiempo máximo de construcción de fotogramas para la observación i.</p> <p><i>Interpretación:</i></p> <p>$T_{max} < 16\text{ms}$ es Satisfactorio</p> <p>Con base en la NTP ISO/IEC 9126-2 para calcular la cantidad promedio de fotogramas por segundo se debe aplicar la</p>		<p>Firestore Performance Monitoring</p>

VARIABLE	DEFINICIÓN	DIMENSIONES	INDICADORES	MÉTRICAS	FÓRMULA	PROPÓSITO	INSTRUMENTO DE MEDICIÓN
					siguiente fórmula: $T_{max} = \text{MAX}(T_{max}(i))$		
					<i>Donde:</i> Tmax(i): tiempo máximo de construcción de fotogramas para la observación i.		
					<i>Interpretación:</i> Tmax < 16ms es Satisfactorio		
Utilización de recursos				Utilización máxima de memoria RAM	Con base en la NTP ISO/IEC 9126-2 para calcular la utilización máxima de memoria RAM se debe aplicar la	Determinar el máximo uso de memoria RAM al realizar las tareas.	Firestore Performance Monitoring

VARIABLE	DEFINICIÓN	DIMENSIONES	INDICADORES	MÉTRICAS	FÓRMULA	PROPÓSITO	INSTRUMENTO DE MEDICIÓN
					siguiente fórmula: Mmáx = Cantidad máxima de memoria RAM utilizada al realizar las observaciones		
					<i>Interpretación:</i> Max ≤ 400 MB es Satisfactorio		
			Conformidad de Eficiencia	Conformidad de la Eficiencia	Con base en la NTP ISO/IEC 9126-2 para medir la conformidad de eficiencia se debe calcular de la siguiente manera:	Determinar el porcentaje de métricas que satisficieron sus grados de conformidad	Tabla de Registros
					$X = A / B * 100$		

VARIABLE	DEFINICIÓN	DIMENSIONES	INDICADORES	MÉTRICAS	FÓRMULA	PROPÓSITO	INSTRUMENTO DE MEDICIÓN
----------	------------	-------------	-------------	----------	---------	-----------	-------------------------

Donde:

A: número de métricas que satisficieron sus grados de conformidad

B: número total de métricas consideradas para medición.

Interpretación:

$X \geq 93.8\%$ es Satisfactorio

Fuente: Rabanal y Zegarra (2019)

Operacionalización de variable independiente

VARIABLE	DEFINICIÓN	DIMENSIONES	INDICADORES	PROPÓSITO	INSTRUMENTOS DE MEDICIÓN
Procesamiento de la información del área de SSOMA en operaciones mineras.	Forma en la que las personas prestan atención a los sucesos del medio, codifican la información relacionándola con datos existentes para luego almacenarla como una nueva información y en un futuro poder recuperarla (Shuell, 1986)	Proceso de campo	Evaluación de la actividad más demandante	¿Cuál es la actividad de campo más demandante por el personal de SSOMA en operaciones mineras?	Cuestionario en Google Forms
		Cobertura de red	Porcentaje de la cobertura de red	¿Cuenta con cobertura de red en todo momento?	Cuestionario en Google Forms
			Porcentaje de las zonas con menor cobertura	¿Cuáles son las zonas en donde la cobertura de red es más débil?	
		Aplicaciones móviles	Porcentaje de aplicaciones utilizadas.	¿Se tienen aplicaciones móviles para las actividades de campo?	Cuestionario en Google Forms

Anexo N° 5. Encuesta para identificar el problema

**INSTRUMENTO PARA RECOLECCIÓN Y
ANÁLISIS DE DATOS**

ALCANCE

Este documento muestra el modelo de la encuesta utilizada para recopilar información que ayude a identificar el uso de aplicaciones móviles y el proceso que realizan con mayor demanda los trabajadores del área de SSOMA en operaciones mineras, el cuestionario fue elaborado con la herramienta Google Forms, el cual fue compartido a diferentes personas que laboran directamente dentro de una empresa minera o en empresas terceras que brindan servicios en minería.

A su vez, este documento valida que la herramienta de Google Forms puede obtener información estadística por medio gráficos, los cuales se muestran de acuerdo a la información respondida en sus formularios, los tipos de gráficos que muestra esta herramienta son fáciles de interpretar permitiendo realizar deducciones directas a partir de los datos obtenidos, de esta manera se facilita su descripción mediante el método descriptivo.

Finalmente, se muestra una ficha de validación por parte de expertos en desarrollo y calidad de software, los cuales validaron como un instrumento apto para recopilar, analizar e interpretar la información.

1. ENCUESTA

Este formulario ha sido planteado con preguntas directas, las cuales están orientadas a recopilar la siguiente información:

- Identificar la actividad de campo que se realiza con mayor frecuencia en las operaciones mineras por parte del personal de SSOMA.
- Identificar si la cobertura de red es buena en cualquier zona de las operaciones mineras.
- Identificar si el personal de SSOMA cuenta con aplicaciones de tecnología móvil para desarrollar sus actividades.

Datos Personales

*Obligatorio

Dirección de correo electrónico *

abazan@electrodata.com.pe

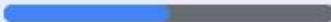
Indicar sus Apellidos y Nombres *

Anderson Bazan

Indicar la Operación Minera o Empresa en la que se encuentra laborando *

Bvn

[Siguiente](#)

 Página 1 de 2

ENCUESTA PARA IDENTIFICAR EL USO DE TECNOLOGÍAS MÓVILES EN OPERACIONES MINERAS

Según las actividades que realiza en el área, podría indicar cual o cuales son las que implica tener que salir a campo seguidamente.

Tu respuesta

¿Cuenta con cobertura de red en todo momento cuando se encuentra en la operación minera?

- Sí
- No

Marque los lugares de la operación minera donde la cobertura de red es menor

- Oficinas
- Tajo
- Planta
- Socavón
- Laboratorio
- Almacenes
- Balanza
- Otros: _____

¿Su área o empresa cuenta con herramientas de tecnología móvil adecuadas que ayuden a realizar sus actividades de campo dentro de la operación minera con normalidad?

- Sí
- No

¿Cuenta con algún dispositivo móvil del tipo Smartphone?

Sí

No

Señale el sistema operativo del dispositivo móvil con el que cuenta

Android

Ios (Iphone)

Otros: _____

¿Usted cree que su dispositivo móvil sería de gran ayuda para realizar sus actividades de campo dentro de la operación minera?

Sí

No

Tal vez

¿Usted utilizaría una aplicación móvil que le ayude agilizar sus actividades de campo dentro de la operación minera?

Sí

No

Tal vez

[Atrás](#)

[Enviar](#)

 Página 2 de 2

2. GOOGLE FORMS

Esta herramienta permite crear y compartir formularios que son respondidos por distintas personas, así mismo permite visualizar estadísticamente las respuestas indicadas en los formularios permitiendo analizar e interpretar los indicadores de una manera más sencilla.

Tipos de gráficos

Los tipos de gráficos que presenta la herramienta Google Forms son los siguientes:

- Gráfico de barras, según Castillero (2017), este tipo de gráficos son utilizados generalmente para mostrar la frecuencia de diferentes condiciones o variables cuantitativas, siendo estos de fácil interpretación.

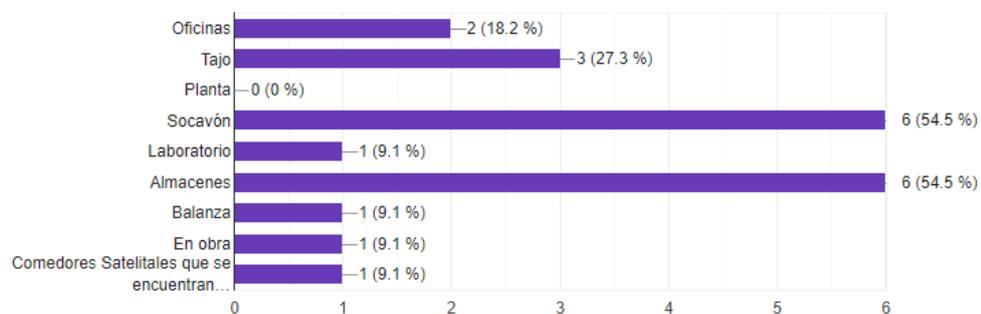


Figura 6: Gráfico de barras

- Gráfico circular, según Castillero (2017), este tipo de gráficos son utilizados habitualmente cuando se desea mostrar la proporción de casos dentro del total, utilizandolos para ser interpretados mediante datos porcentuales.

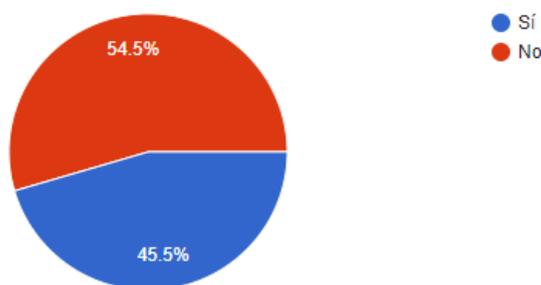


Figura 7: Gráfico circular

3. VALIDACIÓN DE EXPERTOS



UNIVERSIDAD
PRIVADA DEL NORTE

FICHA PARA VALIDACION DEL INSTRUMENTO

I. REFERENCIA

- 1.1. Experto: Jefferson Junior Grados Diope
- 1.2. Especialidad: Ingeniería de Sistemas
- 1.3. Cargo actual: Analista de Gestión de la Información
- 1.4. Grado académico: Titulado
- 1.5. Institución: Financiera Credinka
- 1.6. Tipo de instrumento: Encuesta en Google documents
- 1.7. Lugar y fecha: Lima 23/06/2020

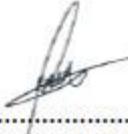
II. TABLA DE VALORACION POR EVIDENCIAS

N°	EVIDENCIAS	VALORACION					
		5	4	3	2	1	0
1	Pertinencia de indicadores		x				
2	Formulado con lenguaje apropiado		x				
3	Adecuado para los sujetos en estudio	x					
4	Facilita la prueba de hipótesis		x				
5	Suficiencia para medir la variable		x				
6	Facilita la interpretación del instrumento	x					
7	Acorde al avance de la ciencia y tecnología		x				
8	Expresado en hechos perceptibles		x				
9	Tiene secuencia lógica	x					
10	Basado en aspectos teóricos		x				
Total		15	28				

Coeficiente de valoración porcentual: $c = 86\%$

III. OBSERVACIONES Y/O RECOMENDACIONES

Ser más específico al querer obtener respuesta en la primera pregunta de la encuesta, colocar un conjunto actividades definidas por la empresa y que el usuario escoja, ya que podría haber N cantidades de actividades y sería bueno definirlos.



.....
Firma y sello del Experto



FICHA PARA VALIDACIÓN DEL INSTRUMENTO

I. REFERENCIA

- 1.1. Experto: Abigail Miriam Castillo Castro
- 1.2. Especialidad: Analista del aseguramiento de Calidad
- 1.3. Cargo actual: Analista senior de Calidad de Software
- 1.4. Grado académico: Ingeniero de sistemas e Informática
- 1.5. Institución: Financiera Credinka
- 1.6. Tipo de instrumento: Encuesta, Google Documents
- 1.7. Lugar y fecha: Lima, 26 de junio del 2020

¡TABLA DE VALORACIÓN POR EVIDENCIAS

N°	EVIDENCIAS	VALORACIÓN					
		5	4	3	2	1	0
1	Pertinencia de indicadores		x				
2	Formulado con lenguaje apropiado		x				
3	Adecuado para los sujetos en estudio	x					
4	Facilita la prueba de hipótesis	x					
5	Suficiencia para medir la variable		x				
6	Facilita la interpretación del instrumento		x				
7	Acorde al avance de la ciencia y tecnología	x					
8	Expresado en hechos perceptibles		x				
9	Tiene secuencia lógica		x				
10	Basado en aspectos teóricos	x					
	Total	20	24				

Coefficiente de valoración porcentual: $c = 88\%$

II. OBSERVACIONES Y/O RECOMENDACIONES

Se observa que la mayoría de preguntas sólo es para marcar, sin embargo, se puede optar para que las preguntas sean en su 100% opción múltiple o específicas para que al usuario no le demande mucho tiempo en responderlo.



.....
Firma y sello del Experto

FICHA PARA VALIDACION DEL INSTRUMENTO

I. REFERENCIA

- 1.1. Experto: Julio César Calvo Torres
- 1.2. Especialidad: Desarrollo de Software
- 1.3. Cargo actual: Analista de aplicaciones
- 1.4. Grado académico: Ingeniero de Sistemas e Informática
- 1.5. Institución: Electrodata
- 1.6. Tipo de instrumento: Encuesta
- 1.7. Lugar y fecha: Lima, 23/06/2020

II. TABLA DE VALORACION POR EVIDENCIAS

Nº	EVIDENCIAS	VALORACION					
		5	4	3	2	1	0
1	Pertinencia de indicadores		X				
2	Formulado con lenguaje apropiado	X					
3	Adecuado para los sujetos en estudio	X					
4	Facilita la prueba de hipótesis		X				
5	Suficiencia para medir la variable		X				
6	Facilita la interpretación del instrumento	X					
7	Acorde al avance de la ciencia y tecnología	X					
8	Expresado en hechos perceptibles		X				
9	Tiene secuencia lógica	X					
10	Basado en aspectos teóricos		X				
	Total	25	20				

Coefficiente de valoración porcentual: $c = \frac{20}{25} = 80\%$

III. OBSERVACIONES Y/O RECOMENDACIONES

.....

.....

.....

.....

.....
Firma y sello del Experto

**Anexo N° 6. Firebase Performance Monitoring para recolectar y analizar
información**

**FIREBASE PERFORMANCE
MONITORING INSTRUMENTO PARA
RECOLECCIÓN Y ANÁLISIS DE DATOS**

ALCANCE

Este documento tiene como finalidad validar que mediante la herramienta Firebase Performance Monitoring se puede obtener información estadística por medio de gráficos, los cuales se muestran de acuerdo a la información recolectada de los diferentes dispositivos móviles donde haya sido ejecutada y utilizada la aplicación.

Los gráficos que muestra esta herramienta son de diferentes tipos, permitiendo realizar deducciones directas a partir de los datos obtenidos, de esta manera se facilita su descripción mediante el método descriptivo.

Asu vez este documento muestra una ficha de validación por parte de expertos en desarrollo y calidad de software, los cuales validaron como un instrumento apto para recolectar, analizar e interpretar la información.

1. FIREBASE PERFORMANCE MONITORING

Según la documentación de Google esta herramienta permite recopilar datos de rendimiento de las aplicaciones móviles para luego poder ser revisados y analizados; así mismo, ayuda a comprender dónde se puede mejorar el rendimiento del aplicativo de manera que la información se pueda usar para solucionar problemas de rendimiento.

Permite medir automáticamente el tiempo de inicio de las aplicaciones, el tiempo de demora las solicitudes de red HTTP/S y la cantidad de memoria asignada y utilizada en el dispositivo móvil, funciones claves para medir el rendimiento de una aplicación.

Los datos son recopilados por cada dispositivo en el cual se ejecuta la aplicación:

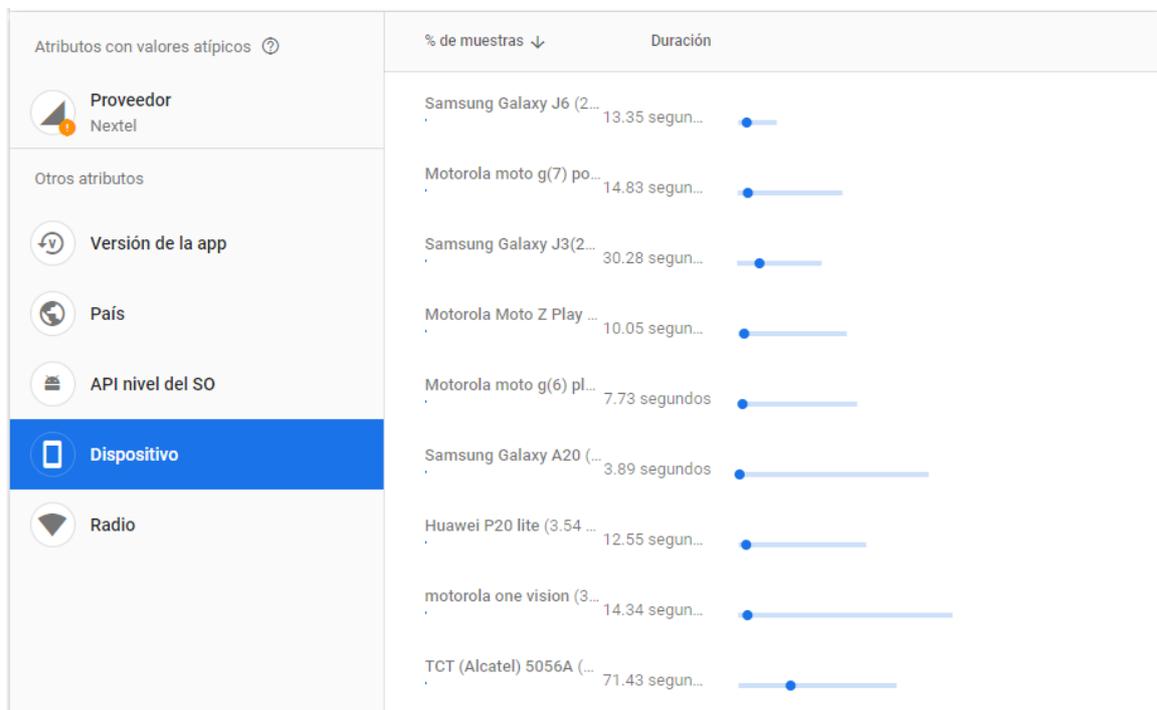


Figura 8: Cuadro de recolección de tiempos de duración

Atributos con valores atípicos ?	% de muestras ↓	Procesamiento lento
Dispositivo Huawei P20 lite	Motorola moto g(7) po...	0 %
Otros atributos	Samsung Galaxy J6 (2...	0 %
➕ Versión de la app	Samsung Galaxy A20 (...)	0 %
🌐 País	Motorola moto g(6) pl...	0 %
📱 API nivel del SO	motorola one vision (7...	0 %
📶 Radio	🚩 Huawei P20 lite (6.9 %)	8.93 %
🏠 Proveedor	Motorola moto g(6) (6...	0 %

Figura 9: Cuadro de recolección de procesamiento lento

Atributos con valores atípicos ?	% de muestras ↓	Tiempo de respuesta
➕ Versión de la app	Samsung Galaxy J3(2...	461 ms
🌐 País	Samsung Galaxy J6 (1...	569 ms
📱 API nivel del SO	Motorola Moto Z Play ...	787 ms
Dispositivo	Huawei P20 lite (11.18...	438 ms
📶 Radio	Motorola moto g(7) po...	489 ms
🏠 Proveedor	motorola one vision (4...	416 ms
📄 Tipo MIME	Motorola moto g(6) (4...	279 ms
	TCT (Alcatel) 5056A (...)	367 ms
	Motorola moto g(6) pl...	1.18 segundos
	Samsung Galaxy A20 (...)	330 ms

Figura 10: Cuadro de recolección de tiempos de respuesta

2. DATOS ESTADÍSTICOS

Esta herramienta permite visualizar estadísticamente los datos recolectados, permitiendo analizar e interpretar los indicadores de una manera más sencilla

Los tipos de gráficos más representativos que presenta la herramienta Firebase Performance Monitoring son los siguientes:

- Gráfico de anillo, según documentación de Google, este tipo de gráficos son utilizados habitualmente para comparar las proporciones de las partes de una o más series de datos con el todo.



Figura 11: Gráfico de anillo

- Gráfico de área, según documentación de Google, este tipo de gráficos son utilizados generalmente para mostrar tendencias en los datos en el transcurso del tiempo, presentando el desarrollo de valores cuantitativos durante un intervalo de tiempo.



Figura 12: Gráfico de área de tiempo

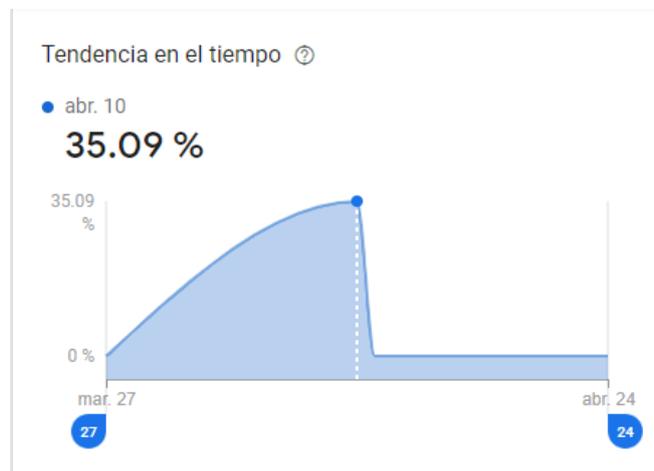


Figura 13: Gráfico de área de porcentaje

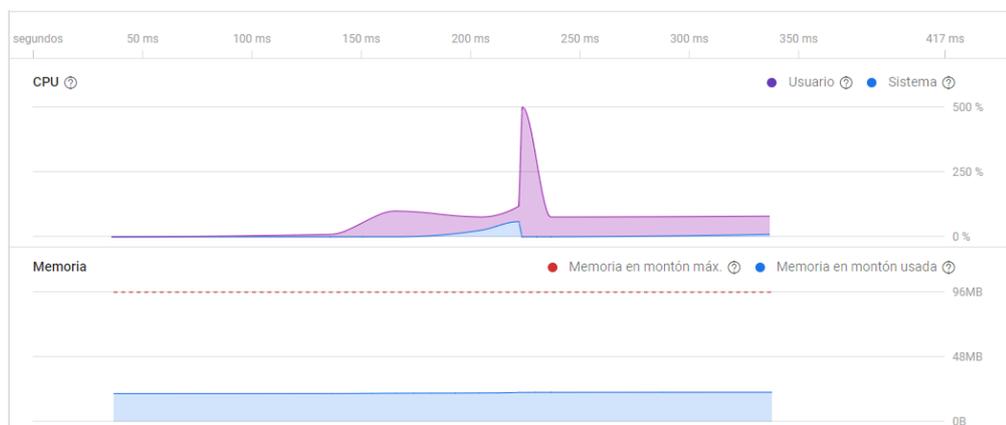


Figura 14: Gráfico de área de memoria

3. VALIDACIÓN DE EXPERTOS



FICHA PARA VALIDACIÓN DEL INSTRUMENTO

I. REFERENCIA

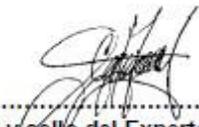
- 1.1. Experto: Abigail Miriam Castillo Castro
- 1.2. Especialidad: Analista del aseguramiento de Calidad
- 1.3. Cargo actual: Analista senior de Calidad de Software
- 1.4. Grado académico: Ingeniero de sistemas e Informática
- 1.5. Institución: Financiera Credinka
- 1.6. Tipo de instrumento: Firebase Performance Monitoring
- 1.7. Lugar y fecha: Lima, 26 de junio del 2020

II. TABLA DE VALORACIÓN POR EVIDENCIAS

N°	EVIDENCIAS	VALORACIÓN					
		5	4	3	2	1	0
1	Muestra indicadores	X					
2	Recolecta información	X					
3	Adecuado para analizar información	X					
4	Suficiencia para medir la variable	X					
5	Acorde al avance de la ciencia y tecnología	X					
6	Facilita la interpretación del rendimiento de las aplicaciones	X					
	Total	30					

Coefficiente de valoración porcentual: $c = 100\%$

III. OBSERVACIONES Y/O RECOMENDACIONES



 Firma y sello del Experto

FICHA PARA VALIDACION DEL INSTRUMENTO

I. REFERENCIA

- 1.1. Experto: Elmer Malca Palomino
- 1.2. Especialidad: Desarrollo de Software
- 1.3. Cargo actual: Analista de aplicaciones
- 1.4. Grado académico: Ingeniero de sistemas
- 1.5. Institución: Grupo Electrodata
- 1.6. Tipo de instrumento: Herramienta Firebase Performance Monitoring
- 1.7. Lugar y fecha: Lima, 26/06/2020

II. TABLA DE VALORACION POR EVIDENCIAS

N°	EVIDENCIAS	VALORACION					
		5	4	3	2	1	0
1	Muestra indicadores	X					
2	Recolecta información	X					
3	Adecuado para analizar información	X					
4	Suficiencia para medir la variable	X					
5	Acorde al avance de la ciencia y tecnología	X					
6	Facilita la interpretación del rendimiento de las aplicaciones	X					
	Total	30					

Coefficiente de valoración porcentual: $c = 100\%$

III. OBSERVACIONES Y/O RECOMENDACIONES

.....



Firma y sello del Experto

FICHA PARA VALIDACION DEL INSTRUMENTO

I. REFERENCIA

- 1.1. Experto: Julio César Calvo Torres
- 1.2. Especialidad: Desarrollo de Software
- 1.3. Cargo actual: Analista de aplicaciones
- 1.4. Grado académico: Ingeniero de Sistemas e Informática
- 1.5. Institución: Electrodata
- 1.6. Tipo de instrumento: Firestore Performance Monitoring
- 1.7. Lugar y fecha: Lima, 23/06/2020

II. TABLA DE VALORACION POR EVIDENCIAS

N°	EVIDENCIAS	VALORACION					
		5	4	3	2	1	0
1	Muestra indicadores		X				
2	Recolecta información	X					
3	Adecuado para analizar información	X					
4	Suficiencia para medir la variable		X				
5	Acorde al avance de la ciencia y tecnología	X					
6	Facilita la interpretación del rendimiento de las aplicaciones	X					
	Total	20	8				

Coefficiente de valoración porcentual: $c = \underline{93.3\%}$

III. OBSERVACIONES Y/O RECOMENDACIONES

.....

.....

.....

.....

.....


.....
Firma y sello del Experto

Anexo N° 7. Metodología de desarrollo

METODOLOGÍA ICONIX

ANÁLISIS DEL REQUERIMIENTO

1. Requerimiento del software

El factor principal para la elaboración del proyecto viene a ser la necesidad de contar con un aplicativo que permita reportar actos y condiciones en las unidades mineras.

1.1. Perspectiva de producto

El aplicativo en adelante Yanapay, es un aplicativo único, independiente, desarrollado a medida, que cumplirá con los requerimientos recolectados por las personas involucradas al proceso de seguridad y salud ocupacional, no dependerá de ningún otro sistema que use la empresa.

1.2. Requerimientos funcionales

Yanapay es un aplicativo móvil de seguridad y salud ocupacional que debe cumplir funciones como:

- Registrar actos y condiciones observadas.
- Registrar evidencias encontradas en cada observación realizada.
- Notificar a las personas involucradas mediante un correo.
- Registrar las soluciones por cada observación reportada.

1.3. Interfaces de usuario

Las interfaces del usuario que se presentan en este proyecto son de fácil interacción para el usuario y cumplen con objetivos de usabilidad, además de tener un diseño coherente y claro, por ende, un mejor y más sencillo uso del aplicativo.

1.4. Interfaces con hardware

El aplicativo no cuenta con el uso de configuraciones externas más que las de la base de datos y servicio IIS que se pondrá en un servidor público.

1.5. Interfaces de software

Este aplicativo no requiere el uso de otros productos de software además del utilizado para configurar la base de datos.

1.6. Interfaces de comunicación

Este aplicativo no tiene comunicación con otro sistema o dispositivo.

1.7. Requerimientos de adecuación al entorno

El aplicativo solo necesitará de un servidor público para instalar la base de datos y el servicio IIS, los cuales deben estar siempre activos para el funcionamiento del aplicativo.

1.8. Características de los usuarios

El aplicativo estará construido para tener como usuarios a los supervisores y/o colaboradores del área de salud y seguridad ocupacional y medio ambiente de las operaciones mineras.

Para tener acceso al sistema se definen usuarios mediante un correo y una clave que será asignada por el administrador del aplicativo.

1.9. Supuestos y dependencias

Algunos factores que pueden afectar los requerimientos del sistema son:

- Agregar nuevas funcionalidades al sistema.
- Cambios en los requerimientos.
- Migrar la base de datos a otra.
- Migrar el aplicativo con un IDE diferente a Android Studio
- Utilización de otro lenguaje diferente a Java.

2. Modelo del dominio

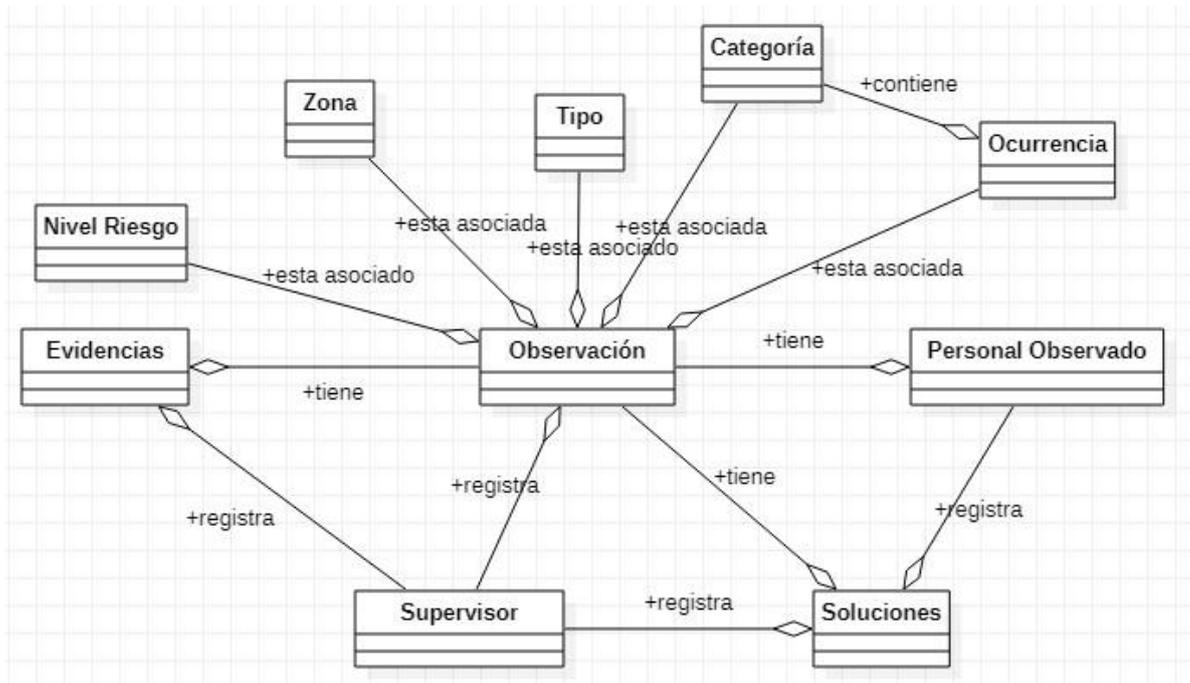


Figura 15: Modelo del dominio

2.1. Listado de posibles objetos o clases de dominio

- **Supervisor:** persona encargada de reportar observaciones encontradas en el área de trabajo.
- **Observación:** acto o condición observada en el área de trabajo.
- **Evidencias:** documentos o imágenes que se adjuntan a las observaciones con la finalidad de ser evidencia de lo reportado.
- **Soluciones:** documentos o imágenes que se adjuntan a las observaciones con la finalidad de ser evidencia de la solución realizada.
- **Personal observado:** personal responsable de la zona o área donde se identifica el acto o condición observada.
- **Nivel riesgo:** prioridad que se le da a las observaciones para su posterior solución.
- **Zonas:** lugar o área donde es reportada la observación.
- **Tipo:** está referido al tipo de observación, los cuales pueden ser SSO (seguridad y salud ocupacional) o MA (medio ambiente).
- **Categoría:** está referido a la categoría de la observación, las cuales pueden ser acto o condición.

- **Ocurrencia:** está referido a la clasificación de la observación según su tipo (SSO, MA).

3. Modelo de casos de uso

3.1. Actores

- **Supervisor**

Usuario que se encarga de registrar las observaciones identificadas en el área de trabajo, también será el encargado de dar seguimiento y cerrar las observaciones.

- **Personal observado**

Personas responsables de un área, tienen colaboradores a su cargo, los cuales son reportados en las observaciones, serán los encargados de registrar o adjuntar las soluciones a las observaciones donde figuran como personas observadas.

3.2. Diagrama de casos de uso

3.2.1. Actores

- **Supervisor**

Usuario que se encargará de reportar los actos o condiciones observadas.

- **Personal observado**

Usuario que se encargará de registrar las soluciones en las observaciones que ha sido observado.

3.2.2. Diagramas de caso de uso

- **Observación – Supervisor**

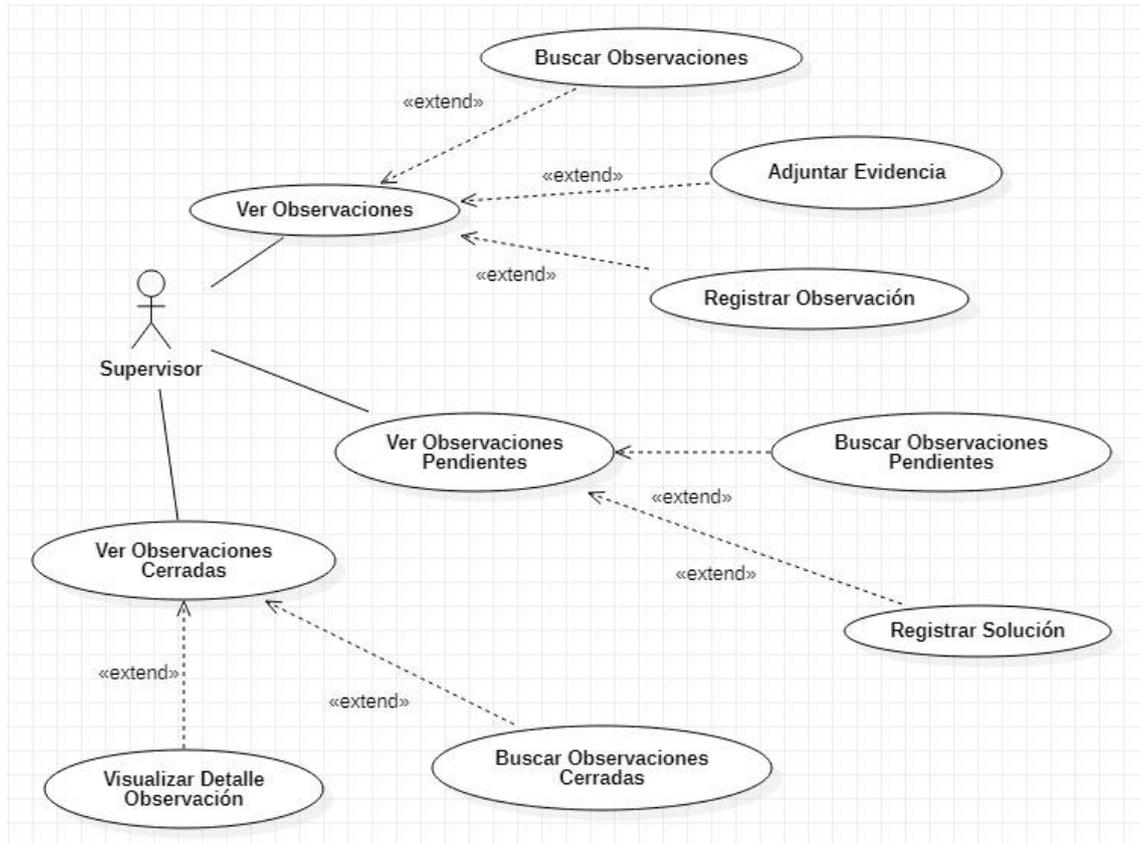


Figura 16: Diagrama de caso de uso supervisor

○ **Observación – Personal observado**

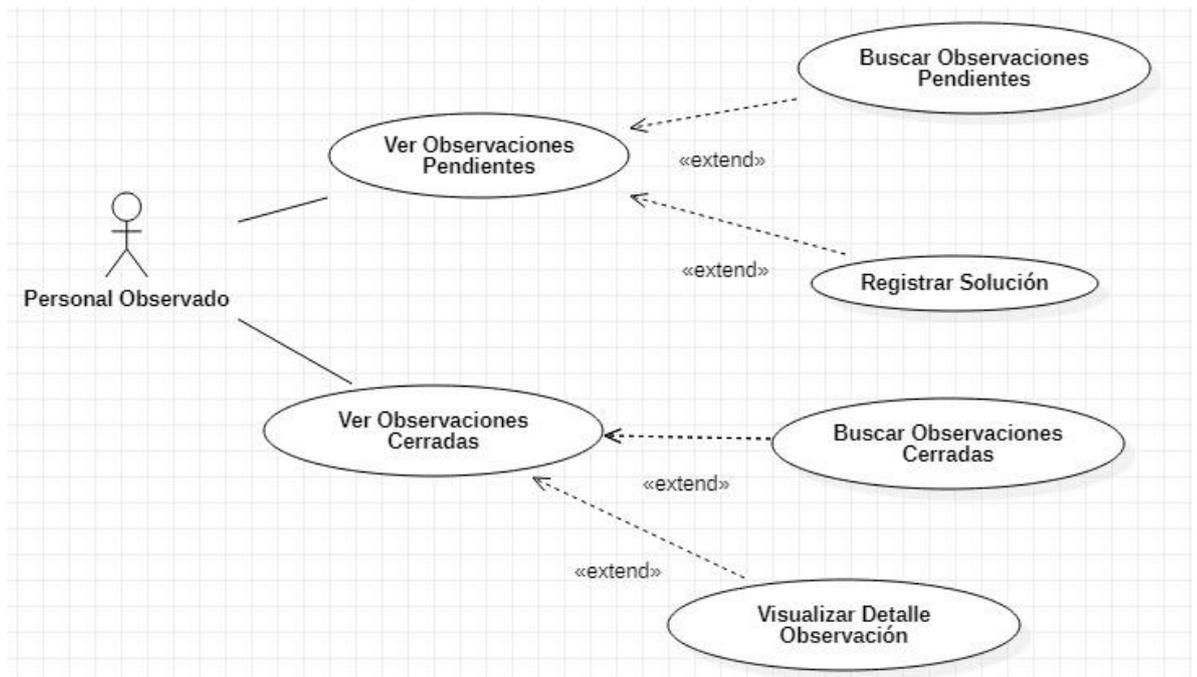


Figura 17: Diagrama de caso de uso personal observado

4. Casos de uso

4.1. Ver observaciones

a) Descripción

Este caso de uso permitirá al supervisor poder visualizar las observaciones registradas ordenándolas desde la más reciente.

b) Prototipo



Figura 18: Prototipo ver observaciones

- ✓ En el prototipo se muestra la manera de cómo se van a visualizar las observaciones, se cargan en la lista los actos o condiciones ordenándolas desde la más reciente, además se tiene la opción de crear una nueva observación o se podrá seleccionar alguno de ellos para registrar una evidencia.

4.2. Ver observaciones pendientes

a) Descripción

Este caso de uso permitirá al supervisor y personal observado poder visualizar las observaciones pendientes ordenándolas desde la más reciente.

b) Prototipo



Figura 19: Prototipo ver observaciones pendientes

- ✓ En el prototipo se muestra la manera de cómo se van a visualizar las observaciones pendientes, se cargan en la lista los actos o condiciones ordenándolas desde la más reciente, además se podrá seleccionar alguno de ellos para registrar su solución

4.3. Ver observaciones cerradas

a) Descripción

Este caso de uso permitirá al supervisor y personal observado poder visualizar las observaciones cerradas ordenándolas desde la más reciente.

b) Prototipo

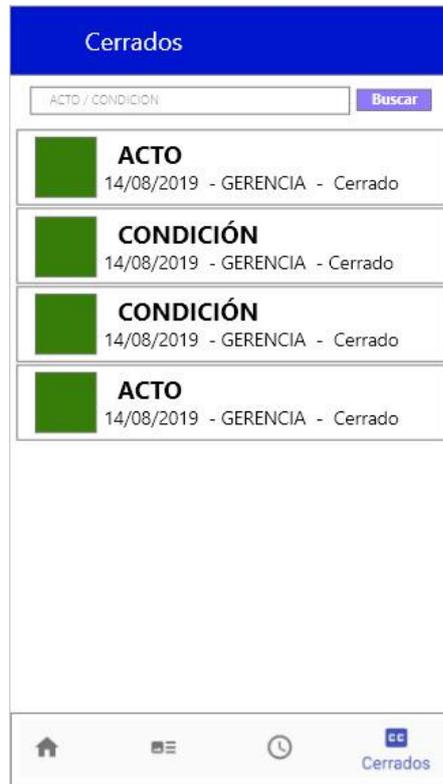


Figura 20: Prototipo ver observaciones cerradas

- ✓ En el prototipo se muestra la manera de cómo se van a visualizar las observaciones cerradas, se cargan en la lista los actos o condiciones ordenándolas desde la más reciente, además se podrá seleccionar alguno de ellos para visualizar su detalle.

4.4. Buscar observaciones

a. Descripción

Este caso de uso permitirá al supervisor poder buscar las observaciones registradas.

b. Prototipo



Figura 21: Prototipo buscar observaciones

- ✓ En el prototipo se muestra la manera de cómo se va a realizar la búsqueda de observaciones, se digita un código de acto o condición, luego se selecciona en el botón “Buscar”, se cargan en la lista los actos o condiciones que cumplan con la descripción del filtro, además se tiene la opción de crear una nueva observación o se podrá seleccionar alguno de ellos para registrar una evidencia.

4.5. Buscar observaciones pendientes

a. Descripción

Este caso de uso permitirá al supervisor y personal observado poder buscar las observaciones pendientes por solucionar.

b. Prototipo



Figura 22: Prototipo buscar observaciones pendientes

- ✓ En el prototipo se muestra la manera de cómo se va a realizar la búsqueda de observaciones pendientes, se digita un código de acto o condición, luego se selecciona en el botón “Buscar”, se cargan en la lista los actos o condiciones que cumplan con la descripción del filtro, además se podrá seleccionar alguno de ellos para registrar su solución.

4.6. Buscar observaciones cerradas

a. Descripción

Este caso de uso permitirá al supervisor y personal observado poder buscar las observaciones cerradas para visualizar su detalle.

b. Prototipo

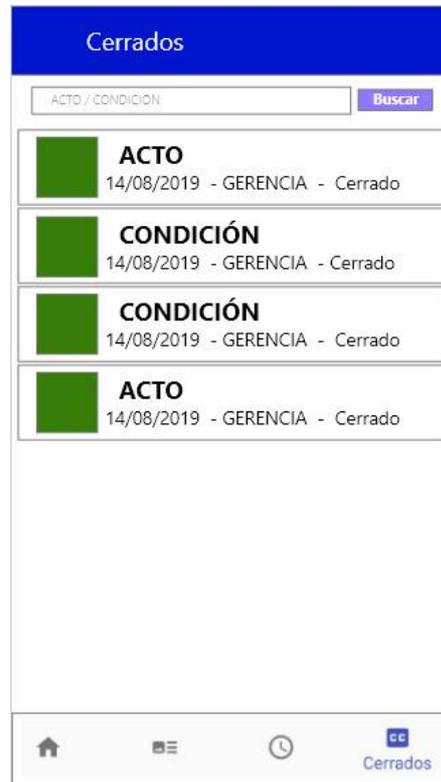


Figura 23: Prototipo buscar observaciones cerradas

- ✓ En el prototipo se muestra la manera de cómo se va a realizar la búsqueda de observaciones cerradas, se digita un código de acto o condición, luego se selecciona en el botón “Buscar”, se cargan en la lista los actos o condiciones que cumplan con la descripción del filtro, además se podrá seleccionar alguno de ellos para visualizar su detalle.

4.7. Registrar observación

a. Descripción

Este caso de uso permitirá al supervisor poder registrar una nueva observación.

b. Prototipo



Registrar Acto / Condición

Acto **Condición**

Categoría

Zona

Observación

Ocurrencia

Nivel Riesgo

Acción Correctiva

N° Días para cierre de Observación

Personal Observado

Comentarios

GUARDAR

Figura 24: Prototipo registrar observación

- ✓ En el prototipo se muestra la manera de cómo se va a realizar el registro de una observación, se ingresan los datos y se selecciona el botón “Guardar”, adicional a ello por medio de este caso de uso se podrá editar la observación antes de ser reportada.

4.8. Adjuntar evidencia

a. Descripción

Este caso de uso permitirá al supervisor poder registrar las evidencias de una observación registrada, adicional a ello se podrá visualizar el detalle de la observación.

b. Prototipo



Acto / Condición

Categoría	Nivel Riesgo
Zona	Fecha Cierre

Personal Observado ▼

#####

Observación ▼

Acción Correctiva ▼

Evidencias - Reportar

#####



Figura 25: Prototipo adjuntar evidencia

- ✓ En el prototipo se muestra la manera de cómo se va a realizar el registro de las evidencias, se selecciona en el botón de “Adjuntar” y se escogen los documentos o imágenes de evidencia.

4.9. Registrar solución

a. Descripción

Este caso de uso permitirá al supervisor y personal observado poder registrar la solución de una observación reportada, adicional a ello se podrá visualizar el detalle de la observación.

b. Prototipo

Acto / Condición

Categoría	Nivel Riesgo
Zona	Fecha Cierre

Observación ▼

Acción Correctiva ▼

Solución

Evidencia - Solución

GUARDAR

Figura 26: Prototipo registrar solución

- ✓ En el prototipo se muestra la manera de cómo se va a realizar el registro de la solución, se ingresa el detalle de la solución y se selecciona el botón “Guardar”, adicional a ello se muestra la manera de cómo se va a realizar el registro de las evidencias de solución, se selecciona en el botón de “Adjuntar” y se escogen los documentos o imágenes de evidencia solución.

4.10. Visualizar detalle observación

a. Descripción

Este caso de uso permitirá al supervisor y personal observado poder visualizar el detalle de una observación reportada cerrada.

b. Prototipo

Figura 27: Prototipo visualizar detalle observación

- ✓ En el prototipo se muestra la manera de cómo se va a visualizar el detalle de una observación reportada cerrada, se selecciona la evidencia reportada o solución para descargar el archivo adjunto.

5. Diagrama de paquetes

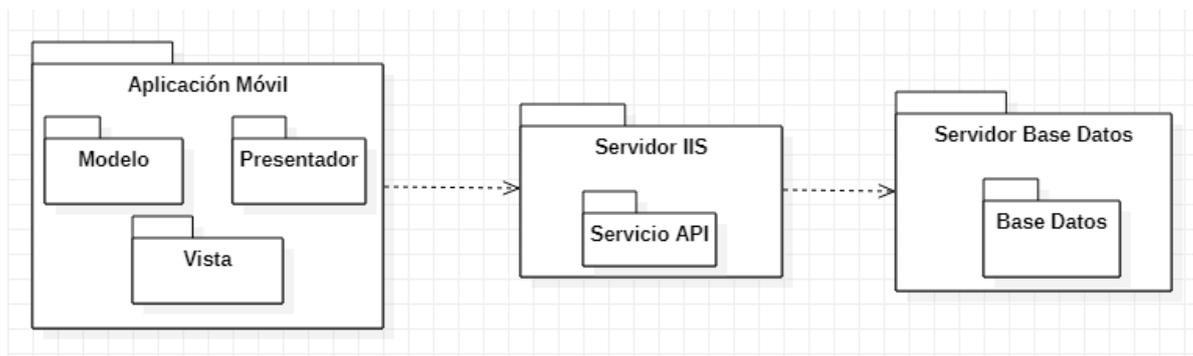


Figura 28: Diagrama de paquetes

6. Fichas de caso de uso

6.1. Caso de uso ver observaciones

Caso de uso:

Ver observaciones

Autores:

Anderson Kevin Bazán Urteaga

Oscar Geny Saavedra Guarniz

Propósito:

Mostrar una lista de las observaciones registradas.

Actores:

Supervisor

Precondición:

- Usuario debe autenticarse.
- Usuario debe ubicarse en la opción “observaciones” de la vista menú.
- Debe existir información de observaciones registradas que no hayan sido reportadas.

Flujo básico

- El aplicativo muestra una casilla en blanco “acto / condición” ubicado en la parte superior.
- El aplicativo muestra un botón circular con el símbolo de adición ubicado en la parte inferior derecha.
- El aplicativo muestra una lista de las observaciones registradas, considerándolas desde la fecha más reciente.

Flujo(s) alternativo(s):

- **Buscar observación:** si el usuario desea buscar una observación, este debe ubicarse en la casilla en blanco “acto / condición” y se iniciará el caso de uso “Buscar Observaciones”.
- **Registrar observación:** si el usuario desea registrar una observación, este debe presionar el botón circular con el símbolo de adición ubicado en la parte inferior derecha de la vista y se iniciará el caso de uso “Registrar Observación”.
- **Adjuntar evidencia:** si el usuario desea registrar evidencias, este debe seleccionar una observación de la lista y se iniciará el caso de uso “Adjuntar Evidencia”.

- **Reportar:** si el usuario desea reportar una observación, el usuario debe presionar en el botón “reportar” ubicado al lado derecho de cada observación de la lista y el aplicativo mostrará un mensaje de confirmación para que el usuario confirme o decline la acción.

Postcondición:

La información se encuentra registrada correctamente en la base de datos.

6.2. Caso de uso ver observaciones pendientes

Caso de uso:

Ver observaciones pendientes

Autores:

Anderson Kevin Bazán Urteaga

Oscar Geny Saavedra Guarniz

Propósito:

Mostrar una lista de las observaciones que han sido reportadas.

Actores:

Supervisor

Personal Observado

Precondición:

- Usuario debe autenticarse.
- Usuario debe ubicarse en la opción “pendientes” de la vista menú.
- Debe existir información de observaciones reportadas.

Flujo básico

- El aplicativo muestra una casilla en blanco “acto / condición” ubicado en la parte superior.
- El aplicativo muestra una lista de las observaciones reportadas, considerándolas desde la fecha más reciente.

Flujo(s) alternativo(s):

- **Buscar observación pendiente:** si el usuario desea buscar una observación pendiente, este debe ubicarse en la casilla en blanco “acto / condición” y se iniciará el caso de uso “Buscar Observaciones Pendientes”.

- **Registrar solución:** si se desea registrar la solución adjuntando sus evidencias, el usuario debe seleccionar una observación de la lista y se iniciará el caso de uso “Registrar Solución”.
- **Solucionar:** si el usuario personal observado o supervisor desea solucionar la observación, este debe presionar en el botón “solucionar” ubicado al lado derecho de cada observación de la lista y el aplicativo mostrará un mensaje de confirmación para que el usuario confirme o decline la acción.
- **Cerrar:** si el usuario supervisor desea dar por cerrada la observación, el usuario debe presionar en el botón “cerrar” ubicado al lado derecho de cada observación de la lista y el aplicativo mostrará un mensaje de confirmación para que el usuario confirme o decline la acción.

Postcondición:

La información se encuentra registrada correctamente en la base de datos.

6.3. Caso de uso ver observaciones cerradas

Caso de uso:

Ver observaciones cerradas

Autores:

Anderson Kevin Bazán Urteaga

Oscar Geny Saavedra Guarniz

Propósito:

Mostrar una lista de las observaciones que han sido dadas por cerradas.

Actores:

Supervisor

Personal Observado

Precondición:

- Usuario debe autenticarse.
- Usuario debe ubicarse en la opción “cerrados” de la vista menú.
- Debe existir información de observaciones cerradas.

Flujo básico

- El aplicativo muestra una casilla en blanco “acto / condición” ubicado en la parte superior.

- El aplicativo muestra una lista de las observaciones cerradas, considerándolas desde la fecha más reciente.

Flujo(s) alternativo(s):

- **Buscar observación cerradas:** si el usuario desea buscar una observación cerrada, este debe ubicarse en la casilla en blanco “acto / condición” y se iniciará el caso de uso “Buscar Observaciones Cerradas”.
- **Visualizar detalle:** si el usuario desea visualizar el detalle de la observación, este debe seleccionar una observación de la lista y se iniciará el caso de uso “Visualizar Detalle Observación”.

Postcondición:

La información se encuentra registrada correctamente en la base de datos.

6.4. Caso de uso buscar observaciones

Caso de uso:

Buscar observaciones

Autores:

Anderson Kevin Bazán Urteaga

Oscar Geny Saavedra Guarniz

Propósito:

Mostrar una lista de las observaciones registradas según un criterio de búsqueda que luego serán reportadas.

Actores:

Supervisor

Precondición:

- Usuario debe autenticarse.
- Usuario debe ubicarse en la opción “observaciones” de la vista menú.
- Debe existir información de observaciones registradas que no hayan sido reportadas.

Flujo básico

- El aplicativo muestra una casilla en blanco “acto / condición” ubicado en la parte superior.
- El aplicativo muestra un botón circular con el símbolo de adición ubicado en la parte inferior derecha.

- El usuario ingresa un código de observación y hace click en el botón “buscar” del teclado, el sistema muestra en la lista los datos de las observaciones que coincidan con el parámetro ingresado.

Flujo(s) alternativo(s):

- **Registrar observación:** si el usuario desea registrar una observación, este debe presionar el botón circular con el símbolo de adición ubicado en la parte inferior derecha de la vista y se iniciará el caso de uso “Registrar Observación”.
- **Adjuntar evidencia:** si el usuario desea registrar evidencias, este debe seleccionar una observación de la lista y se iniciará el caso de uso “Adjuntar Evidencia”.
- **Reportar:** si el usuario desea reportar una observación, el usuario debe presionar en el botón “reportar” ubicado al lado derecho de cada observación de la lista y el aplicativo mostrará un mensaje de confirmación para que el usuario confirme o decline la acción.

Postcondición:

La información se encuentra registrada correctamente en la base de datos.

6.5. Caso de uso buscar observaciones pendientes

Caso de uso:

Buscar observaciones pendientes

Autores:

Anderson Kevin Bazán Urteaga

Oscar Geny Saavedra Guarniz

Propósito:

Mostrar una lista de las observaciones que han sido reportadas según un criterio de búsqueda.

Actores:

Supervisor

Personal Observado

Precondición:

- Usuario debe autenticarse.

- Usuario debe ubicarse en la opción “pendientes” de la vista menú.
- Debe existir información de observaciones reportadas.

Flujo básico

- El aplicativo muestra una casilla en blanco “acto / condición” ubicado en la parte superior.
- El usuario ingresa un código de observación y hace click en el botón “buscar” del teclado, el sistema muestra en la lista los datos de las observaciones que coincidan con el parámetro ingresado.

Flujo(s) alternativo(s):

- **Registrar solución:** si se desea registrar la solución adjuntando sus evidencias, el usuario debe seleccionar una observación de la lista y se iniciará el caso de uso “Registrar Solución”.
- **Solucionar:** si el usuario personal observado o supervisor desea solucionar la observación, este debe presionar en el botón “solucionar” ubicado al lado derecho de cada observación de la lista y el aplicativo mostrará un mensaje de confirmación para que el usuario confirme o decline la acción.
- **Cerrar:** si el usuario supervisor desea dar por cerrada la observación, el usuario debe presionar en el botón “cerrar” ubicado al lado derecho de cada observación de la lista y el aplicativo mostrará un mensaje de confirmación para que el usuario confirme o decline la acción.

Postcondición:

La información se encuentra registrada correctamente en la base de datos.

6.6. Caso de uso buscar observaciones cerradas

Caso de uso:

Buscar observaciones cerradas

Autores:

Anderson Kevin Bazán Urteaga

Oscar Geny Saavedra Guarniz

Propósito:

Mostrar una lista de las observaciones que han sido dadas por cerradas según un criterio de búsqueda.

Actores:

Supervisor

Personal Observado

Precondición:

- Usuario debe autenticarse.
- Usuario debe ubicarse en la opción “cerrados” de la vista menú.
- Debe existir información de observaciones cerradas.

Flujo básico

- El aplicativo muestra una casilla en blanco “acto / condición” ubicado en la parte superior.
- El usuario ingresa un código de observación y hace click en el botón “buscar” del teclado, el sistema muestra en la lista los datos de las observaciones que coincidan con el parámetro ingresado.

Flujo(s) alternativo(s):

- **Visualizar detalle:** si el usuario desea visualizar el detalle de la observación, este debe seleccionar una observación de la lista y se iniciará el caso de uso “Visualizar Detalle Observación”.

Postcondición:

La información se encuentra registrada correctamente en la base de datos.

6.7. Caso de uso registrar observación

Caso de uso:

Registrar observación

Autores:

Anderson Kevin Bazán Urteaga

Oscar Geny Saavedra Guarniz

Propósito:

Registrar observaciones para que sean reportadas al personal observado.

Actores:

Supervisor

Precondición:

- Usuario debe autenticarse.
- Usuario ha debido haber ejecutado el caso de uso “Ver Observaciones”.
- Debe existir información de categorías.

- Debe existir información de zonas.
- Debe existir información de ocurrencias.
- Debe existir información de nivel de riesgo.
- Debe existir información de personas.

Flujo básico

- El aplicativo muestra por defecto seleccionado el botón de la opción de “acto”, el usuario puede escoger la opción “condición”.
- El usuario despliega y selecciona una categoría.
- El usuario despliega y selecciona una zona.
- El usuario ingresa una descripción en observación.
- El usuario despliega y selecciona una ocurrencia.
- El usuario despliega y selecciona un nivel riesgo, el aplicativo automáticamente genera el campo “N° días” para el cierre de la observación.
- El usuario ingresa una descripción en “acción correctiva”.
- El usuario despliega y selecciona un personal observado, este paso lo puede realizar tantas veces sea necesario hasta que el usuario haya seleccionado todas las personas involucradas en la observación.
- El usuario ingresa una descripción en “comentarios”.
- El usuario presiona el botón “guardar”, el aplicativo guarda la observación y muestra un mensaje indicando que los datos han sido guardados correctamente.

Flujo(s) alternativo(s):

- **Cambiar N° días para cierre de observación:** si el usuario desea cambiar el número de días para el cierre de la observación, este debe modificar la cantidad de días de la opción “N° días”.

Postcondición:

La información se registra correctamente en la base de datos.

6.8. Caso de uso adjuntar evidencia

Caso de uso:

Adjuntar evidencia

Autores:

Anderson Kevin Bazán Urteaga

Oscar Geny Saavedra Guarniz

Propósito:

Adjuntar las evidencias encontradas al realizar la observación.

Actores:

Supervisor

Precondición:

- Usuario debe autenticarse.
- Debe existir información de observaciones.
- Usuario ha debido haber ejecutado el caso de uso “Ver Observaciones”.

Flujo básico

- El aplicativo muestra en la parte superior información del detalle de la observación.
- El aplicativo muestra un botón circular con el símbolo de adjuntar.
- El usuario presiona en el triángulo invertido ubicado al lado derecho de la opción “personal observado”, el aplicativo muestra todas las personas involucradas en la observación.
- El usuario presiona en el triángulo invertido ubicado al lado derecho de la opción “observación”, el aplicativo muestra la descripción de la observación
- El usuario presiona en el triángulo invertido ubicado al lado derecho de la opción “acción correctiva”, el aplicativo muestra la descripción de la acción correctiva.
- El usuario adjunta evidencia presionando el botón “adjuntar”, el aplicativo le muestra las opciones para adjuntar evidencias (fotos o archivos).

Flujo(s) alternativo(s):

Postcondición:

La información se adjunta y registra correctamente en la base de datos y en el servidor de archivos.

6.9. Caso de uso registrar solución

Caso de uso:

Registrar solución

Autores:

Anderson Kevin Bazán Urteaga

Oscar Geny Saavedra Guarniz

Propósito:

Registrar la solución y adjuntar las evidencias de la observación reportada.

Actores:

Supervisor

Personal Observado

Precondición:

- Usuario debe autenticarse.
- Debe existir información de observaciones reportadas.
- Usuario ha debido haber ejecutado el caso de uso “Buscar Observaciones Pendientes”.

Flujo básico

- El aplicativo muestra en la parte superior información del detalle de la observación.
- El aplicativo muestra dos botones uno circular con el símbolo de adjuntar y otro que indica guardar.
- El usuario presiona en el triángulo invertido ubicado al lado derecho de la opción “observación”, el aplicativo muestra la descripción de la observación.
- El usuario presiona en el triángulo invertido ubicado al lado derecho de la opción “acción correctiva”, el aplicativo muestra la descripción de la acción correctiva.
- El usuario ingresa una descripción en el campo “solución”.
- El usuario presiona el botón “guardar”, el aplicativo guarda la solución de la observación y muestra un mensaje indicando que los datos han sido guardados correctamente.

Flujo(s) alternativo(s):

- **Adjuntar evidencia solución:** si el usuario desea registrar evidencias de la solución, este debe presionar el botón “adjuntar” ubicado en la parte inferior derecha de la vista, el aplicativo le muestra las opciones para adjuntar evidencias (fotos o archivos).
- **Descargar evidencias reportadas:** si el usuario desea descargar las evidencias reportadas en la solución, este debe presionar en la evidencia que desea descargar, el aplicativo descarga la evidencia adjuntada en el servidor de archivos.

Postcondición:

La información se registra correctamente en la base de datos y en el servidor de archivos.

6.10. Caso de uso visualizar detalle observación

Caso de uso:

Visualizar detalle observación

Autores:

Anderson Kevin Bazán Urteaga

Oscar Geny Saavedra Guarniz

Propósito:

Poder visualizar el detalle de las observaciones que han sido cerradas.

Actores:

Supervisor

Personal Observado

Precondición:

- Usuario debe autenticarse.
- Debe existir información de observaciones cerradas.
- Usuario ha debido haber ejecutado el caso de uso “Buscar Observaciones Cerradas”.

Flujo básico

- El aplicativo muestra en la parte superior información del detalle de la observación.
- El usuario presiona en el triángulo invertido ubicado al lado derecho de la opción “observación”, el aplicativo muestra la descripción de la observación.
- El usuario presiona en el triángulo invertido ubicado al lado derecho de la opción “acción correctiva”, el aplicativo muestra la descripción de la acción correctiva.
- El usuario presiona en el triángulo invertido ubicado al lado derecho de la opción “solución”, el aplicativo muestra la descripción de la solución.
- El usuario presiona en el triángulo invertido ubicado al lado derecho de la opción “personal observado”, el aplicativo muestra la lista de personas que fueron observadas.
- El usuario visualiza las evidencias reportadas.
- El usuario visualiza las evidencias de la solución.

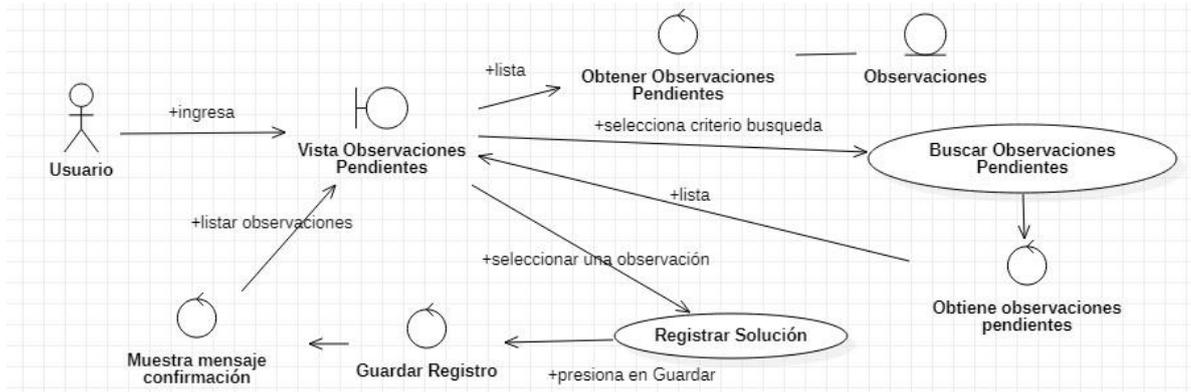


Figura 30: Diagrama de robustez ver observaciones pendientes

7.3. Caso de uso ver observaciones cerradas

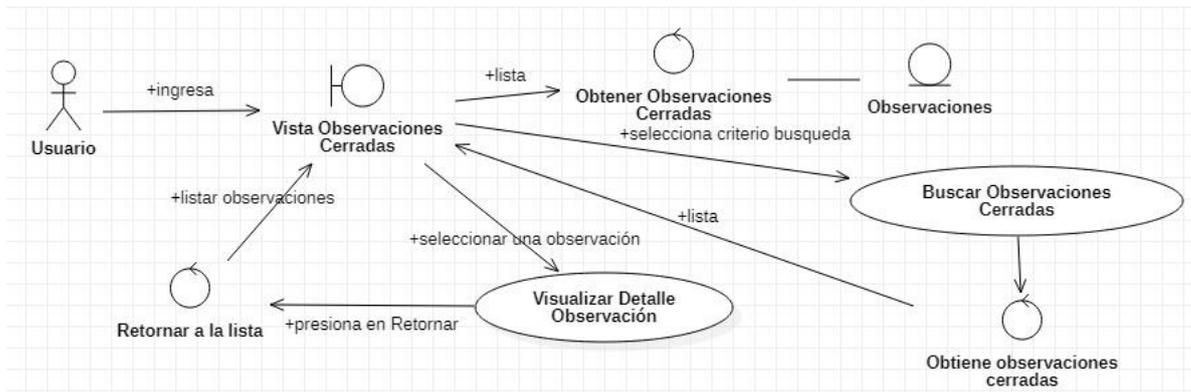


Figura 31: Diagrama de robustez ver observaciones cerradas

7.4. Caso de uso buscar observaciones

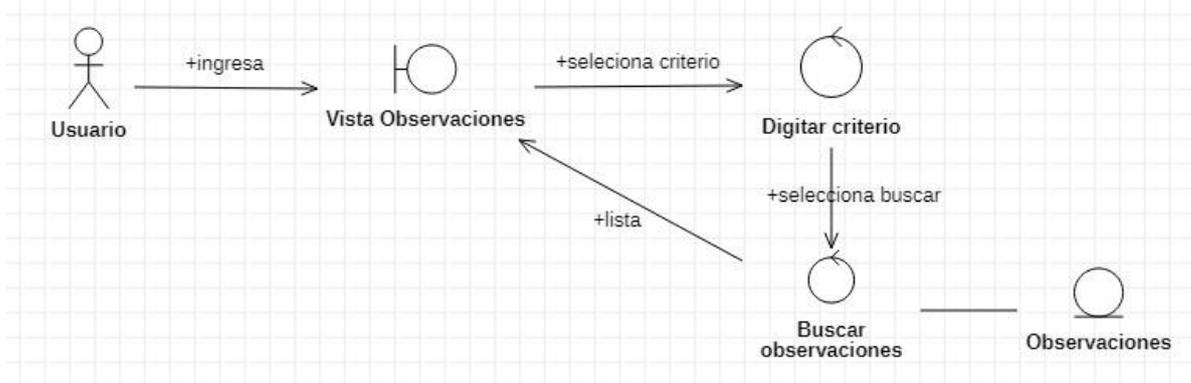


Figura 32: Diagrama de robustez buscar observaciones

7.5. Caso de uso buscar observaciones pendientes

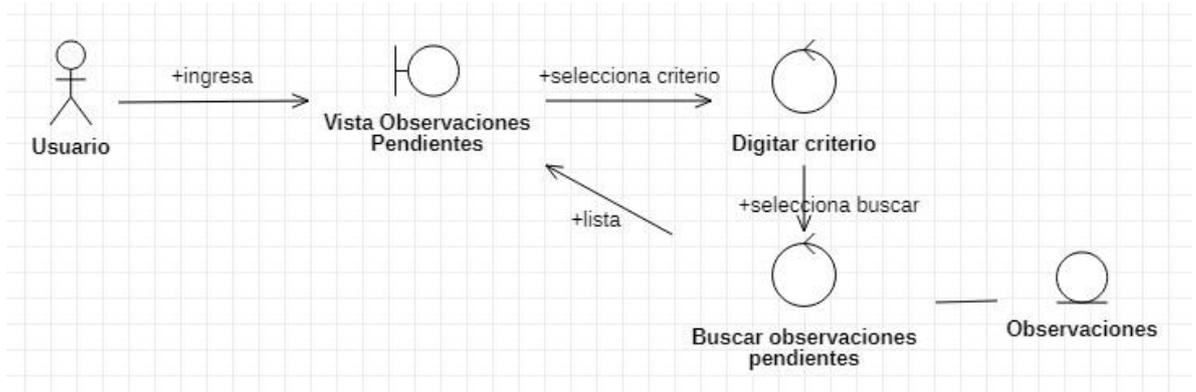


Figura 33: Diagrama de robustez buscar observaciones pendientes

7.6. Caso de uso buscar observaciones cerradas

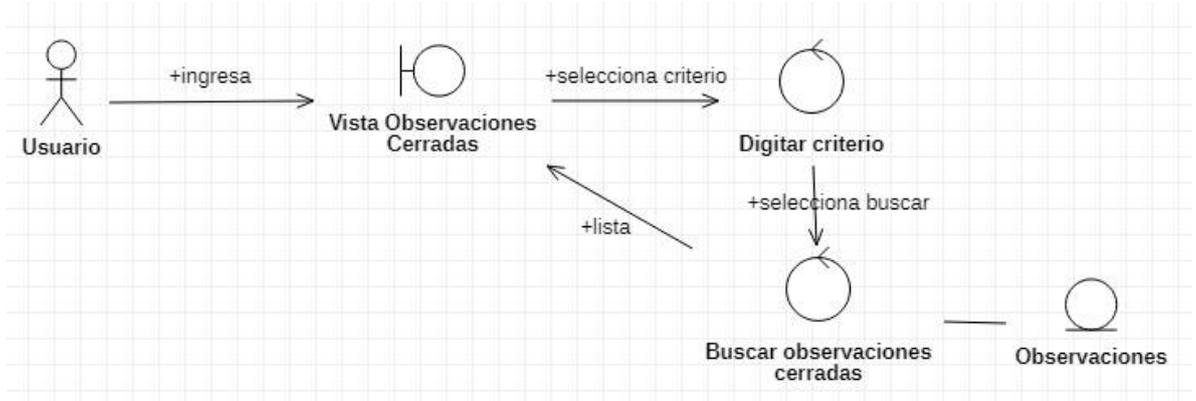


Figura 34: Diagrama de robustez buscar observaciones cerradas

7.7. Caso de uso registrar observación

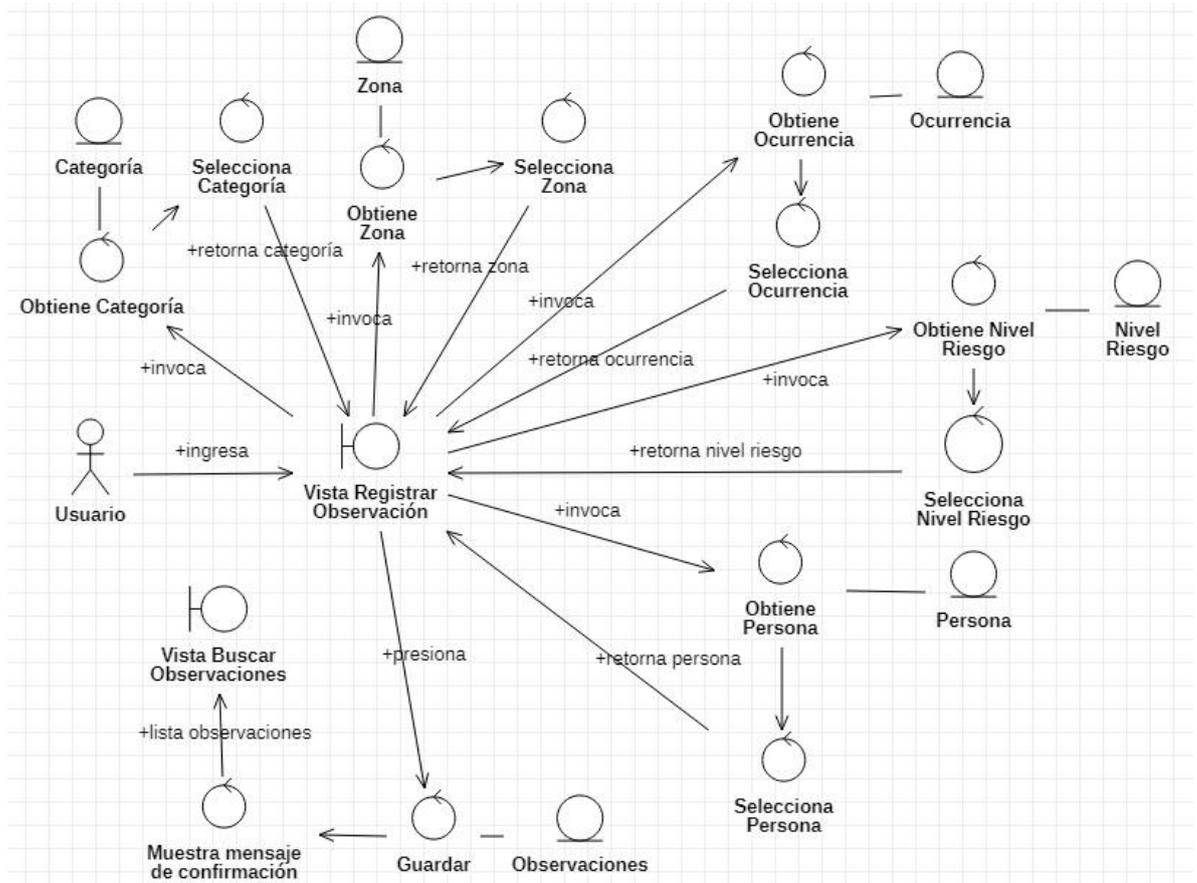


Figura 35: Diagrama de robustez registrar observación

7.8. Caso de uso adjuntar evidencia



Figura 36: Diagrama de robustez adjuntar evidencia

7.9. Caso de uso registrar solución



Figura 37: Diagrama de robustez registrar solución

7.10. Caso de uso visualizar detalle observación

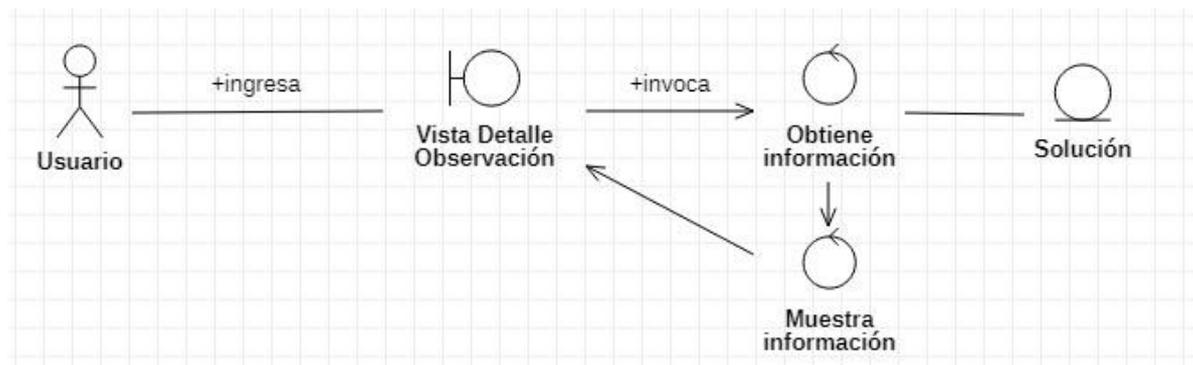


Figura 38: Diagrama de robustez visualizar detalle observación

8. Diagrama de clase

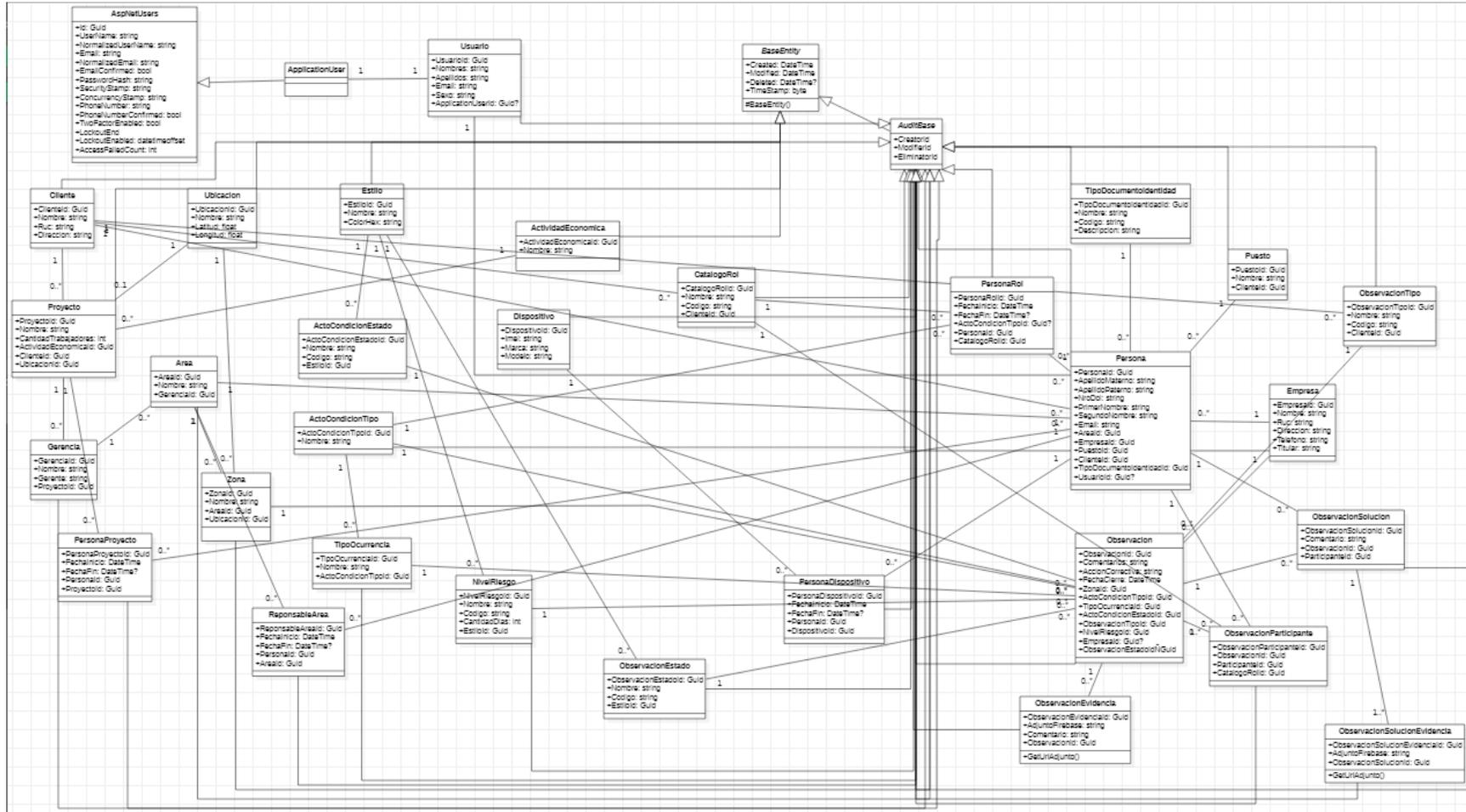


Figura 39: Diagrama de clases

Anexo N° 8. Manual de usuario

MANUAL DE USUARIO YANAPAY

ALCANCE

Este documento describe los pasos que se deben de realizar para poder interactuar de manera correcta con el aplicativo móvil Yanapay.

El aplicativo Yanapay, es una app que brinda la funcionalidad de reportar actos o condiciones observadas en las diversas áreas de trabajo, adicional a ello permite adjuntar evidencia mediante la cámara fotográfica o por medio del explorador de la memoria interna del teléfono.

A su vez este documento está orientado a usuarios con conocimientos sobre temas de seguridad y salud ocupacional en el trabajo.

MANUAL DE USUARIO

A continuación, se detallan los pasos que se deben de realizar para poder interactuar de manera adecuada con el aplicativo.

1. Pre requisitos

- Tener instalad el aplicativo en un celular Android
- Brindar los permisos de cámara.
- Brindar los permisos de lectura y escritura del almacenamiento interno.
- Tener un usuario y contraseña para el ingreso.

2. Iniciar sesión

- Ubicar el aplicativo instalado en el celular e ingresar, posterior a ello se mostrará una vista de presentación.



Figura 40: Vista presentación

- Ingresar las credenciales proporcionados por el administrador, se recomienda seleccionar la opción **recordar contraseña** con la finalidad de ingresar con mayor rapidez la próxima vez que se utilice el aplicativo.

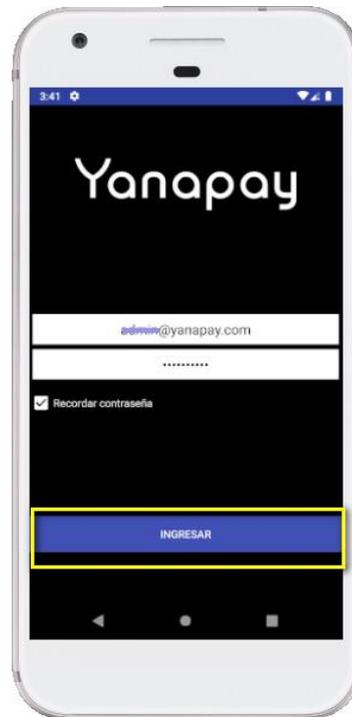


Figura 41: Vista iniciar sesión

Al presionar en **“INGRESAR”** se mostrará una vista con información de la cantidad de actos y condiciones registradas por el usuario, adicional a ello el aplicativo muestra información de registros que estén pendientes por sincronizar.



Figura 42: Vista pantalla de inicio

3. Registrar un acto o condición

Para registrar un nuevo acto o condición observada se debe ubicar la pestaña de “Observaciones”.



Figura 43: Vista opción observaciones

El aplicativo muestra una lista de observaciones registradas, para crear la nueva observación es necesario presionar sobre el botón de “*Nuevo*”.

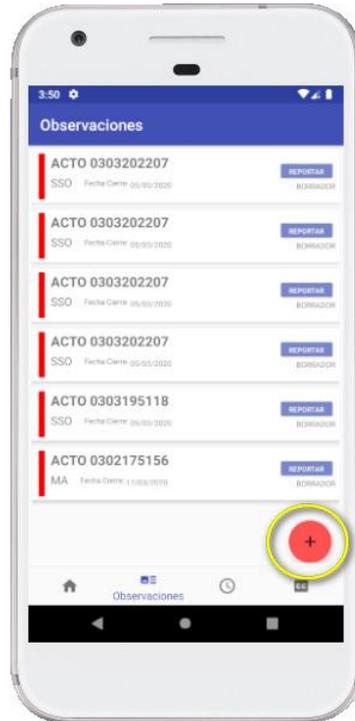


Figura 44: Vista opción agregar observación

Luego de seleccionar en **“Nuevo”**, se muestra el siguiente formulario, en el cual se podrá ingresar información concerniente al acto o condición observada.

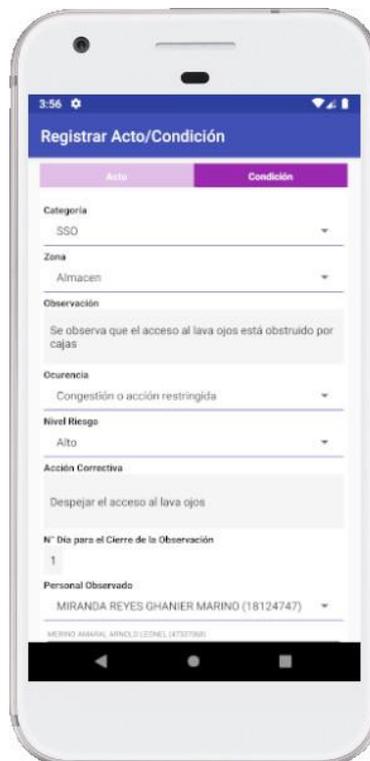


Figura 45: Vista registrar observación

Al finalizar de completar la información se deberá de presionar en **“GUARDAR”**, ubicado al final del formulario.

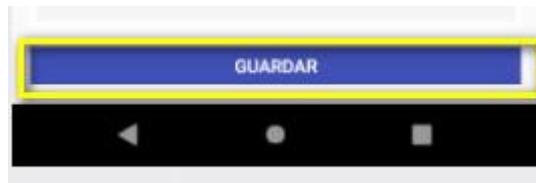


Figura 46: Vista opción guardar

El aplicativo guarda la información y posterior a ello muestra el formulario con todas las observaciones creadas.

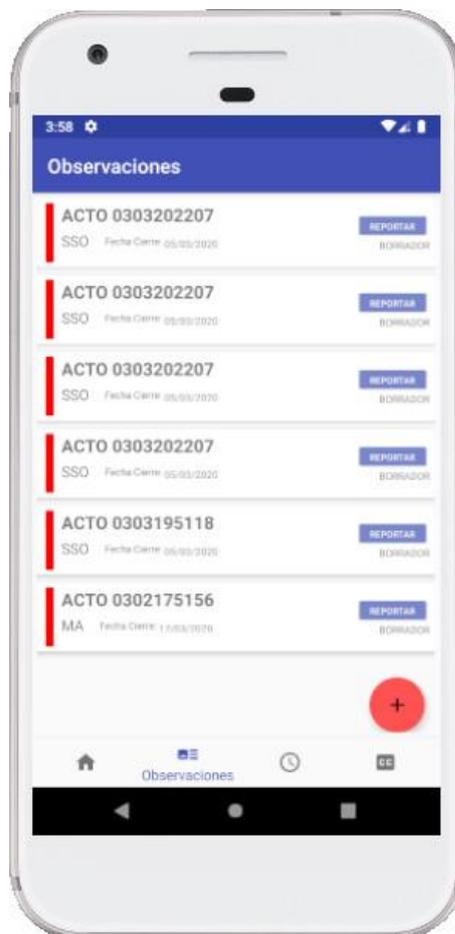


Figura 47: Vista lista de observaciones

4. Adjuntar evidencia

Luego de haber registrado un Acto o Condición, el aplicativo permite adjuntar archivos como evidencia de lo reportado, para eso se debe ubicar en la lista de “Observaciones”.



Ubicamos la observación a la cual deseamos adjuntar una evidencia y presionamos sobre ella:

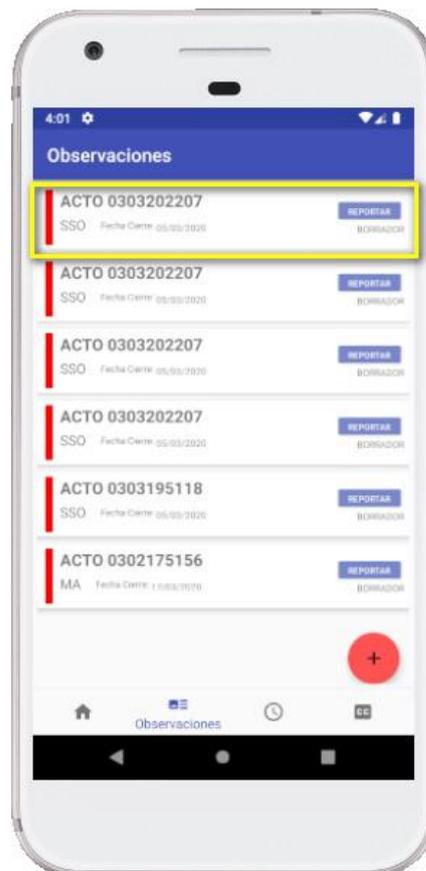


Figura 48: Vista seleccionar observación

Se mostrará una vista con el detalle de la observación registrada, en ella se podrá adjuntar archivos o fotos que sirvan como evidencia de lo reportado, para ello presionar sobre el botón de **“Adicionar”**.



Figura 49: Vista opción adjuntar evidencia

Según la opción que se escoja, el aplicativo abre la cámara para tomar una fotografía como evidencia o abre el explorador de archivos para adjuntar algún archivo.

Adjuntar evidencia desde la cámara



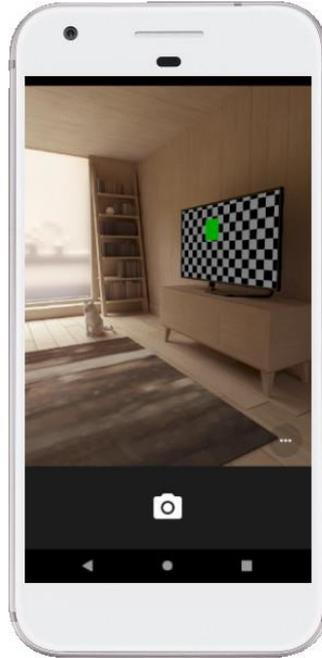


Figura 50: Vista captura de fotografía

Adjuntar evidencia desde el explorador de archivos

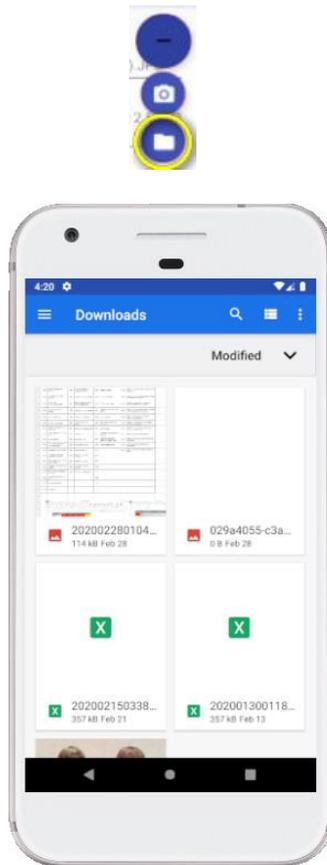


Figura 51: Vista buscar archivo de evidencia

5. Activar modo offline

Para activar este modo y que el aplicativo funcione en modo offline, se puede realizar de dos maneras, las cuales se detallan a continuación:

5.1. 1° Manera:

El aplicativo reconoce que no se tenga una cobertura de red “4G LTE” o que no esté conectado una señal de “Wifi”.

Cuando se está fuera de red, el aplicativo analiza la cobertura antes de iniciar sesión y al ingresar se mostrará un mensaje de “*Modo Offline*”.



Figura 52: Vista estado de red

Ingresar las credenciales con la que ingreso en un inicio cuando había conexión de red.



Figura 53: Vista opción ingresar

El aplicativo valida las credenciales guardadas y muestra la ventana inicio con el mensaje “Modo Offline”.

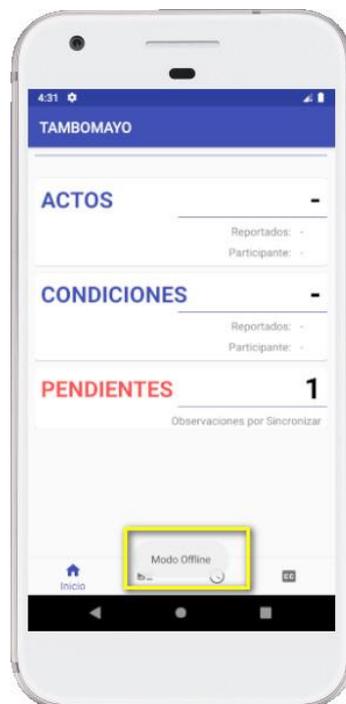


Figura 54: Mensaje modo offline

5.2. 2° Manera:

Antes de salir a campo ingresar al aplicativo y ubicarse en la opción de “Configuración”:



Figura 55: Vista opción de configuración

El aplicativo muestra una vista, en la cual se podrá observar el detalle de los datos de la persona que inicio sesión, adicional a ello muestra una opción que indica “**Activar modo Offline**”.



Figura 56: Vista configuración

Para activar este modo se debe presionar sobre la opción de “**Activar modo Offline**”.

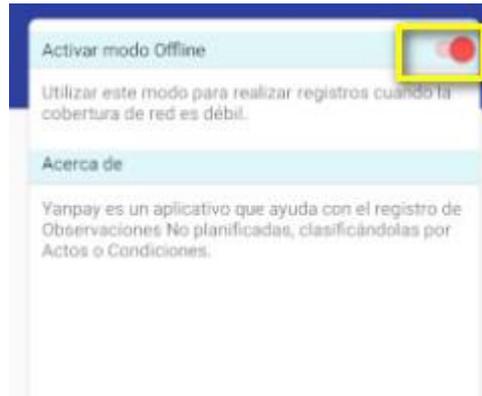


Figura 57: Vista botón offline

Cuando se encuentra en modo offline, el registro de información es igual como se describe en los primeros pasos cuando se ingresó teniendo cobertura de red.

Al finalizar de registrar la información en modo offline, cerramos el aplicativo, posterior a ello cuando se encuentra en una zona con cobertura de red o wifi se ingresa al aplicativo y se muestra la opción de sincronizar datos pendientes.



Figura 58: Vista opción sincronizar

Cuando seleccionemos en sincronizar, el aplicativo guarda la información para tenerla disponible en modo online, esta sincronización se realiza internamente en segundo plano por lo que es posible utilizar el aplicativo con total normalidad.

Luego de haber sincronizado la información, se podrá verificar en la pestaña de “Observaciones”.



6. Reportar una observación

Para dar por reportada una observación, se ha debido adjuntar la evidencia, posterior a ello en la vista de “Observaciones” se muestra un botón de **“REPORTAR”**.



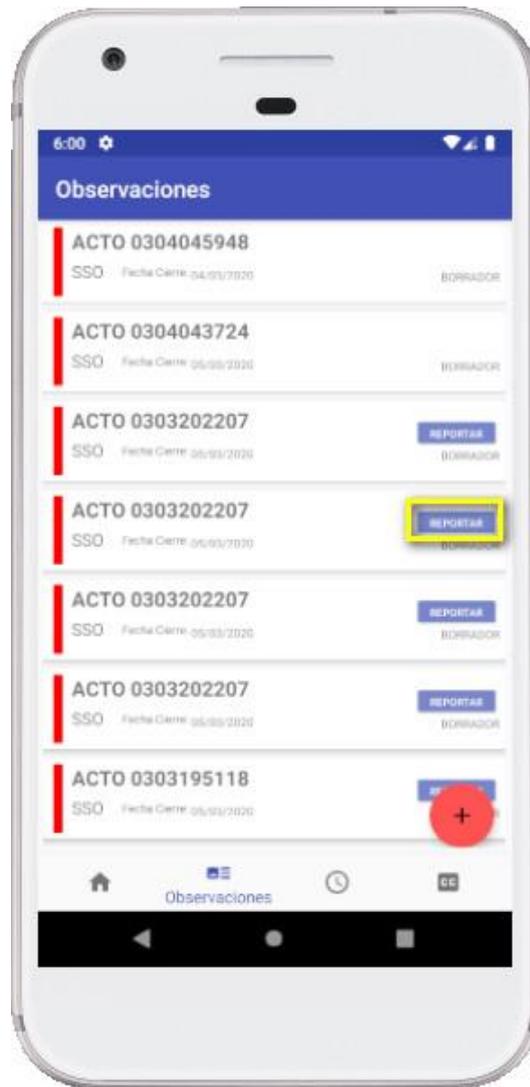


Figura 59: Vista opción reportar

Luego de seleccionar en el botón el aplicativo envía un correo al personal involucrado informando que ha sido reportado en una observación.

Yanapay

Observaciones

Observación pendiente de implementación



Estimado(a) hola,

Ssoma 03 Seguridad y Salud Ocupacional ha reportado la Observación:
Disposición de residuos incorrecta. en la zona Almacen. Por lo cual se
requiere que levante las observaciones hasta el 17/03/2020.

Evidencias

[20200302125606_TimePhoto_20200302_030657.jpg](#)

[20200302125606_TimePhoto_20200302_030657.jpg](#)

Figura 60: Vista mensaje de correo

Anexo N° 9. Vincular aplicativo hacia Firebase Performance Monitoring

**VINCULAR YANAPAY A FIREBASE
PERFORMANCE MONITORING**

ALCANCE

Este documento describe los pasos que se realizaron para vincular el aplicativo móvil Yanapay hacia la plataforma Firebase Performance Monitoring.

Firebase Performance Monitoring es una plataforma que se encarga de obtener estadísticas sobre el rendimiento de los aplicativos vinculados, brinda diagramas estadísticos los cuales pueden ser analizados para dar seguimiento y mejorar el rendimiento de las aplicaciones.

FIREBASE

Se utiliza este servicio de Google ya que cuenta con diferentes herramientas que ayudan al desarrollo y monitoreo de aplicaciones móviles, a continuación, se detallan los pasos realizados para vincular el aplicativo Yanapay hacia Performance Monitoring.

1. Prerrequisitos

- Tener una cuenta de Google.
- Tener el código fuente del aplicativo móvil al cual vincularemos con Firebase.

2. Crear un proyecto en Firebase

Para crear un proyecto en Firebase es necesario registrarse e ir a la consola en la cual crearemos el proyecto Yanapay que luego vincularemos a nuestro aplicativo móvil.



Figura 61: Página de inicio de Firebase

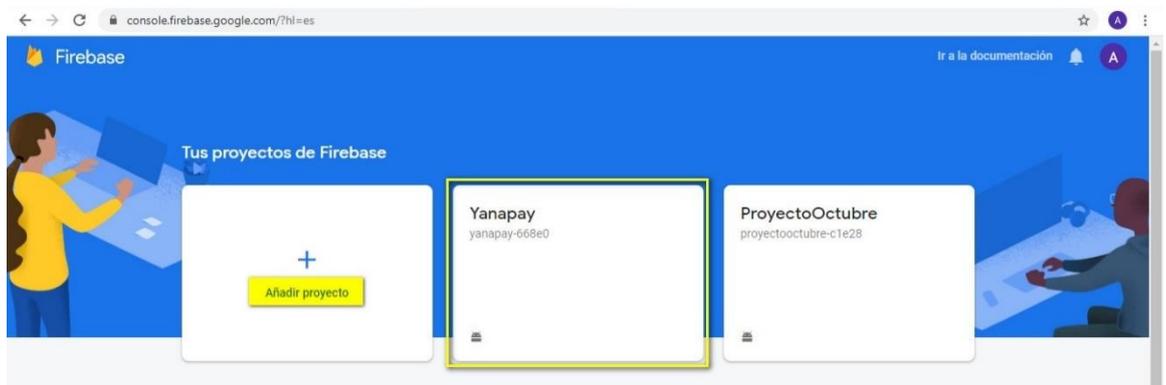


Figura 62: Añadir proyecto Yanapay en consola de Firebase

3. Vincular Performance Firebase al aplicativo móvil

Luego de haber creado el proyecto en Firebase de debe ubicar el IDE de Android Studio donde esté abierto el código fuente del aplicativo móvil.

Para vincular esta herramienta se realizaron los siguientes pasos:

3.1. Agregar el SDK de Performance Monitoring al aplicativo

Ubicar el archivo Gradle a nivel de la app y agregar la dependencia de Performance.

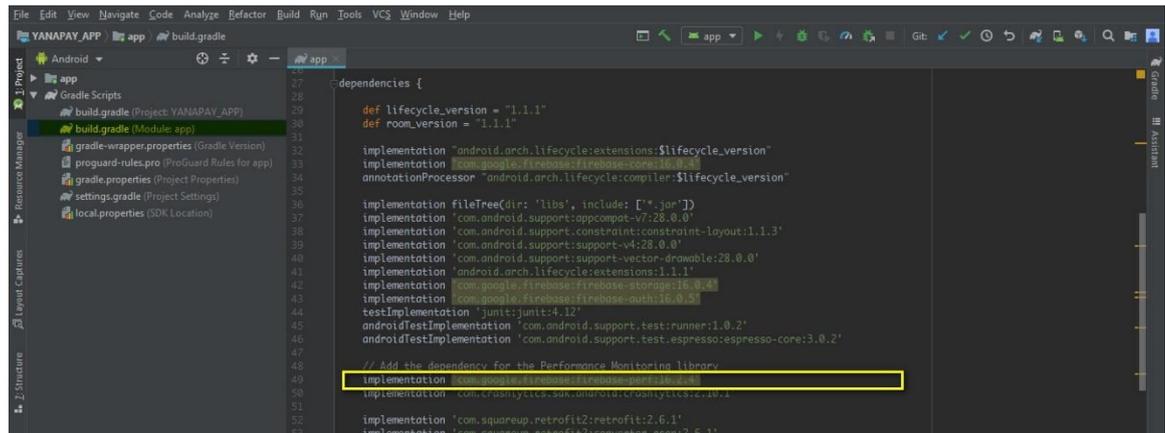


Figura 63: Dependencia para adicionar el SDK

Luego de agregar la dependencia se compila la app.

3.2. Agregar el complemento de Performance Monitoring al aplicativo

Ubicar el archivo Gradle a nivel de la app y agregar el complemento.

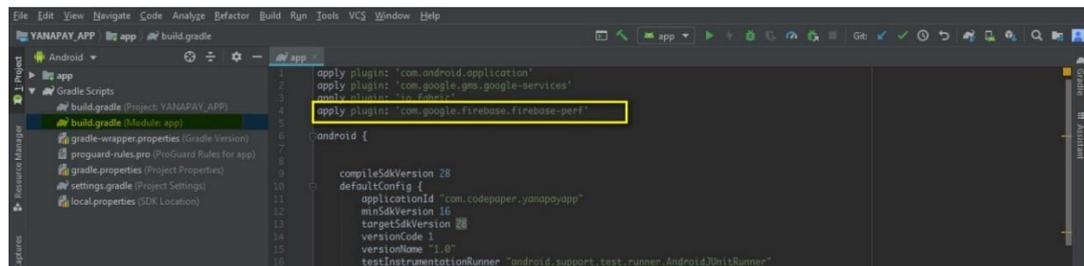


Figura 64: Complemento para Performance Monitoring

Luego de ello ubicar el archivo Gradle a nivel del proyecto y agregar la siguiente dependencia.

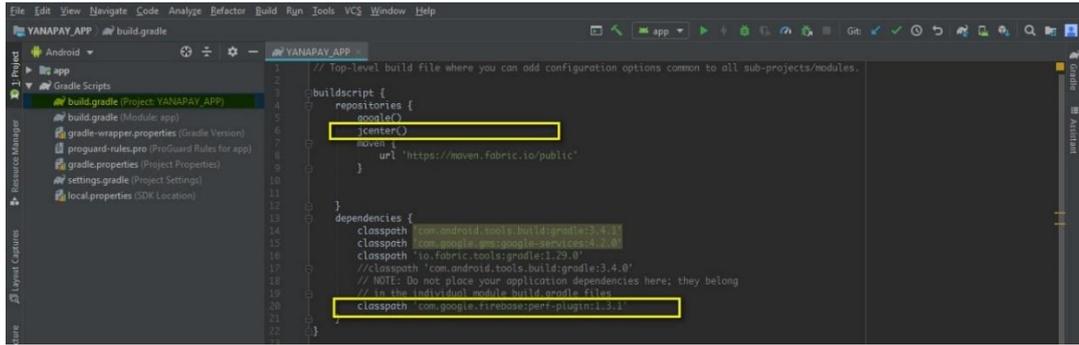


Figura 65: Complemento y dependencia para Performance Monitoring

Finalmente, al terminar con los pasos anteriores se compila la app.

4. Revisar en la consola de Firebase que se haya vinculado el aplicativo

Para ello debemos ubicarse en la consola de Firebase e ingresar a la herramienta de Performance Monitoring.

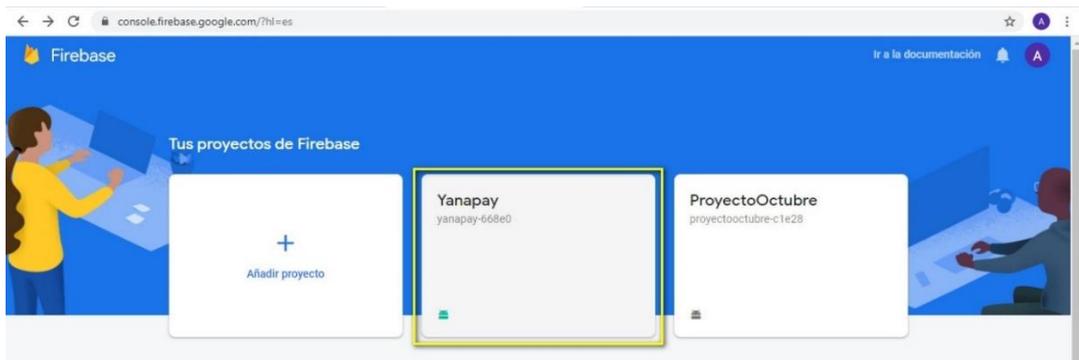


Figura 66: Proyecto Yanapay creado en Firebase

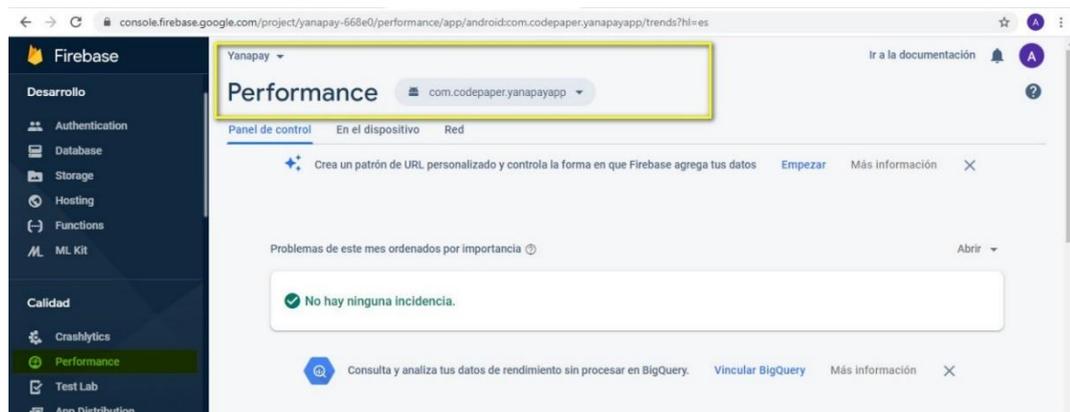


Figura 67: Herramienta Performance vinculada al aplicativo

Anexo N° 10. Resultados de la cobertura de red según la encuesta

¿Cuenta con cobertura de red en todo momento cuando se encuentra en la operación minera?
11 respuestas

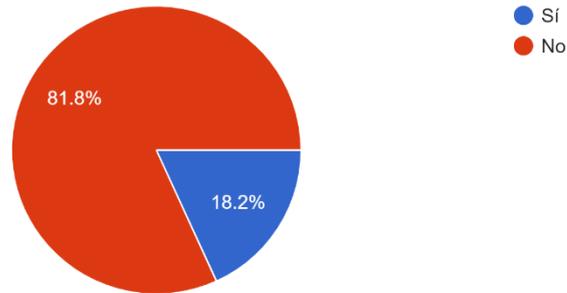


Figura 68: Resultados de la cobertura de red dentro de la operación minera

Marque los lugares de la operación minera donde la cobertura de red es menor
11 respuestas

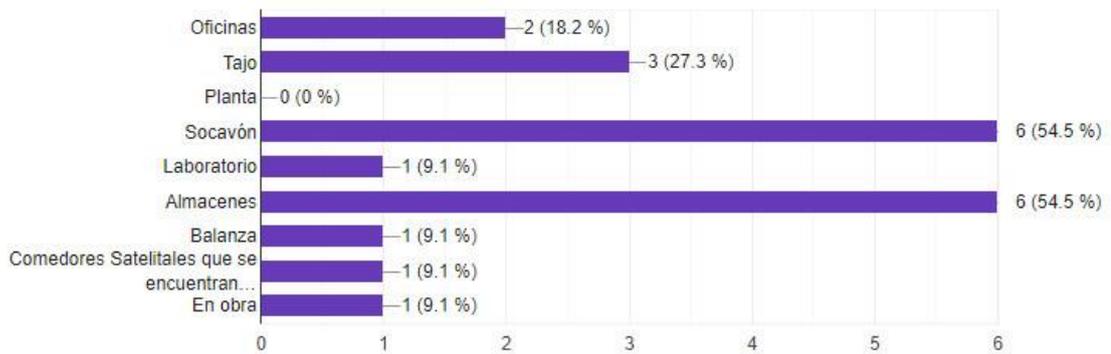


Figura 69: Resultados de las zonas donde la cobertura de red es menor

Anexo N° 11. Resultados de herramientas de tecnología móvil según la encuesta

¿Su área o empresa cuenta con herramientas de tecnología móvil adecuadas que ayuden a realizar sus actividades de campo dentro de la operación minera con normalidad?

11 respuestas

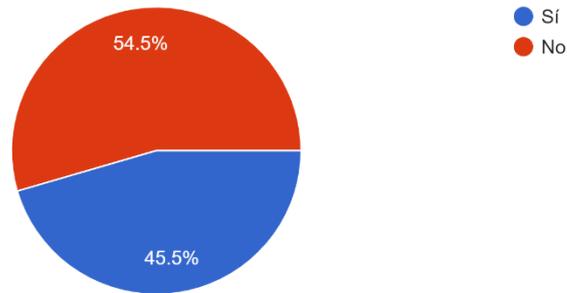


Figura 70: Resultados que indican si se utilizan tecnologías móviles en las actividades de campo

¿Cuenta con algún dispositivo móvil del tipo Smartphone?

11 respuestas

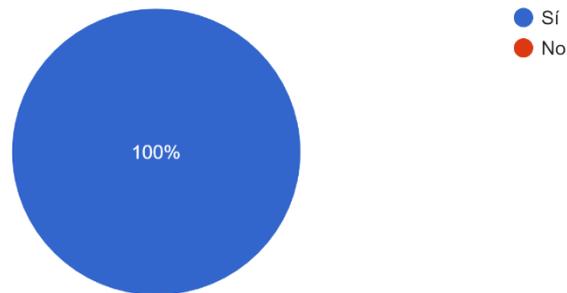


Figura 71: Resultados que indican si cuentan con dispositivos móviles del tipo Smartphone

Señale el sistema operativo del dispositivo móvil con el que cuenta

11 respuestas

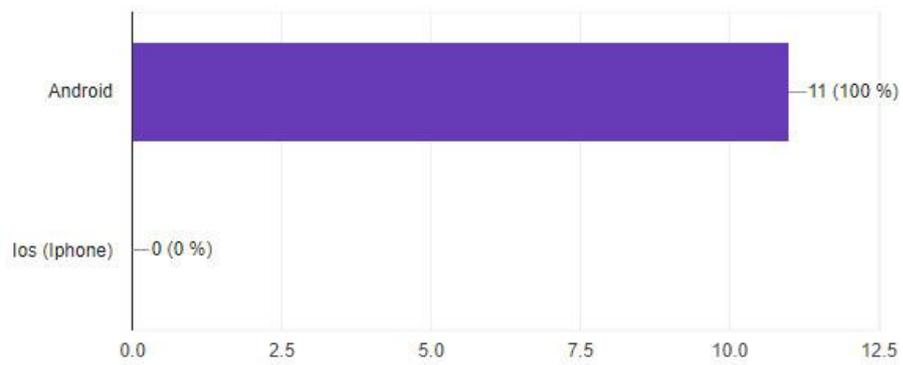


Figura 72: Resultados del sistema operativo con el que cuenta su dispositivo móvil

Anexo N° 12. Resultados ayuda de herramientas móviles según la encuesta

¿Usted cree que su dispositivo móvil sería de gran ayuda para realizar sus actividades de campo dentro de la operación minera?

11 respuestas

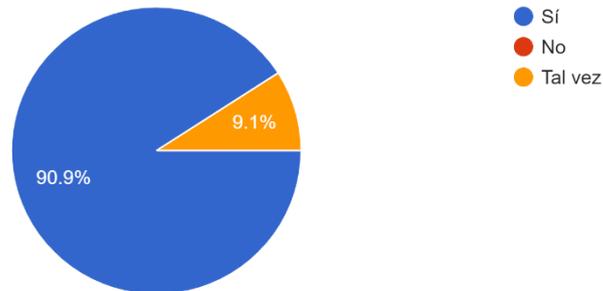


Figura 73: Resultados que indican si el dispositivo móvil sería de ayuda para realizar actividades de campo

¿Usted utilizaría una aplicación móvil que le ayude agilizar sus actividades de campo dentro de la operación minera?

11 respuestas

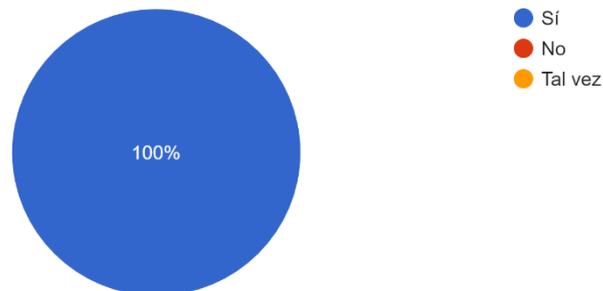


Figura 74: Resultados que indican si se utilizaría una aplicación móvil que ayude agilizar las actividades de campo

Anexo N° 13. Desarrollo de la aplicación en Android Studio

**DESARROLLO DE LA APLICACIÓN YANAPAY
EN ANDROID STUDIO**

ALCANCE

Este documento describe los pasos y la arquitectura utilizada para el desarrollo del aplicativo móvil Yanapay en el IDE de Android Studio.

El desarrollo de este aplicativo está orientado a cumplir con la actividad de reportar actos y condiciones observadas dentro de la operación minera por el personal del área de Seguridad y Salud Ocupacional y Medio Ambiente mientras realizan sus actividades de supervisión en campo.

Así mismo para el desarrollo también se tuvo en cuenta que en las zonas alejadas de la operación minera no se cuenta con una señal de red estable, por lo que se realizó el desarrollo en las modalidades online y offline.

1. Android Studio

Se considera este software ya que es el IDE oficial para el desarrollo de aplicaciones Android, así mismo cuenta con la opción de desarrollar en los lenguajes de programación Java y Kotlin; para el desarrollo del aplicativo Yanapay se optó por utilizar Java, esta decisión fue tomada por que las personas involucradas con el desarrollo dominan dicho lenguaje de programación.

2. Arquitectura

La arquitectura utilizada para el desarrollo del aplicativo es MVP (modelo, vista, presentador), incluyendo en el modelo una clase Interactor que cumpla la función de decidir si la información es obtenida o enviada a una fuente de datos remota o local, esto con la necesidad de trabajar bajo las modalidades online y offline.

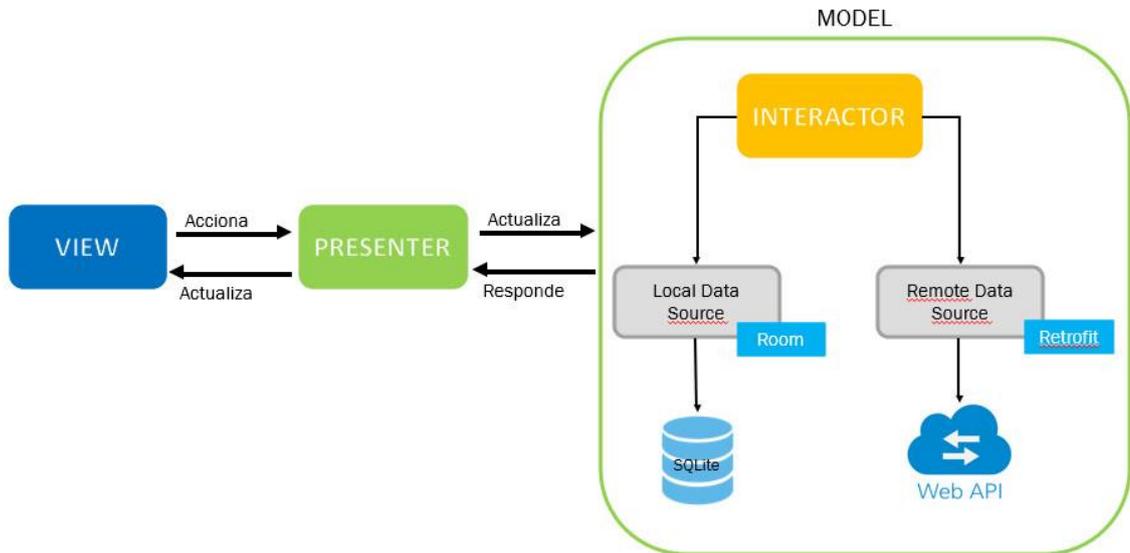


Figura 75: Arquitectura MVP

Esta arquitectura organiza mejor la distribución de archivos y asigna responsabilidades dependientes por cada capa, contribuyendo a tener un código mucho más limpio y con la facilidad de poder realizar cualquier tipo de cambio en el tiempo.

2.1. Capa View

En esta capa podemos encontrar todas clases de las interfaces de usuario que han sido creadas, estas interfaces han sido creadas considerando los principios que debe tener una aplicación móvil lite, la cual indica que las interfaces deben ser limpias, sencillas,

sin sobrecarga de animaciones e imágenes que puedan provocar tiempos de demora a la hora de cargar la información.

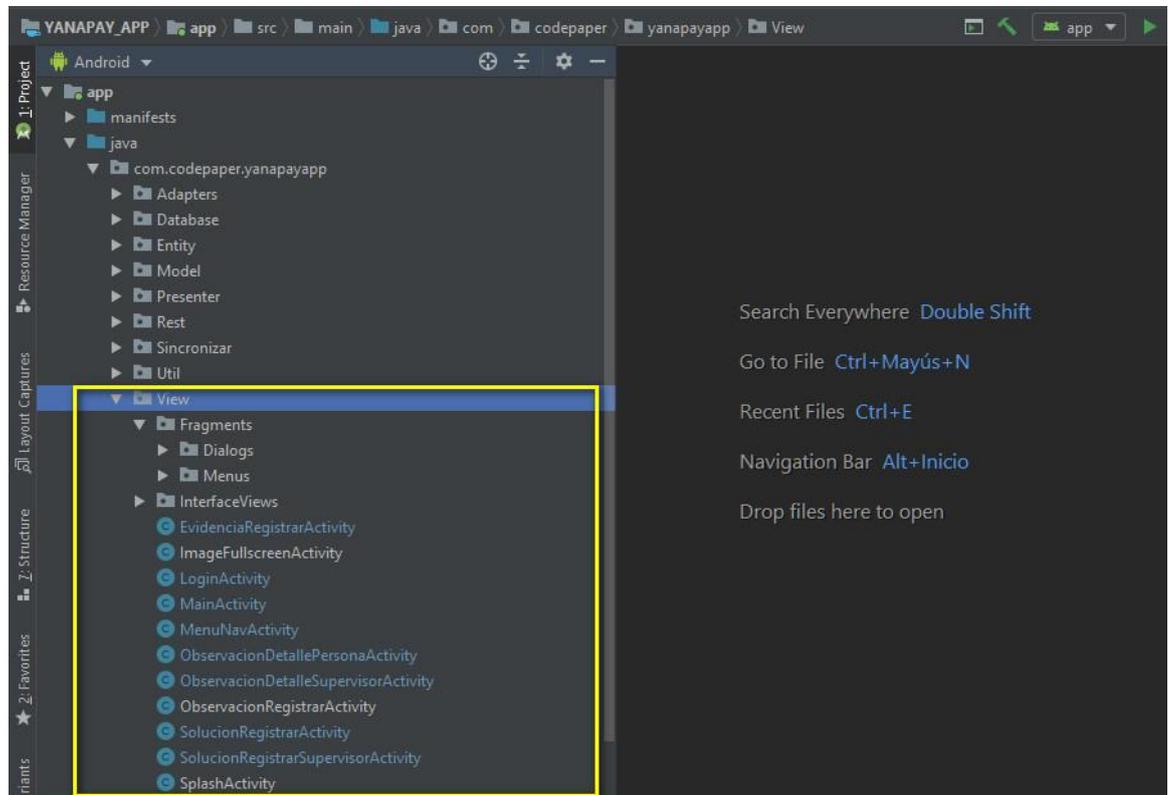


Figura 76: Paquete con las clases View

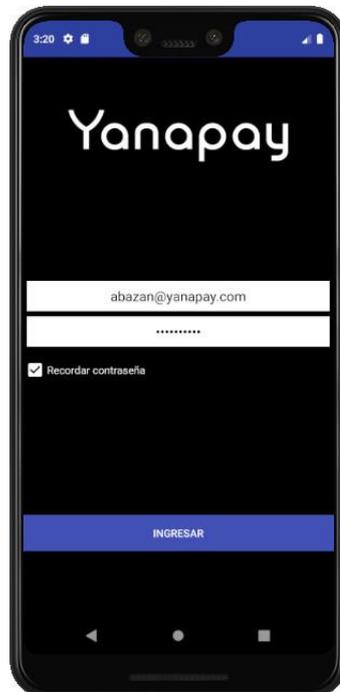


Figura 77: Vista Login



Figura 78: Vista Inicio

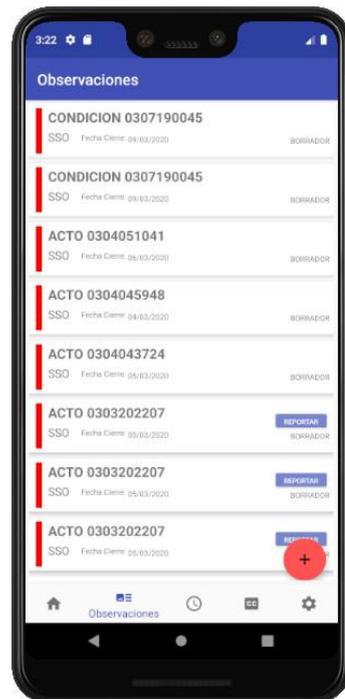


Figura 79: Vista Observaciones

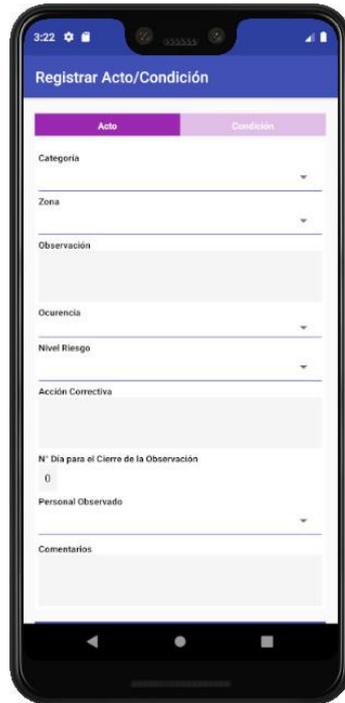


Figura 80: Vista registrar acto/condición



Figura 81: Vista evidencias



Figura 82: Vista configuración

2.2. Capa Presenter

Esta capa cumple la función de comunicar a la “Vista” con el “Modelo”, es por ello que en ella se puede encontrar las clases “Presenter” que solicitan información al “Modelo” para luego enviárselo a la “Vista”.

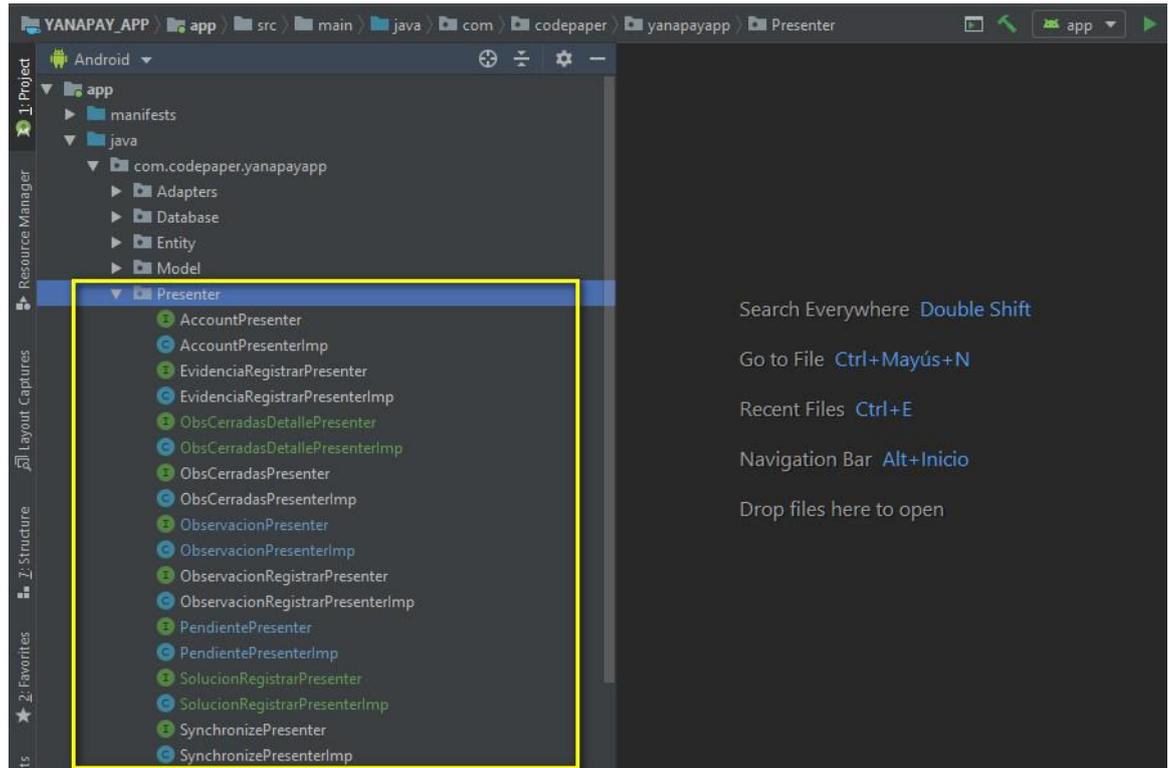


Figura 83: Paquete con las clases Presenter

2.3. Capa Model

En esta capa se encuentran todas las clases relacionadas al acceso de datos u obtención de información, en ellas se adicionó una condición que valida si la configuración de la aplicación se encuentra en modo online u offline; dependiendo el modo en el que se encuentre la configuración de la aplicación los datos son obtenidos de una fuente de datos remota o local.

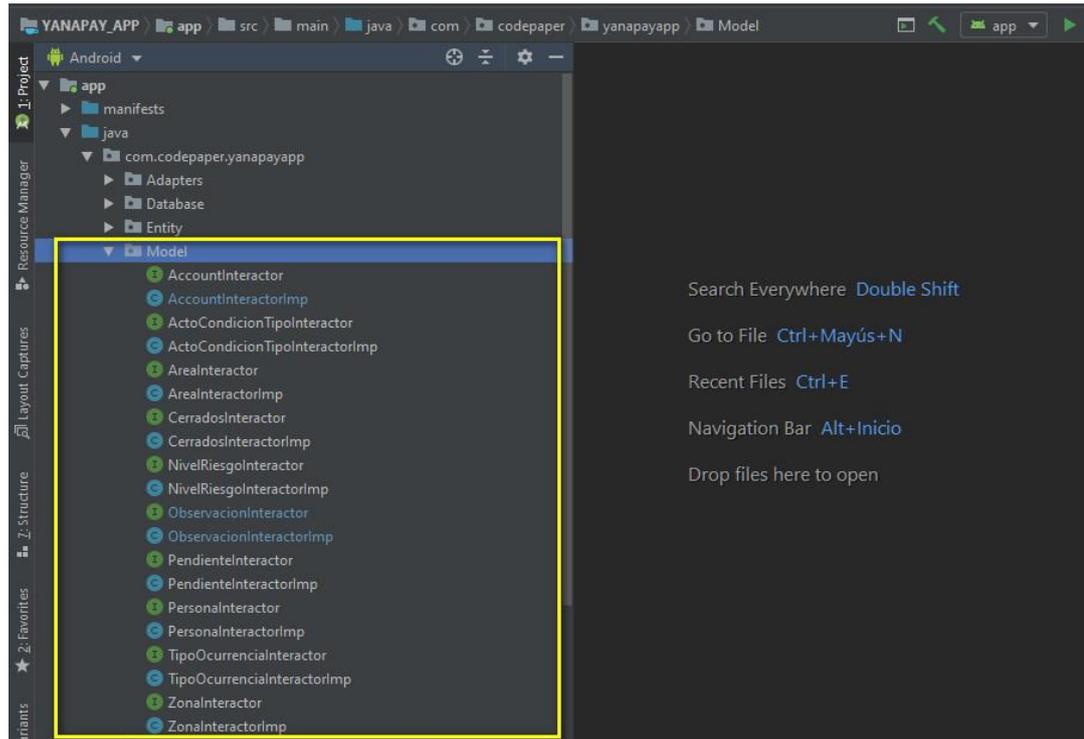


Figura 84: Paquete con las clases Model

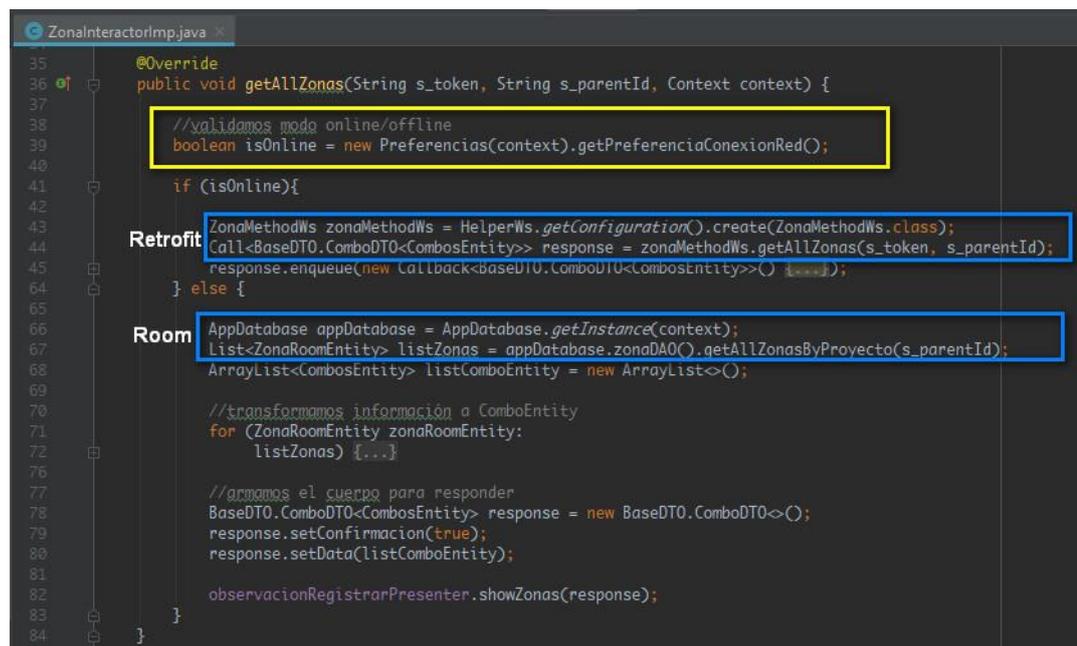


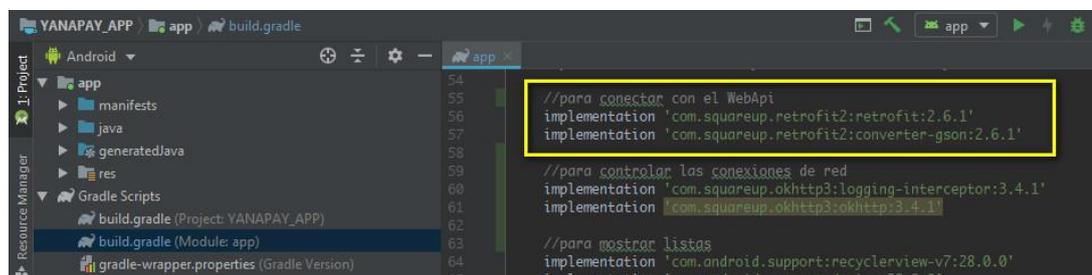
Figura 85: Clases Interactor que valida de donde se obtiene la información

3. Complementos SDK

Para la elaboración del aplicativo móvil se tuvieron que considerar algunos complementos que contribuyeran a las funcionalidades que deseamos obtener, estos complementos son detallados a continuación.

3.1. Retrofit

Complemento que brinda la posibilidad de consumir y obtener información del servicio Web Api, este servicio solicita y devuelve información en formato Json, es por ello que se incluyó la librería **GSON**, la cual brinda la posibilidad de convertir objetos Java en su representación Json y viceversa.

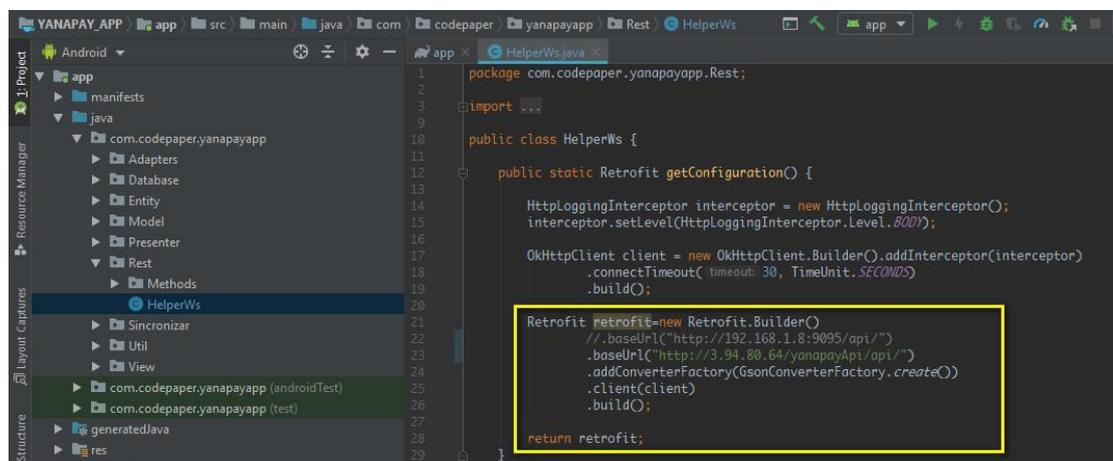


```

54
55
56 //para conectar con el WebApi
57 implementation 'com.squareup.retrofit2:retrofit:2.6.1'
58 implementation 'com.squareup.retrofit2:converter-gson:2.6.1'
59
60 //para controlar las conexiones de red
61 implementation 'com.squareup.okhttp3:logging-interceptor:3.4.1'
62 implementation 'com.squareup.okhttp3:okhttp:3.4.1'
63
64 //para mostrar listas
65 implementation 'com.android.support:recyclerview-v7:28.0.0'
66 implementation 'com.android.support:design:28.0.0'

```

Figura 86: SDK Retrofit y Gson



```

1 package com.codepaper.yanapayapp.Rest;
2
3 import ...
4
5 public class HelperWs {
6
7     public static Retrofit getConfiguration() {
8
9         HttpLoggingInterceptor interceptor = new HttpLoggingInterceptor();
10        interceptor.setLevel(HttpLoggingInterceptor.Level.BODY);
11
12        OkHttpClient client = new OkHttpClient.Builder().addInterceptor(interceptor)
13            .connectTimeout(timeout: 30, TimeUnit.SECONDS)
14            .build();
15
16        Retrofit retrofit=new Retrofit.Builder()
17            // .baseUrl("http://192.168.1.8:9095/api/")
18            .baseUrl("http://3.94.80.64/yanapayApi/api/")
19            .addConverterFactory(GsonConverterFactory.create())
20            .client(client)
21            .build();
22
23        return retrofit;
24    }
25 }

```

Figura 87: Clase HelperView para iniciar Retrofit

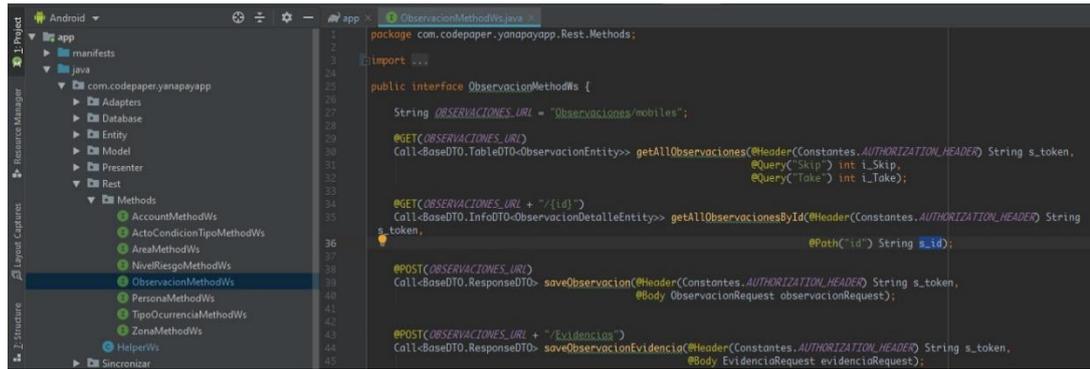


Figura 88: Etiquetas y métodos de Retrofit

3.2. OkHttp3

Librería que se utiliza para controlar la conexión a la red, en ella especificamos el tiempo de espera cuando solicitamos una petición al servicio Web Api.

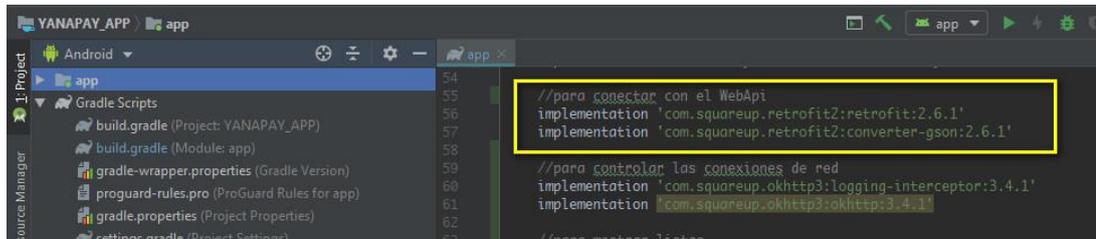


Figura 89: SDK Okhttp3

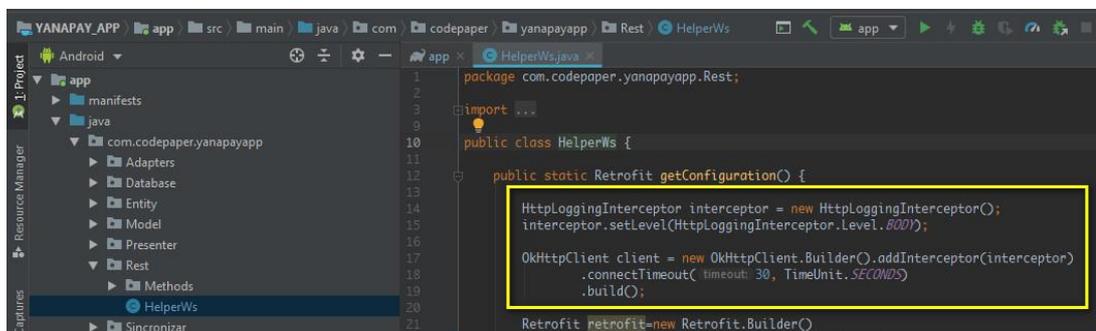


Figura 90: Clases HelperView para iniciar Okhttp3

3.3. Room

Librería que se utiliza para acceder a la base de datos local creada en la memoria cache del dispositivo.

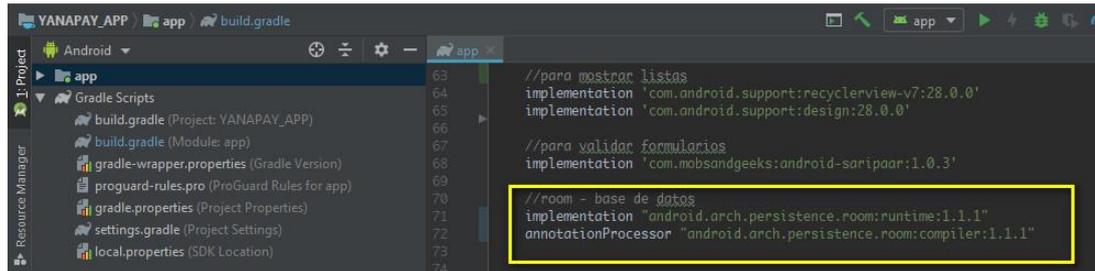


Figura 91: SDK Room

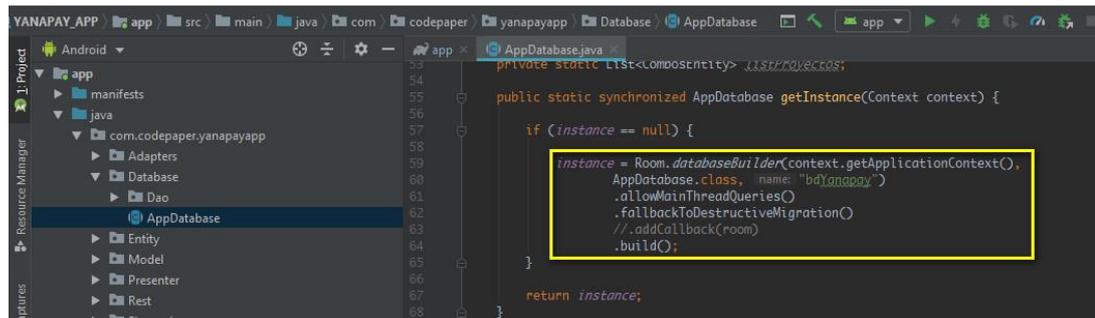


Figura 92: Clase AppDatabase para crear la base de datos local

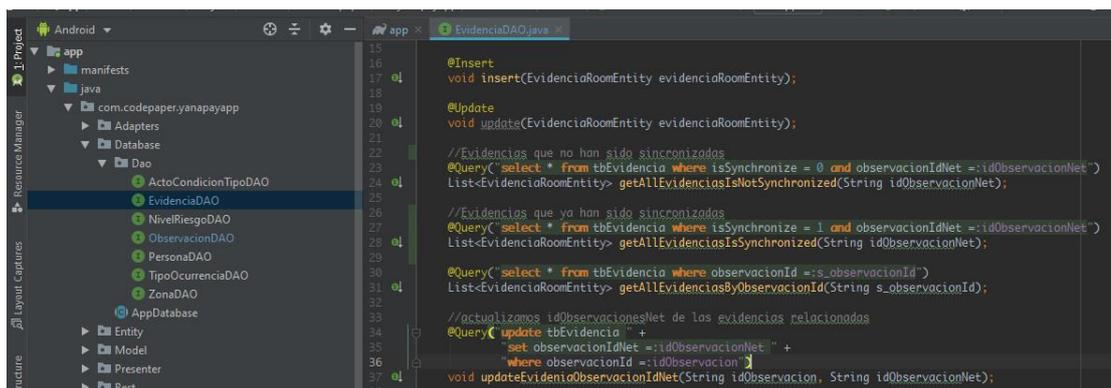


Figura 93: Etiquetas y métodos de Room

3.4. Firebase Storage

Librería que se utiliza para almacenar los archivos de las evidencias adjuntas por los usuarios en las observaciones que realizan, adicionalmente también es utilizada para descargar los archivos adjuntos.

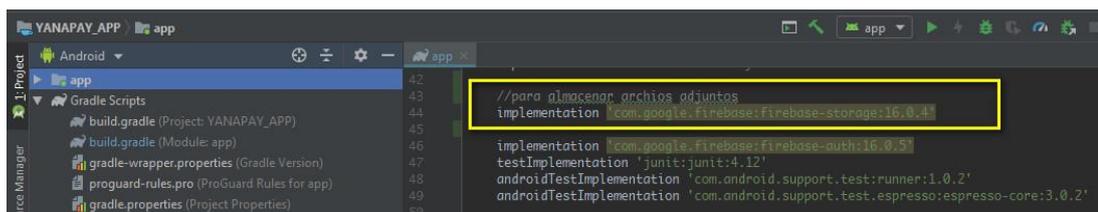
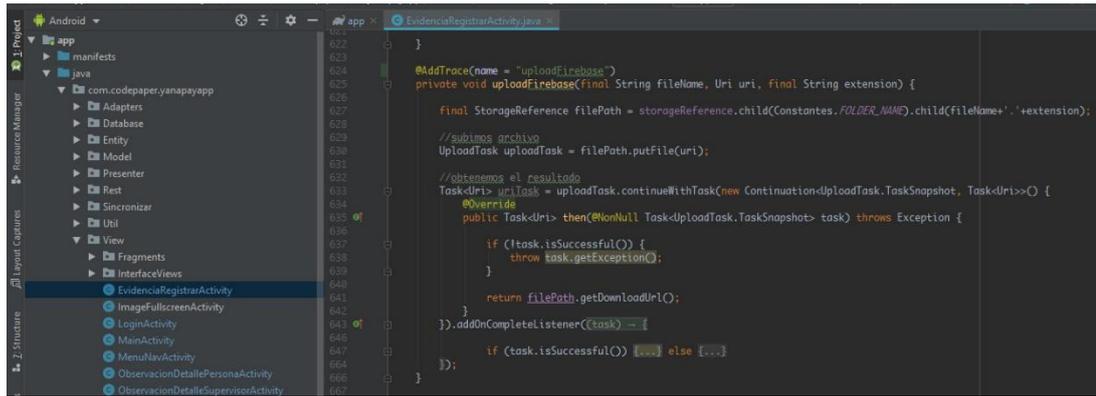


Figura 94: SDK Firebase Storage

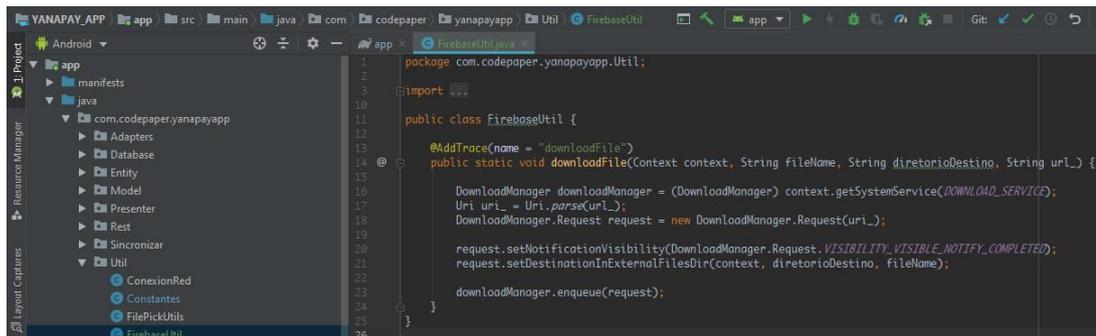


```

622 }
623 }
624
625 @AddTrace(name = "uploadFirebase")
626 private void uploadFirebase(final String fileName, Uri uri, final String extension) {
627     final StorageReference filePath = storageReference.child(Constants.FOLDER_NAME).child(fileName+'.'+extension);
628
629     //subimos archivo
630     UploadTask uploadTask = filePath.putFile(uri);
631
632     //obtenemos el resultado
633     Task<Uri> uriTask = uploadTask.continueWithTask(new Continuation<UploadTask.TaskSnapshot, Task<Uri>>() {
634         @Override
635         public Task<Uri> then(@NonNull Task<UploadTask.TaskSnapshot> task) throws Exception {
636             if (!task.isSuccessful()) {
637                 throw task.getException();
638             }
639             return filePath.getDownloadUrl();
640         }
641     }).addOnCompleteListener((task) -> {
642         if (task.isSuccessful()) {
643             //...
644         } else {
645             //...
646         }
647     });
648 }
649
650 }
651 }
652 }
653 }
654 }
655 }
656 }
657 }
658 }
659 }
660 }
661 }
662 }
663 }
664 }
665 }
666 }
667 }
668 }
669 }
670 }
671 }
672 }
673 }
674 }
675 }
676 }
677 }
678 }
679 }
680 }
681 }
682 }
683 }
684 }
685 }
686 }
687 }
688 }
689 }
690 }
691 }
692 }
693 }
694 }
695 }
696 }
697 }
698 }
699 }
700 }
701 }
702 }
703 }
704 }
705 }
706 }
707 }
708 }
709 }
710 }
711 }
712 }
713 }
714 }
715 }
716 }
717 }
718 }
719 }
720 }
721 }
722 }
723 }
724 }
725 }
726 }
727 }
728 }
729 }
730 }
731 }
732 }
733 }
734 }
735 }
736 }
737 }
738 }
739 }
740 }
741 }
742 }
743 }
744 }
745 }
746 }
747 }
748 }
749 }
750 }
751 }
752 }
753 }
754 }
755 }
756 }
757 }
758 }
759 }
760 }
761 }
762 }
763 }
764 }
765 }
766 }
767 }
768 }
769 }
770 }
771 }
772 }
773 }
774 }
775 }
776 }
777 }
778 }
779 }
780 }
781 }
782 }
783 }
784 }
785 }
786 }
787 }
788 }
789 }
790 }
791 }
792 }
793 }
794 }
795 }
796 }
797 }
798 }
799 }
800 }
801 }
802 }
803 }
804 }
805 }
806 }
807 }
808 }
809 }
810 }
811 }
812 }
813 }
814 }
815 }
816 }
817 }
818 }
819 }
820 }
821 }
822 }
823 }
824 }
825 }
826 }
827 }
828 }
829 }
830 }
831 }
832 }
833 }
834 }
835 }
836 }
837 }
838 }
839 }
840 }
841 }
842 }
843 }
844 }
845 }
846 }
847 }
848 }
849 }
850 }
851 }
852 }
853 }
854 }
855 }
856 }
857 }
858 }
859 }
860 }
861 }
862 }
863 }
864 }
865 }
866 }
867 }
868 }
869 }
870 }
871 }
872 }
873 }
874 }
875 }
876 }
877 }
878 }
879 }
880 }
881 }
882 }
883 }
884 }
885 }
886 }
887 }
888 }
889 }
890 }
891 }
892 }
893 }
894 }
895 }
896 }
897 }
898 }
899 }
900 }
901 }
902 }
903 }
904 }
905 }
906 }
907 }
908 }
909 }
910 }
911 }
912 }
913 }
914 }
915 }
916 }
917 }
918 }
919 }
920 }
921 }
922 }
923 }
924 }
925 }
926 }
927 }
928 }
929 }
930 }
931 }
932 }
933 }
934 }
935 }
936 }
937 }
938 }
939 }
940 }
941 }
942 }
943 }
944 }
945 }
946 }
947 }
948 }
949 }
950 }
951 }
952 }
953 }
954 }
955 }
956 }
957 }
958 }
959 }
960 }
961 }
962 }
963 }
964 }
965 }
966 }
967 }
968 }
969 }
970 }
971 }
972 }
973 }
974 }
975 }
976 }
977 }
978 }
979 }
980 }
981 }
982 }
983 }
984 }
985 }
986 }
987 }
988 }
989 }
990 }
991 }
992 }
993 }
994 }
995 }
996 }
997 }
998 }
999 }
1000 }

```

Figura 95: Adjuntar información a Firebase Storage



```

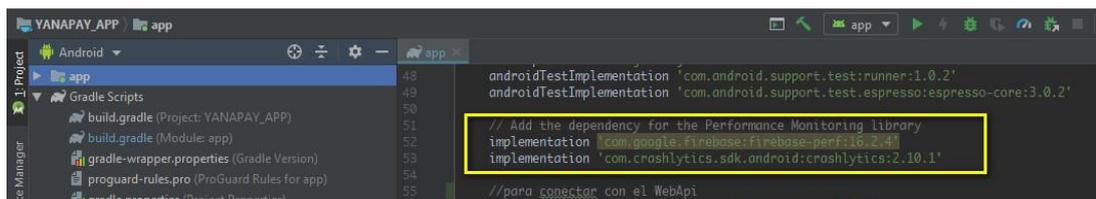
1 package com.codepaper.yanapayapp.Util;
2
3 import ...
4
5 public class FirebaseUtil {
6
7     @AddTrace(name = "downloadFile")
8     public static void downloadFile(Context context, String fileName, String directorioDestino, String url_) {
9
10         DownloadManager downloadManager = (DownloadManager) context.getSystemService(DOWNLOAD_SERVICE);
11         Uri uri_ = Uri.parse(url_);
12         DownloadManager.Request request = new DownloadManager.Request(uri_);
13
14         request.setNotificationVisibility(DownloadManager.Request.VISIBILITY_VISIBLE_NOTIFY_COMPLETED);
15         request.setDestinationInExternalFilesDir(context, directorioDestino, fileName);
16
17         downloadManager.enqueue(request);
18     }
19 }
20
21 }
22 }
23 }
24 }
25 }
26 }
27 }
28 }
29 }
30 }
31 }
32 }
33 }
34 }
35 }
36 }
37 }
38 }
39 }
40 }
41 }
42 }
43 }
44 }
45 }
46 }
47 }
48 }
49 }
50 }
51 }
52 }
53 }
54 }
55 }
56 }
57 }
58 }
59 }
60 }
61 }
62 }
63 }
64 }
65 }
66 }
67 }
68 }
69 }
70 }
71 }
72 }
73 }
74 }
75 }
76 }
77 }
78 }
79 }
80 }
81 }
82 }
83 }
84 }
85 }
86 }
87 }
88 }
89 }
90 }
91 }
92 }
93 }
94 }
95 }
96 }
97 }
98 }
99 }
100 }

```

Figura 96: Descargar información de Firebase Storage

3.5. Firebase Performance

Esta librería que sirve para recopilar información concerniente al rendimiento de la aplicación, esta información es utilizada para comprobar los indicadores de eficiencia del aplicativo móvil.

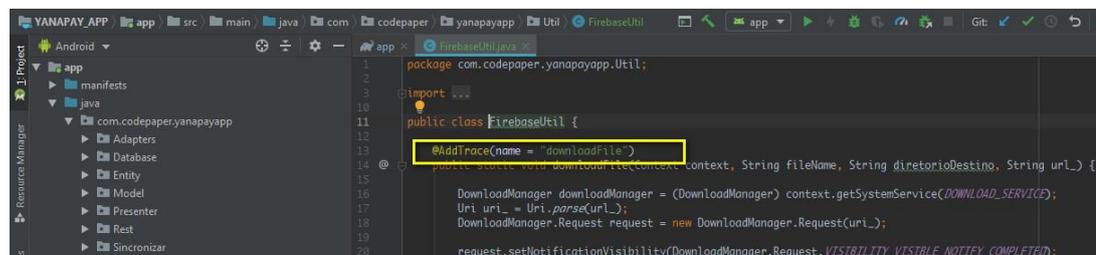


```

48 androidTestImplementation 'com.android.support.test:runner:1.0.2'
49 androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'
50
51 // Add the dependency for the Performance Monitoring library
52 implementation 'com.google.firebase:firebase-perf:10.2.4'
53 implementation 'com.crashlytics.sdk.android:crashlytics:2.10.1'
54
55 //para conectar con el WebApi

```

Figura 97: SDK Firebase Performance



```

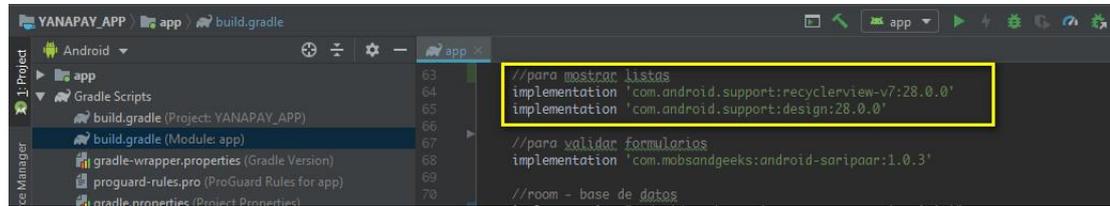
1 package com.codepaper.yanapayapp.Util;
2
3 import ...
4
5 public class FirebaseUtil {
6
7     @AddTrace(name = "downloadFile")
8     public static void downloadFile(Context context, String fileName, String directorioDestino, String url_) {
9
10         DownloadManager downloadManager = (DownloadManager) context.getSystemService(DOWNLOAD_SERVICE);
11         Uri uri_ = Uri.parse(url_);
12         DownloadManager.Request request = new DownloadManager.Request(uri_);
13
14         request.setNotificationVisibility(DownloadManager.Request.VISIBILITY_VISIBLE_NOTIFY_COMPLETED);
15
16         downloadManager.enqueue(request);
17     }
18 }
19
20 }
21 }
22 }
23 }
24 }
25 }
26 }
27 }
28 }
29 }
30 }
31 }
32 }
33 }
34 }
35 }
36 }
37 }
38 }
39 }
40 }
41 }
42 }
43 }
44 }
45 }
46 }
47 }
48 }
49 }
50 }
51 }
52 }
53 }
54 }
55 }
56 }
57 }
58 }
59 }
60 }
61 }
62 }
63 }
64 }
65 }
66 }
67 }
68 }
69 }
70 }
71 }
72 }
73 }
74 }
75 }
76 }
77 }
78 }
79 }
80 }
81 }
82 }
83 }
84 }
85 }
86 }
87 }
88 }
89 }
90 }
91 }
92 }
93 }
94 }
95 }
96 }
97 }
98 }
99 }
100 }

```

Figura 98: Etiquetas de Firebase Performance

3.6. RecyclerView

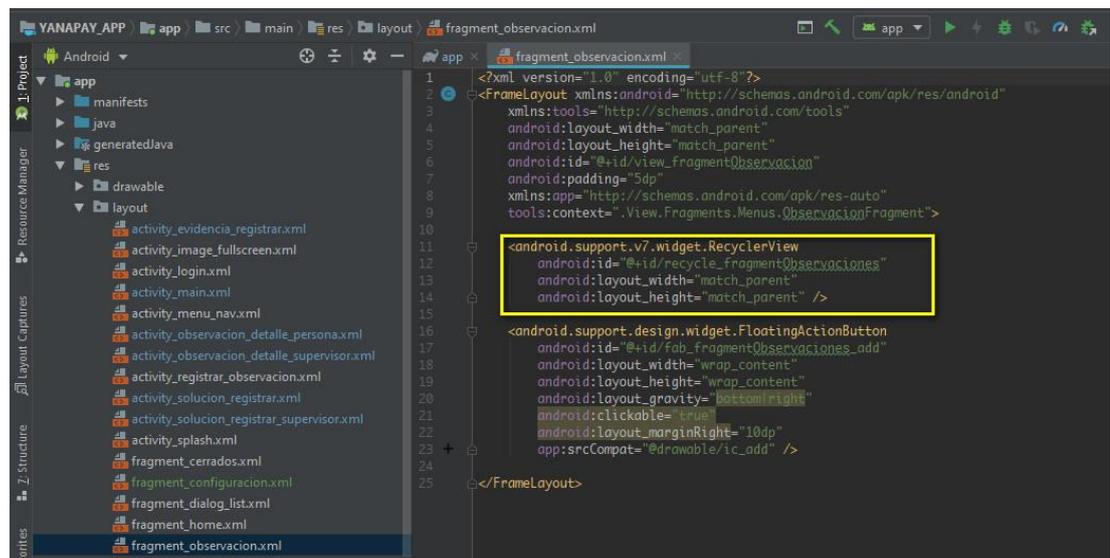
Se utiliza esta librería para mostrar la información de las observaciones en una lista de manera vertical.



```

63
64 //para mostrar listas
65 implementation 'com.android.support:recyclerview-v7:28.0.0'
66 implementation 'com.android.support:design:28.0.0'
67
68 //para validar formularios
69 implementation 'com.mabsandgeeks:android-saripaar:1.0.3'
70
71 //room - base de datos
  
```

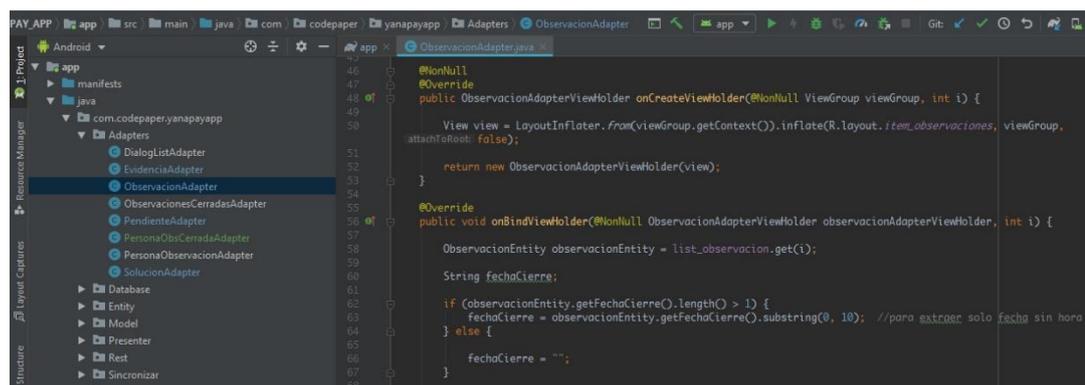
Figura 99: SDK RecyclerView



```

1 <?xml version="1.0" encoding="utf-8"?>
2 <FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
3   xmlns:tools="http://schemas.android.com/tools"
4   android:layout_width="match_parent"
5   android:layout_height="match_parent"
6   android:id="@+id/view_fragmentObservacion"
7   android:padding="5dp"
8   xmlns:app="http://schemas.android.com/apk/res-auto"
9   tools:context=".View.Fragments.Menus.ObservacionFragment">
10
11
12 <android.support.v7.widget.RecyclerView
13   android:id="@+id/recycle_fragmentObservaciones"
14   android:layout_width="match_parent"
15   android:layout_height="match_parent" />
16
17 <android.support.design.widget.FloatingActionButton
18   android:id="@+id/fab_fragmentObservaciones_add"
19   android:layout_width="wrap_content"
20   android:layout_height="wrap_content"
21   android:layout_gravity="bottomRight"
22   android:clickable="true"
23   android:layout_marginRight="10dp"
24   app:srcCompat="@drawable/ic_add" />
25 </FrameLayout>
  
```

Figura 100: RecyclerView implementado en una vista



```

46 @NonNull
47 @Override
48 public ObservacionAdapterViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int i) {
49     View view = LayoutInflater.from(viewGroup.getContext()).inflate(R.layout.item_observaciones, viewGroup,
50     attachToRoot: false);
51     return new ObservacionAdapterViewHolder(view);
52 }
53
54 @Override
55 public void onBindViewHolder(@NonNull ObservacionAdapterViewHolder observacionAdapterViewHolder, int i) {
56     ObservacionEntity observacionEntity = list_observacion.get(i);
57     String fechaCierre;
58     if (observacionEntity.getFechaCierre().length() > 1) {
59         fechaCierre = observacionEntity.getFechaCierre().substring(0, 10); //para extraer solo fecha sin hora
60     } else {
61         fechaCierre = "";
62     }
63 }
64 }
  
```

Figura 101: RecyclerView implementado por medio del patrón ViewHolder

3.7. Saripaar

Librería que se utiliza para validar que las cajas de texto de los formularios sean completadas antes de guardar la información en la base de datos.

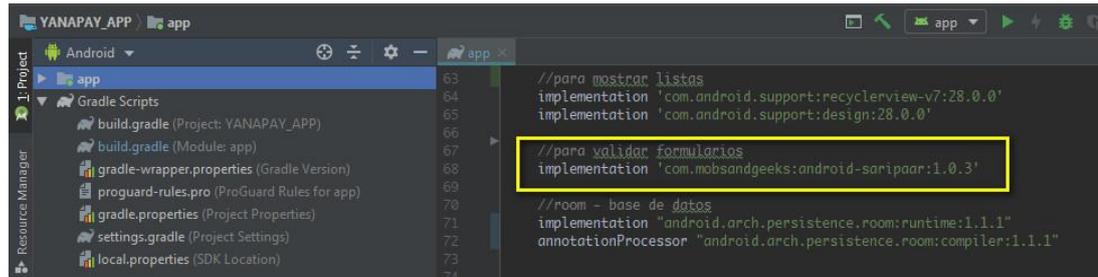


Figura 102: SDK Saripaar

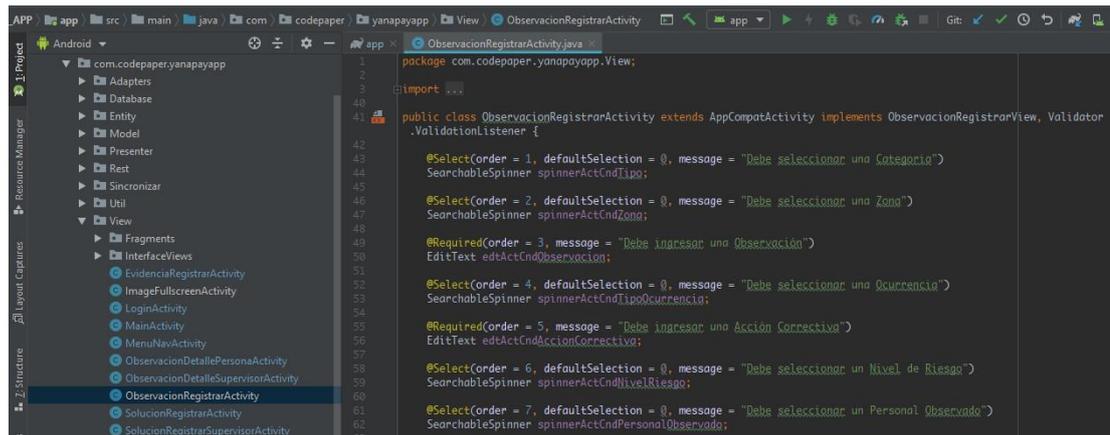


Figura 103: Etiquetas de validación Saripaar

3.8. SearchableSpinner

Librería que se utiliza para mostrar las listas seleccionables, a su vez permite buscar los elementos cargados en la lista para que sean más fáciles de ubicar.

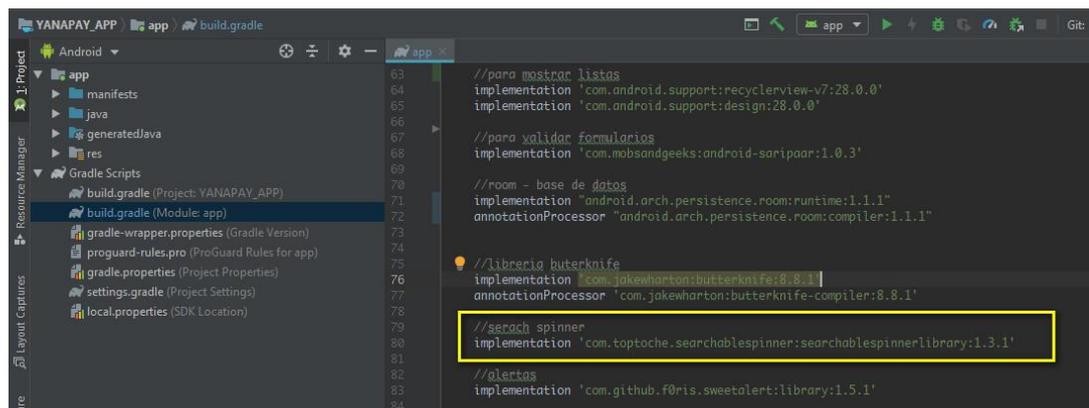
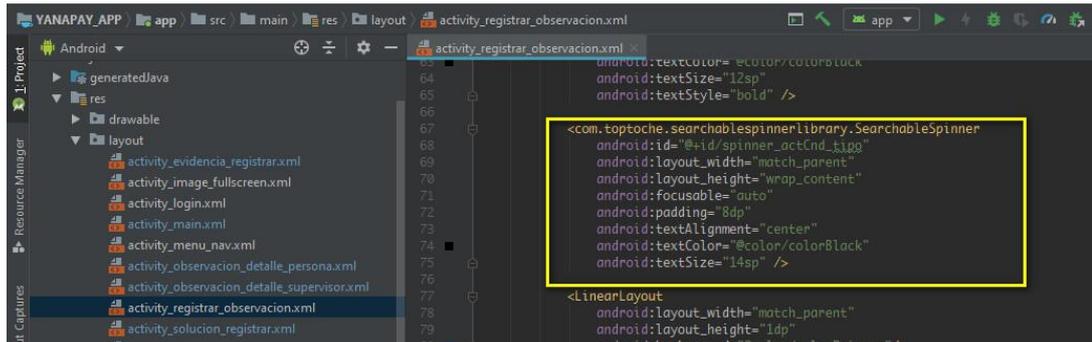


Figura 104: SDK SearchableSpinner



```

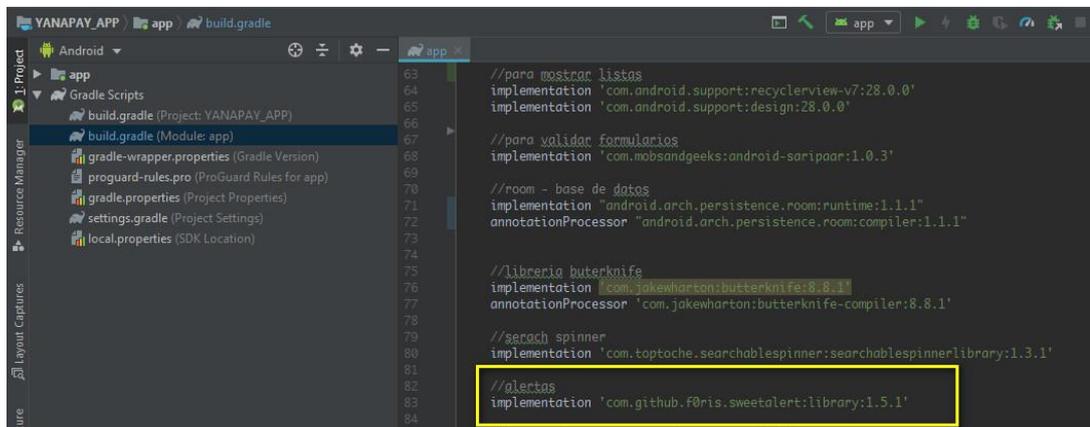
64 android:textColor="@color/colorBlack"
65 android:textSize="12sp"
66 android:textStyle="bold" />
67
68 <com.tptoch.e.searchablespinnerlibrary.SearchableSpinner
69     android:id="@+id/spinner_actCnd_tigg"
70     android:layout_width="match_parent"
71     android:layout_height="wrap_content"
72     android:focusable="auto"
73     android:padding="8dp"
74     android:textAlignment="center"
75     android:textColor="@color/colorBlack"
76     android:textSize="14sp" />
77
78 <LinearLayout
79     android:layout_width="match_parent"
80     android:layout_height="1dp"

```

Figura 105: SearchableSpinner implementado para mostrar los tipos de observaciones

3.9. Sweetalert

Librería utilizada para mostrar los mensajes de alerta de una manera más amigable para el usuario.

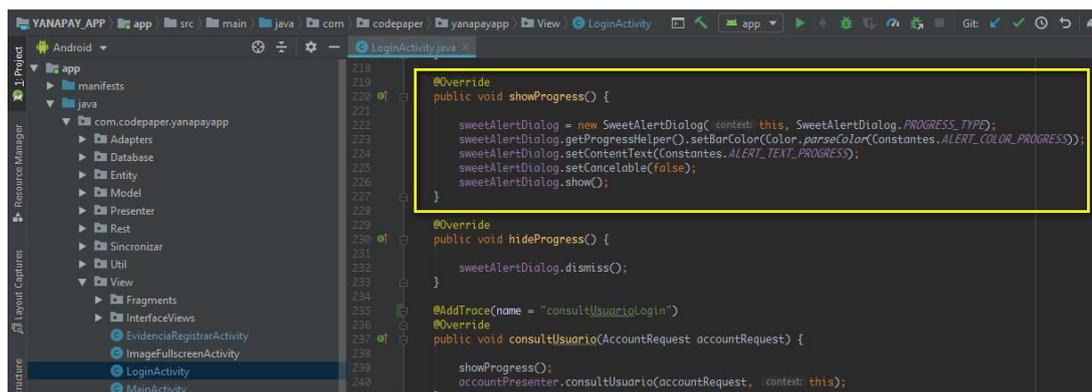


```

63 //para mostrar listas
64 implementation 'com.android.support:recyclerview-v7:28.0.0'
65 implementation 'com.android.support:design:28.0.0'
66
67 //para validar formularios
68 implementation 'com.mobsandgeeks:android-saripaar:1.0.3'
69
70 //room - base de datos
71 implementation 'android.arch.persistence.room:runtime:1.1.1'
72 annotationProcessor 'android.arch.persistence.room:compiler:1.1.1'
73
74
75 //libreria butterknife
76 implementation 'com.jakewharton:butterknife:8.8.1'
77 annotationProcessor 'com.jakewharton:butterknife-compiler:8.8.1'
78
79 //search spinner
80 implementation 'com.tptoch.e.searchablespinner:searchablespinnerlibrary:1.3.1'
81
82 //alertas
83 implementation 'com.github.f0ris.sweetalert:library:1.5.1'
84

```

Figura 106: SDK SweetAlert



```

218
219
220 @Override
221 public void showProgress() {
222     sweetAlertDialog = new SweetAlertDialog(context, SweetAlertDialog.PROGRESS_TYPE);
223     sweetAlertDialog.getProgressHelper().setBarColor(Color.parseColor(Constants.ALERT_COLOR_PROGRESS));
224     sweetAlertDialog.setContentText(Constants.ALERT_TEXT_PROGRESS);
225     sweetAlertDialog.setCancelable(false);
226     sweetAlertDialog.show();
227 }
228
229 @Override
230 public void hideProgress() {
231     sweetAlertDialog.dismiss();
232 }
233
234 @AddTrace(name = "consultUsuarioLogin")
235 @Override
236 public void consultUsuario(AccountRequest accountRequest) {
237     showProgress();
238     accountPresenter.consultUsuario(accountRequest, context, this);
239 }
240
241

```

Figura 107: SweetAlert implementada para mostrar una alerta de progreso

Anexo N° 14. Configurar la Biblioteca de Persistencia de datos ROOM

CONFIGURAR ROOM EN YANAPAY

ALCANCE

Este documento describe los pasos realizados para configurar la biblioteca de persistencia de datos Room al aplicativo móvil Yanapay.

Room, es una biblioteca de persistencia de datos que brinda la posibilidad de acceder a la base de datos local de manera rápida y sencilla sin generar problemas.

Así mismo esta biblioteca ayudará a crear una memoria caché de los datos en el dispositivo móvil donde se ejecute la aplicación, este archivo funcionará como una base de datos local, permitiendo el funcionamiento en modo offline del aplicativo Yanapay.

ROOM

A continuación, se detallan los pasos que se realizaron para incorporar la biblioteca Room al aplicativo Yanapay.

1. Prerrequisitos

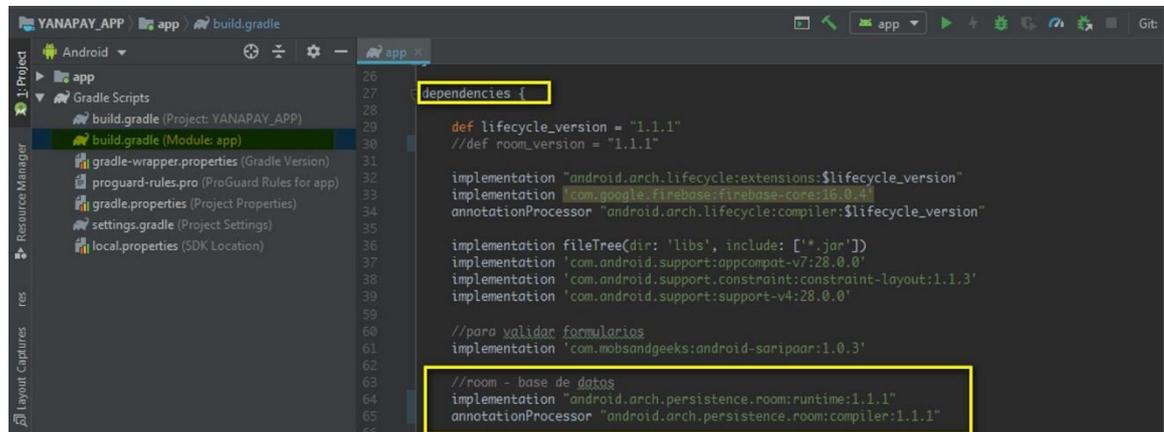
- Tener el código fuente del aplicativo móvil al cual se incorporará Room.

2. Incorporar Room

Para incorporar esta biblioteca se realizaron los siguientes pasos:

2.1. Agregar el SDK de Room

Ubicarse en el archivo Gradle a nivel de la app y agregar las dependencias de Room.



```

dependencies {
    def lifecycle_version = "1.1.1"
    //def room_version = "1.1.1"

    implementation "android.arch.lifecycle:extensions:$lifecycle_version"
    implementation "com.google.firebase:firebase-core:16.0.4"
    annotationProcessor "android.arch.lifecycle:compiler:$lifecycle_version"

    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation 'com.android.support:appcompat-v7:28.0.0'
    implementation 'com.android.support.constraint:constraint-layout:1.1.3'
    implementation 'com.android.support:support-v4:28.0.0'

    //para validar formularios
    implementation 'com.mobsandgeeks:android-saripaar:1.0.3'

    //room - base de datos
    implementation "android.arch.persistence.room:runtime:1.1.1"
    annotationProcessor "android.arch.persistence.room:compiler:1.1.1"
  }
  
```

Figura 108: Dependencias para incorporar Room

2.2. Crear clases para la base de datos local

Estas clases serán utilizadas para mapear y crear las tablas que se guardaran en la base de datos local, permite guardar información necesaria para que el aplicativo funcione en modo offline; para ello se creó un paquete con el nombre Room dentro del paquete Entity.

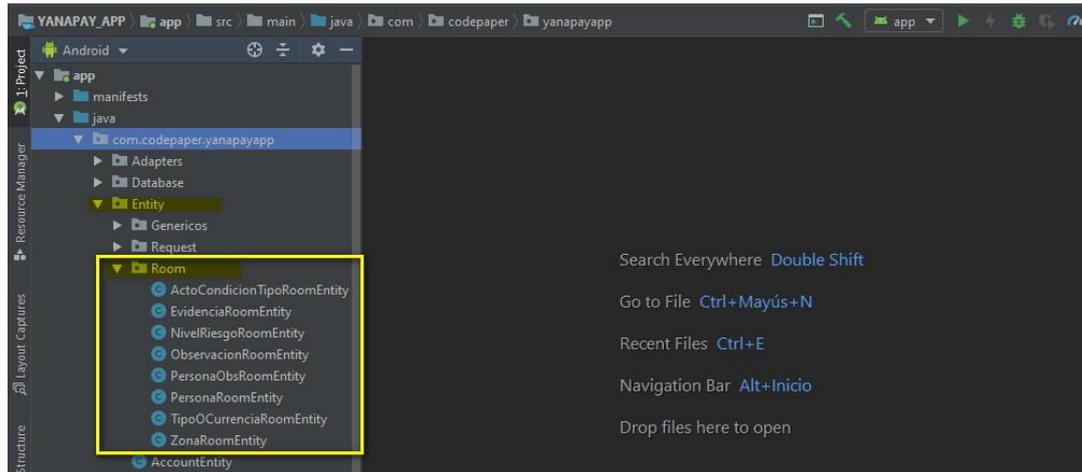


Figura 109: Clases para las tablas de la base datos local

Cada una de estas clases cuentan con atributos necesarios para guardar la información que se registre en el modo offline.

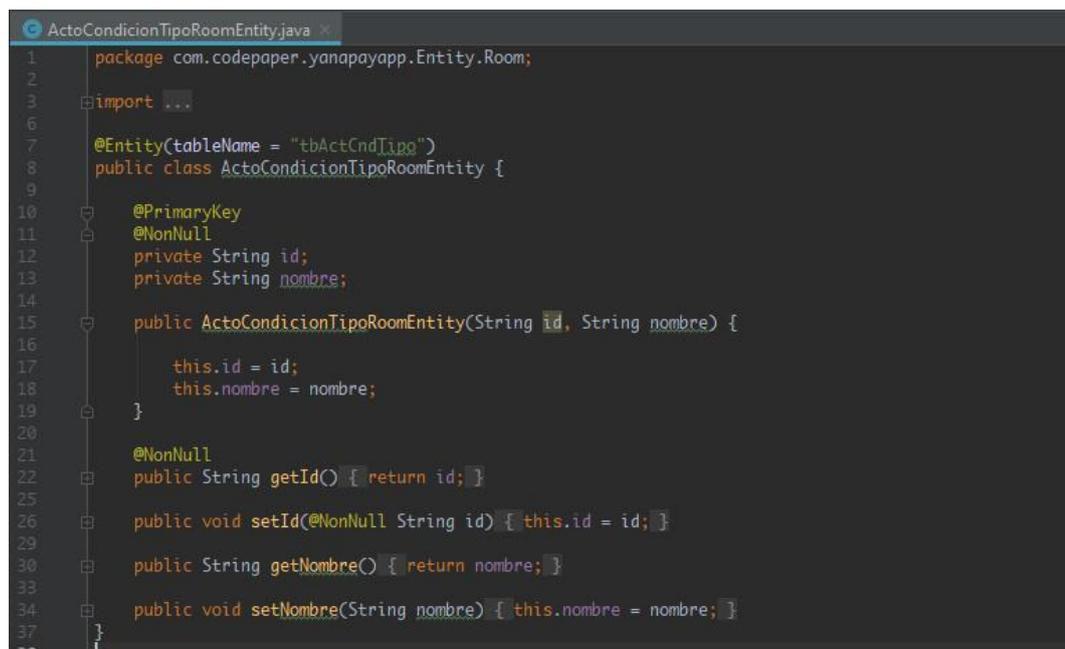


Figura 110: Clase ActoCondiciónRoomEntity

```

EvidenciaRoomEntity.java
1  package com.codepaper.yanapayapp.Entity.Room;
2
3  import ...
9
10 @Entity(tableName = "tbEvidencia")
11 public class EvidenciaRoomEntity {
12
13     @PrimaryKey
14     @NonNull
15     private String id;
16     private String observacionId;
17     private String path;
18     private String fileName;
19     private boolean preview;
20     private String linkDescarga;
21     private String observacionIdNet;
22     private int isSynchronize;
23
24     @NonNull
25     public String getId() { return id; }
28
29     public void setId(@NonNull String id) { this.id = id; }
32
33     public String getObservacionId() { return observacionId; }
36
37     public void setObservacionId(String observacionId) { this.observacionId = observacionId; }
40
41     public String getPath() { return path; }
44
45     public void setPath(String path) { this.path = path; }

```

Figura 111: Clase EvidenciaRoomEntity

```

NivelRiesgoRoomEntity.java
1  package com.codepaper.yanapayapp.Entity.Room;
2
3  import ...
6
7  @Entity(tableName = "tbNivelRiesgo")
8  public class NivelRiesgoRoomEntity {
9
10     @PrimaryKey
11     @NonNull
12     private String id;
13     private String nombre;
14     private String descripcion;
15
16
17     public NivelRiesgoRoomEntity(String id, String nombre, String descripcion) {
18
19         this.id = id;
20         this.nombre = nombre;
21         this.descripcion = descripcion;
22     }
23
24     @NonNull
25     public String getId() { return id; }
28
29     public void setId(@NonNull String id) { this.id = id; }
32
33     public String getNombre() { return nombre; }
36
37     public void setNombre(String nombre) { this.nombre = nombre; }
40

```

Figura 112: Clase NivelRiesgoRoomEntity

```

ObservacionRoomEntity.java
1 package com.codepaper.yanapayapp.Entity.Room;
2
3 import ...
4
5
6
7 @Entity(tableName = "tbObservacion")
8 public class ObservacionRoomEntity {
9
10     @PrimaryKey
11     @NonNull
12     private String Id;
13     private String Observacion;
14     private String Comentario;
15     private String AccionCorrectiva;
16     private int CantidadDias;
17     private String TipoOcurrenciaId;
18     private String ZonaId;
19     private String ObservacionTipo;
20     private String NivelRiesgoId;
21     private String ObservacionIdNet;
22     private int isSynchronize;
23
24     @NonNull
25     public String getId() { return Id; }
26
27
28
29     public void setId(@NonNull String id) { Id = id; }
30
31
32
33     public String getObservacion() { return Observacion; }
34
35
36
37     public void setObservacion(String observacion) { Observacion = observacion; }
38
39
40

```

Figura 113: Clase ObservacionRoomEntity

```

PersonaObsRoomEntity.java
1 package com.codepaper.yanapayapp.Entity.Room;
2
3 import ...
4
5
6
7
8
9 @Entity(tableName = "tbPersonaObservacion")
10 public class PersonaObsRoomEntity {
11
12     @PrimaryKey
13     @NonNull
14     private String Id;
15     private String IdPersona;
16     private String observacionId;
17
18     public PersonaObsRoomEntity(String Id, String IdPersona, String observacionId) {
19         this.Id = Id;
20         this.IdPersona = IdPersona;
21         this.observacionId = observacionId;
22     }
23
24     @NonNull
25     public String getId() { return Id; }
26
27
28
29     public void setId(@NonNull String id) { Id = id; }
30
31
32
33     public String getIdPersona() { return IdPersona; }
34
35
36
37     public void setIdPersona(String idPersona) { IdPersona = idPersona; }
38
39
40

```

Figura 114: Clase PersonaObsRoomEntity

```
PersonaRoomEntity.java
1 package com.codepaper.yanapayapp.Entity.Room;
2
3 import ...
4
5
6
7 @Entity(tableName = "tbPersona")
8 public class PersonaRoomEntity {
9
10     @PrimaryKey
11     @NonNull
12     private String id;
13
14     private String nombre;
15
16     public PersonaRoomEntity(@NonNull String id, String nombre) {
17
18         this.id = id;
19         this.nombre = nombre;
20     }
21
22     public String getId() { return id; }
23
24
25     public void setId(String id) { this.id = id; }
26
27
28     public String getNombre() { return nombre; }
29
30
31     public void setNombre(String nombre) { this.nombre = nombre; }
32
33
34
35
36
37
38 }
```

Figura 115: Clase PersonaRoomEntity

```
TipoOcurranciaRoomEntity.java
1 package com.codepaper.yanapayapp.Entity.Room;
2
3 import ...
4
5
6
7 @Entity(tableName = "tbTipoOcurrancia")
8 public class TipoOcurranciaRoomEntity {
9
10     @PrimaryKey
11     @NonNull
12     private String id;
13
14     private String nombre;
15     private String actCndTipoId;
16
17     public TipoOcurranciaRoomEntity(@NonNull String id, String nombre, String actCndTipoId) {
18
19         this.id = id;
20         this.nombre = nombre;
21         this.actCndTipoId = actCndTipoId;
22     }
23
24
25     @NonNull
26     public String getId() { return id; }
27
28
29     public void setId(@NonNull String id) { this.id = id; }
30
31
32     public String getNombre() { return nombre; }
33
34
35     public void setNombre(String nombre) { this.nombre = nombre; }
36
37
38 }
```

Figura 116: Clase TipoOcurranciaRoomEntity

```

1  package com.codepaper.yanapayapp.Entity.Room;
2
3  import ...
4
5
6
7  @Entity(tableName = "tbZona")
8  public class ZonaRoomEntity {
9
10     @PrimaryKey
11     @NonNull
12     private String zonaId;
13
14     private String nombre;
15     private String proyectoId;
16
17     public ZonaRoomEntity(@NonNull String zonaId, String nombre, String proyectoId) {
18
19         this.zonaId = zonaId;
20         this.nombre = nombre;
21         this.proyectoId = proyectoId;
22     }
23
24     @NonNull
25     public String getZonaId() { return zonaId; }
26
27
28     public void setZonaId(@NonNull String zonaId) { this.zonaId = zonaId; }
29
30
31     public String getNombre() { return nombre; }
32
33
34     public void setNombre(String nombre) { this.nombre = nombre; }
35
36
37
38
39
40

```

Figura 117: Clase ZonaRoomEntity

2.3. Crear interfaces para procesar la información

Estas interfaces serán utilizadas para procesar la información hacia la base de datos local, permitirán manipular la información por medio de etiquetas Room y sentencias SQL cuando el aplicativo funcione en modo offline; para ello se creó un paquete con el nombre Dao dentro del paquete Database.

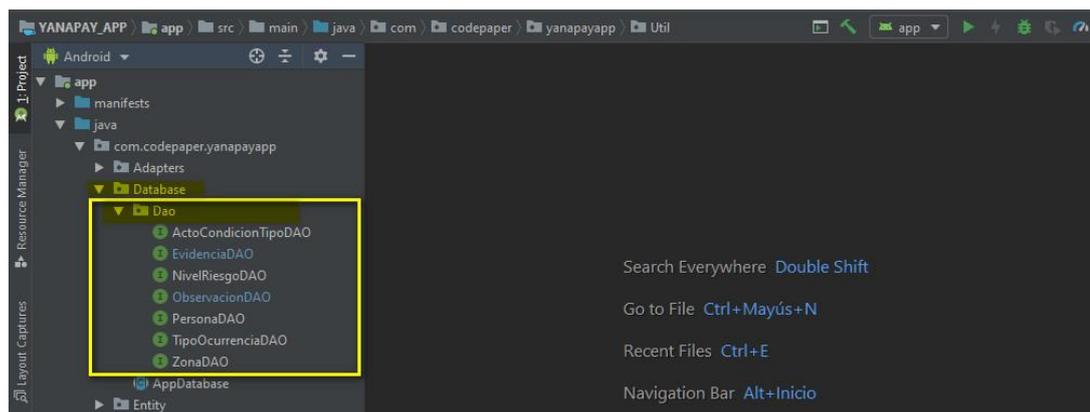


Figura 118: Interfaces para procesar la información de la base datos local

Cada una de estas interfaces cuentan con etiquetas Room y sentencias SQL necesarias para manipular la información de la base de datos local.

```

ActoCondicionTipoDAO.java
1 package com.codepaper.yanapayapp.Database.Dao;
2
3 import ...
10
11 @Dao
12 public interface ActoCondicionTipoDAO {
13
14     @Insert
15     void insert(ActoCondicionTipoRoomEntity actoCondicionTipoRoomEntity);
16
17     @Query("select * from tbActCndTipo")
18     List<ActoCondicionTipoRoomEntity> getAllActoCondicionTippo();
19
20     @Query("select distinct a.nombre " +
21            "from tbActCndTipo a join tbTipOcurrancia b on a.id = b.actCndTipoId " +
22            "where b.id =:tipoOcurranciaId")
23     String getActoCondicionByTipoOcurranciaId(String tipoOcurranciaId);
24 }
    
```

Figura 119: Interfaz ActoCondicionTipoDAO

```

EvidenciaDAO.java
1 package com.codepaper.yanapayapp.Database.Dao;
2
3 import ...
12
13 @Dao
14 public interface EvidenciaDAO {
15
16     @Insert
17     void insert(EvidenciaRoomEntity evidenciaRoomEntity);
18
19     @Update
20     void update(EvidenciaRoomEntity evidenciaRoomEntity);
21
22     //Evidencias que no han sido sincronizadas
23     @Query("select * from tbEvidencia where isSynchronize = 0 and observacionIdNet =:idObservacionNet")
24     List<EvidenciaRoomEntity> getAllEvidenciasIsNotSynchronized(String idObservacionNet);
25
26     //Evidencias que ya han sido sincronizadas
27     @Query("select * from tbEvidencia where isSynchronize = 1 and observacionIdNet =:idObservacionNet")
28     List<EvidenciaRoomEntity> getAllEvidenciasIsSynchronized(String idObservacionNet);
29
30     @Query("select * from tbEvidencia where observacionId =:s_observacionId")
31     List<EvidenciaRoomEntity> getAllEvidenciasByObservacionId(String s_observacionId);
32
33     //actualizamos idObservacionesNet de las evidencias relacionadas
34     @Query("update tbEvidencia " +
35            "set observacionIdNet =:idObservacionNet " +
36            "where observacionId =:idObservacion")
37     void updateEvidenciaObservacionIdNet(String idObservacion, String idObservacionNet);
38
39     @Query("update tbEvidencia " +
40            "set isSynchronize = 1 " +
41            "where observacionIdNet =:s_idObservacionNet")
42     void updateEvidenciasSynchronize(String s_idObservacionNet);
    
```

Figura 120: Interfaz EvidenciaDAO

```

1 package com.codepaper.yanapayapp.Database.Dao;
2
3 import ...
10
11 @Dao
12 public interface NivelRiesgoDAO {
13
14     @Insert
15     void insert(NivelRiesgoRoomEntity nivelRiesgoRoomEntity);
16
17     @Query("select * from tbNivelRiesgo")
18     List<NivelRiesgoRoomEntity> getAllNivelRiesgo();
19 }

```

Figura 121: Interfaz NivelRiesgoDAO

```

15 @Dao
16 public interface ObservacionDAO {
17
18     @Insert
19     void insert(ObservacionRoomEntity observacionRoomEntity);
20
21     @Insert
22     void insertPersonaObs(PersonaObsRoomEntity personaObsRoomEntity);
23
24     @Update
25     void update(ObservacionRoomEntity observacionRoomEntity);
26
27     //observaciones que no han sido sincronizadas
28     @Query("select * from tbObservacion where isSynchronize = 0")
29     List<ObservacionRoomEntity> getAllObservacionIsNotSynchronized();
30
31     //observaciones que ya han sido sincronizadas
32     @Query("select * from tbObservacion where isSynchronize = 1")
33     List<ObservacionRoomEntity> getAllObservacionIsSynchronized();
34
35     @Query("select * from tbObservacion")
36     List<ObservacionRoomEntity> getAllObservacion();
37
38     //@Query("select * from tbObservacion where Id =:id")
39     @Query("select o.Id " +
40         " ,o.observacion " +
41         " ,o.Comentario " +
42         " ,o.accionCorrectiva " +
43         " ,nr.descripcion as CantidadDias " +
44         " ,tpo.nombre as TipoOcurrenciaId " +
45         " ,z.nombre as ZonaId " +
46         " ,(case when o.observacionTipo = '01' then 'ACTO' else 'CONDICION' end) as ObservacionTipo " +
47         " ,nr.nombre as NivelRiesgoId " +
48         " ,o.ObservacionIdNet " +
49

```

Figura 122: Interfaz ObservacionDAO

```

1 package com.codepaper.yanapayapp.Database.Dao;
2
3 import ...
10
11 @Dao
12 public interface PersonaDAO {
13
14     @Insert
15     void insert(PersonaRoomEntity personaRoomEntity);
16
17     @Query("select * from tbpersona")
18     List<PersonaRoomEntity> getAllPersonas();
19
20     //obtenemos personas reportadas en la observacion
21     @Query("select p.* " +
22         " from tbPersonaObservacion po join tbPersona p on po.IdPersona = p.id " +
23         " where po.observacionId =:id")
24     List<PersonaRoomEntity> getAllPersonasByObservacionId(String id);
25 }

```

Figura 123: Interfaz PersonaDAO

```

TipoOcurrenciaDAO.java
1 package com.codepaper.yanapayapp.Database.Dao;
2
3 import ...
10
11 @Dao
12 public interface TipoOcurrenciaDAO {
13
14     @Insert
15     void insert(TipoOcurrenciaRoomEntity tipoOcurrenciaRoomEntity);
16
17     @Query("select * from tbtipoocurrencia where actCndTipoId =:actCndTipoId")
18     List<TipoOcurrenciaRoomEntity> getAllTipoOcurrenciaByActoCondicionTipo(String actCndTipoId);
19 }
20
    
```

Figura 124: Interfaz TipoOcurrenciaDAO

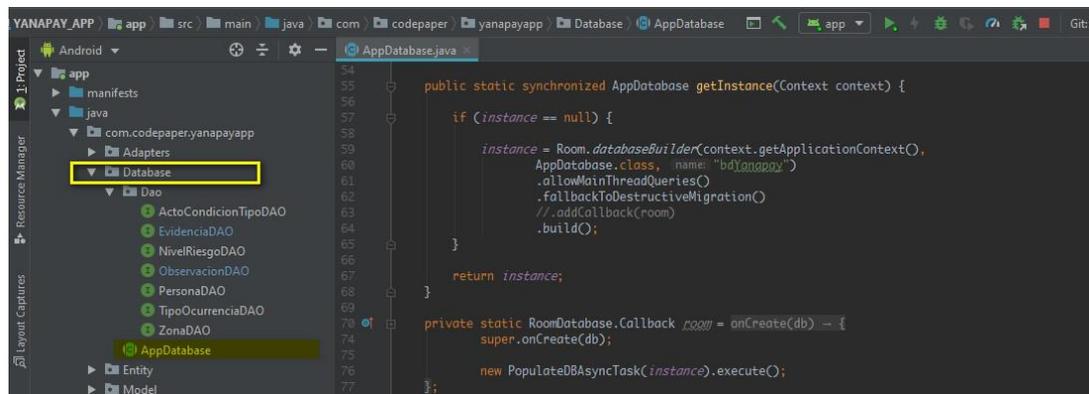
```

ZonaDAO.java
1 package com.codepaper.yanapayapp.Database.Dao;
2
3 import ...
10
11 @Dao
12 public interface ZonaDAO {
13
14     @Insert
15     void insert(ZonaRoomEntity zonaRoomEntity);
16
17     @Query("select * from tbZona where proyectoId =:proyectoId")
18     List<ZonaRoomEntity> getAllZonasByProyecto(String proyectoId);
19 }
20
    
```

Figura 125: Interfaz ZonaDAO

2.4. Crear base datos local

Para la instanciación de la base de datos local se creó una clase abstracta la cual se invoca al iniciar el aplicativo con la finalidad de sincronizar la información requerida por este para que funcione en modo offline, adicionalmente también podrá ser invocada cada vez que se desee manipular información de la base de datos local; para ello se creó la clase abstracta con el nombre AppDatabase dentro del paquete Database.

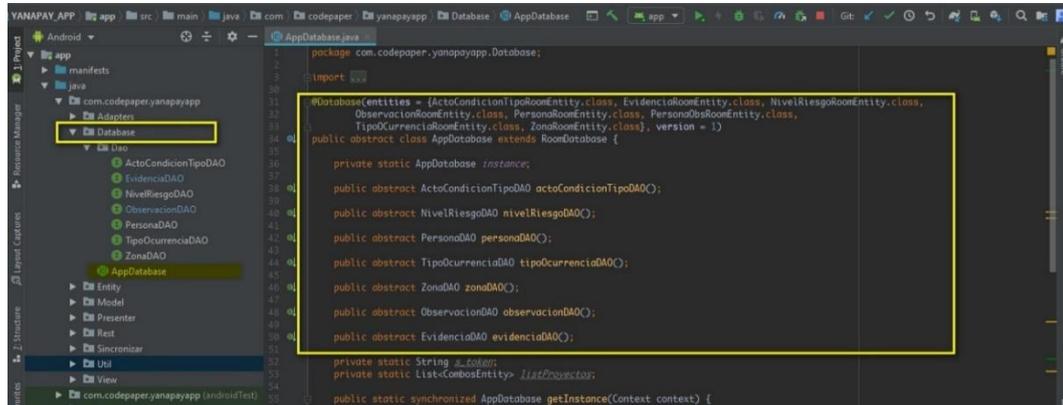


```

AppDatabase.java
54 public static synchronized AppDatabase getInstance(Context context) {
55
56     if (instance == null) {
57
58         instance = Room.databaseBuilder(context.getApplicationContext(),
59             AppDatabase.class, name = "bdYanapay")
60                 .allowMainThreadQueries()
61                 .fallbackToDestructiveMigration()
62                 // .addCallback(room)
63                 .build();
64     }
65
66     return instance;
67 }
68
69 private static RoomDatabase.Callback callback = onCreate(db) -> {
70     super.onCreate(db);
71
72     new PopulateDBAsyncTask(instance).execute();
73 };
74
75
76
77
    
```

Figura 126: Clase abstracta AppDatabase

Dentro de esta clase se invoca a las interfaces creadas en el paquete Dao, con la finalidad de acceder a las etiquetas Room y sentencias SQL.



```

package com.codepaper.yanapayapp.Database;

import androidx.room.Database;
import androidx.room.RoomDatabase;

@Database(entities = {ActoCondicionTipoRoomEntity.class, EvidenciaRoomEntity.class, NivelRiesgoRoomEntity.class,
    ObservacionRoomEntity.class, PersonaRoomEntity.class, PersonaObsRoomEntity.class,
    TipoOcurrenciaRoomEntity.class, ZonaRoomEntity.class}, version = 1)
public abstract class AppDatabase extends RoomDatabase {

    private static AppDatabase instance;

    public abstract ActoCondicionTipoDAO actoCondicionTipoDAO();

    public abstract NivelRiesgoDAO nivelRiesgoDAO();

    public abstract PersonaDAO personaDAO();

    public abstract TipoOcurrenciaDAO tipoOcurrenciaDAO();

    public abstract ZonaDAO zonaDAO();

    public abstract ObservacionDAO observacionDAO();

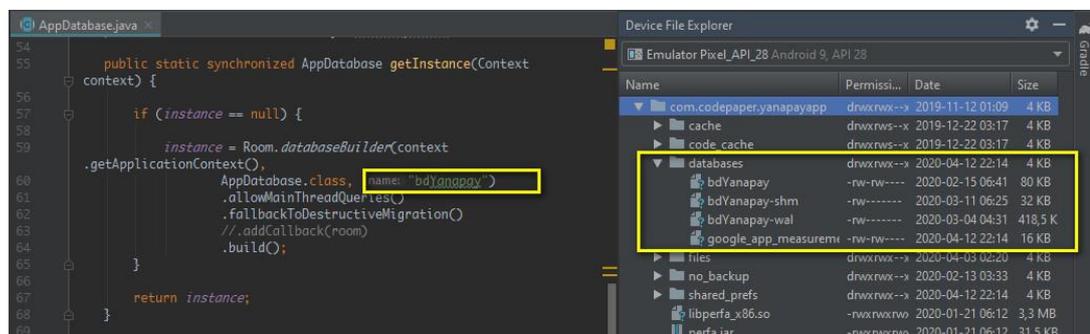
    public abstract EvidenciaDAO evidenciaDAO();

    private static String s_token;
    private static List<ComposEntity> listFactores;

    public static synchronized AppDatabase getInstance(Context context) {
    
```

Figura 127: Interfaces invocadas

Finalmente se podrá comprobar que la base de datos se crea en la memoria caché del dispositivo móvil, para ello es necesario haber iniciado el aplicativo móvil y luego ingresar con las credenciales de inicio de sesión proporcionadas por el administrador.



```

54
55 public static synchronized AppDatabase getInstance(Context
56 context) {
57     if (instance == null) {
58         instance = Room.databaseBuilder(context
59             .getApplicationContext(),
60             AppDatabase.class, "bdYanapay")
61             .allowMainThreadQueries()
62             .fallbackToDestructiveMigration()
63             // .addCallback(room)
64             .build();
65     }
66     return instance;
67 }
68
69

```

Name	Permissi...	Date	Size
com.codepaper.yanapayapp	drwxrwx--x	2019-11-12 01:09	4 KB
cache	drwxrws--x	2019-12-22 03:17	4 KB
code_cache	drwxrws--x	2019-12-22 03:17	4 KB
databases	drwxrwx--x	2020-04-12 22:14	4 KB
bdYanapay	-rw-rw----	2020-02-15 06:41	80 KB
bdYanapay-shm	-rw-----	2020-03-11 06:25	32 KB
bdYanapay-wal	-rw-----	2020-03-04 04:31	418,5 K
google_app_measurement	-rw-rw----	2020-04-12 22:14	16 KB
files	drwxrwx--x	2020-04-03 02:20	4 KB
no_backup	drwxrwx--x	2020-02-13 03:33	4 KB
shared_prefs	drwxrwx--x	2020-04-12 22:14	4 KB
libperfa_x86.so	-rw-rwxrwx	2020-01-21 06:12	3,3 MB
perfa.jar	-rw-rwxrwx	2020-01-21 06:12	31,5 KB

Figura 128: Base de datos local en memoria caché

Anexo N° 15. WEB API YANAPAY

WEB API YANAPAY

ALCANCE

Este documento brinda una descripción de la arquitectura utilizada para el desarrollo del API que utiliza la aplicación Yanapay así como otros aspectos que dan forma al diseño e implementación, explicando el diseño lógico y físico de la misma.

El desarrollo de esta interfaz de programación está orientado a cumplir con la actividad de reportar actos y condiciones observadas dentro de la operación minera por el personal del área de Seguridad y Salud Ocupacional y Medio Ambiente mientras realizan sus actividades de supervisión en campo.

Para el desarrollo de esta aplicación se tuvieron en consideración los requerimientos y exigencias identificadas en la etapa de análisis.

Diseño de la solución

1. Arquitectura

La aplicación será desarrollada e implementada bajo la plataforma .NET Core 3.0, teniendo en cuenta que la mayoría de aplicaciones .NET tradicionales se implementan como unidades únicas que corresponden a un archivo ejecutable o una sola aplicación web que se ejecuta dentro de un único dominio de aplicación de IIS.

La arquitectura implementada es una arquitectura en “N Capas” teniendo en cuenta satisfacer las siguientes características:

- Interoperabilidad: Capacidad para compartir información con otros sistemas.
- Escalabilidad: Capacidad para permitir en un futuro desarrollar, modificar o eliminar funcionalidades.
- Mantenibilidad: Capacidad donde no requiera de mucho esfuerzo para conservar su funcionamiento cuando se presente una falla.
- Seguridad: Capacidad de restringir el acceso a usuarios no autorizados.
- Disponibilidad: Capacidad de que el sistema esté total o parcialmente operativo.
- Tiempo de respuesta: Capacidad de responder a las peticiones que realizan los usuarios de manera inmediata.

1.1. Arquitectura de “N Capas”

Se utiliza esta arquitectura ya que permite poder administrar las aplicaciones dividiéndolas por responsabilidades y a su vez ayuda a mantener el código organizado para que se pueda realizar cualquier cambio de manera más rápida.



Figura 129: Arquitectura de N capas

Teniendo como referencia la arquitectura de “N capas”, se decidió que la aplicación esté compuesta por tres capas con estratificación flexible. Se decidió que este tipo de arquitectura es el más adecuado debido a que el diseño es más conocido entre los desarrolladores, más fácil de comprender y permite un mejor mantenimiento.

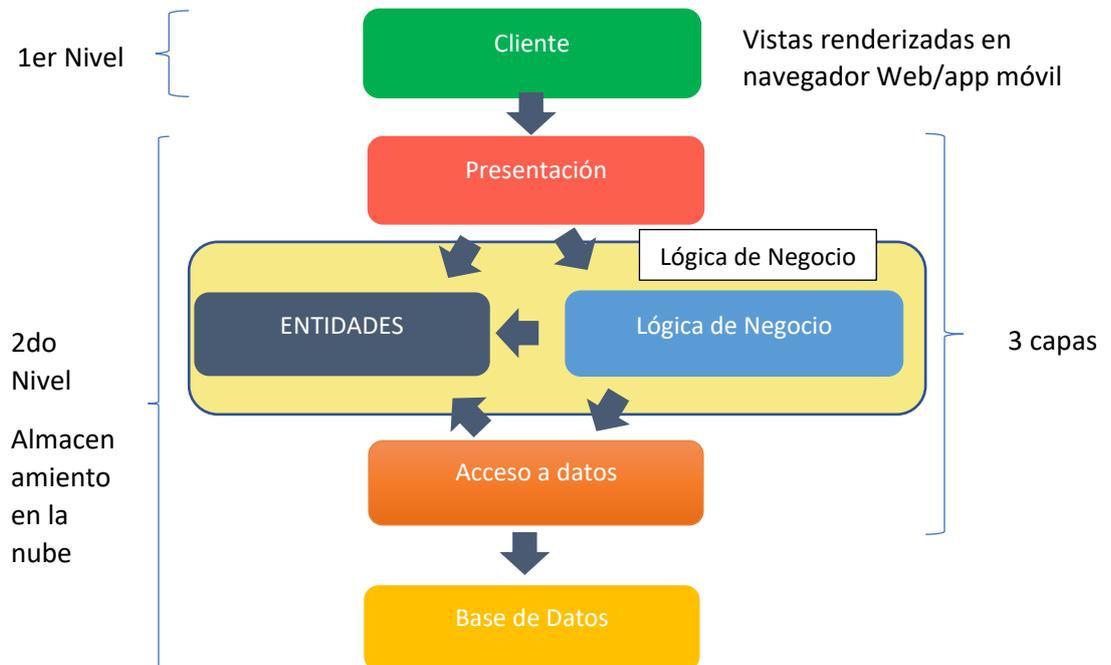


Figura 130: Arquitectura Yanapay

En la imagen Arquitectura Yanapay, se puede identificar los niveles de separación física (1er nivel y 2do nivel) y la separación lógica representada por la arquitectura de 3 capas.

La estructura en el IDE de desarrollo Visual Studio sería de la siguiente manera:

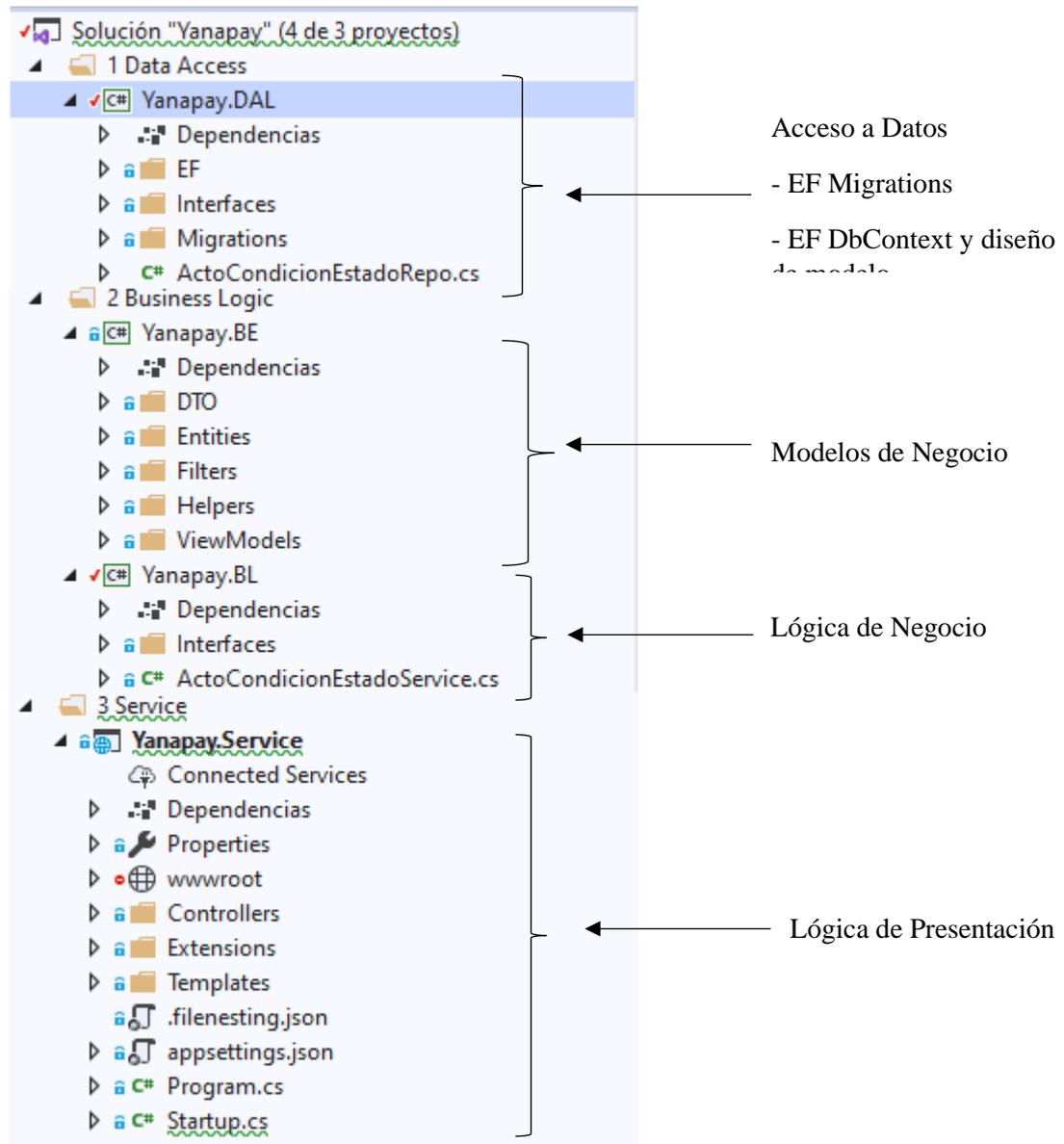


Figura 131: Estructura en Visual Studio

1.2. Capa Presentación

Es la interfaz de usuario de la aplicación, como una página web, CLI o una aplicación de escritorio; para Yanapay se ha considerado usar servicios RESTful conocidos como Web APIs y una aplicación web de página única (SPA) por los siguientes motivos:

- La aplicación tiene que exponer una interfaz de usuario enriquecida con muchas características.
- El equipo está familiarizado con el desarrollo de JavaScript o TypeScript.

- La aplicación tiene que exponer una API para otros clientes (internos o públicos).

1.3. Capa Lógica de Negocio

En esta capa o componente codificamos las reglas de negocio, la cual permite comunicar entre el acceso a datos y la lógica de presentación.

1.4. Capa Acceso a Datos

Esta capa permite almacenar u obtener datos desde la base de datos desplegada en Sql Server.

2. Diagrama de componentes

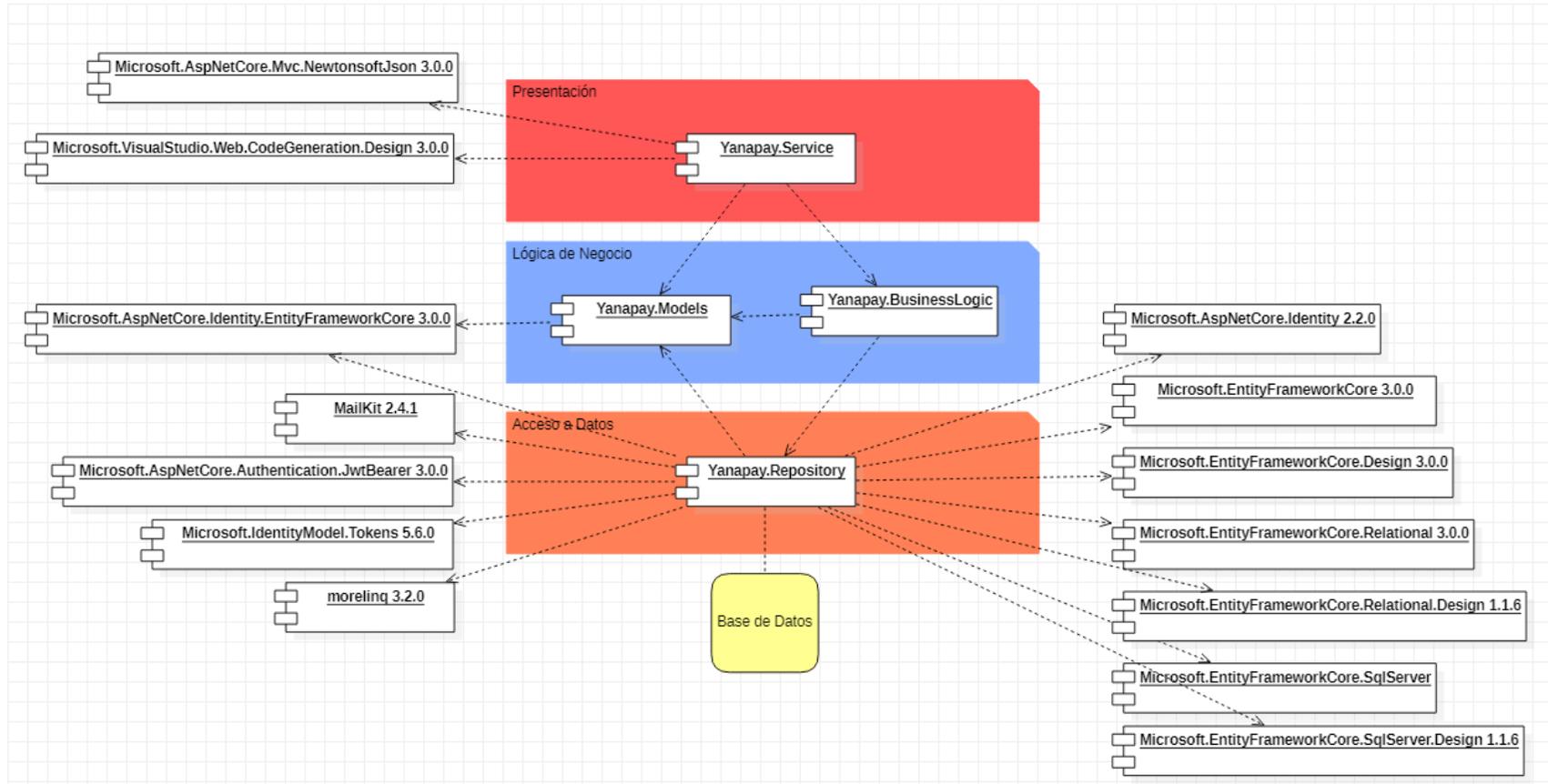


Figura 132: Diagrama de componentes Yanapay

Anexo N° 16. Base de datos Yanapay

BASE DE DATOS YANAPAY

ALCANCE

Este documento muestra el diseño relacional de la base de datos creada para el aplicativo Yanapay, dicha base de datos ha sido creada y desplegada en SQL Server 2016, esto con la finalidad de tener una mejor comunicación o convivencia con el Web API que comunica con el aplicativo móvil, este último también desarrollado con tecnología Microsoft.

1. SQL SERVER

Se considera utilizar Sql Server ya que, al ser un sistema de gestión de bases de datos relacional, facilita la creación de tablas y su relación entre ellas, al igual que la recuperación de los datos almacenados.

A continuación, se muestran las dos estructuras de tablas utilizadas tanto para la seguridad del aplicativo como para el registro de información:

1.1. Estructura de Seguridad

Estas tablas son utilizadas para almacenar y controlar la información de la seguridad en los inicios de sesión del aplicativo móvil.

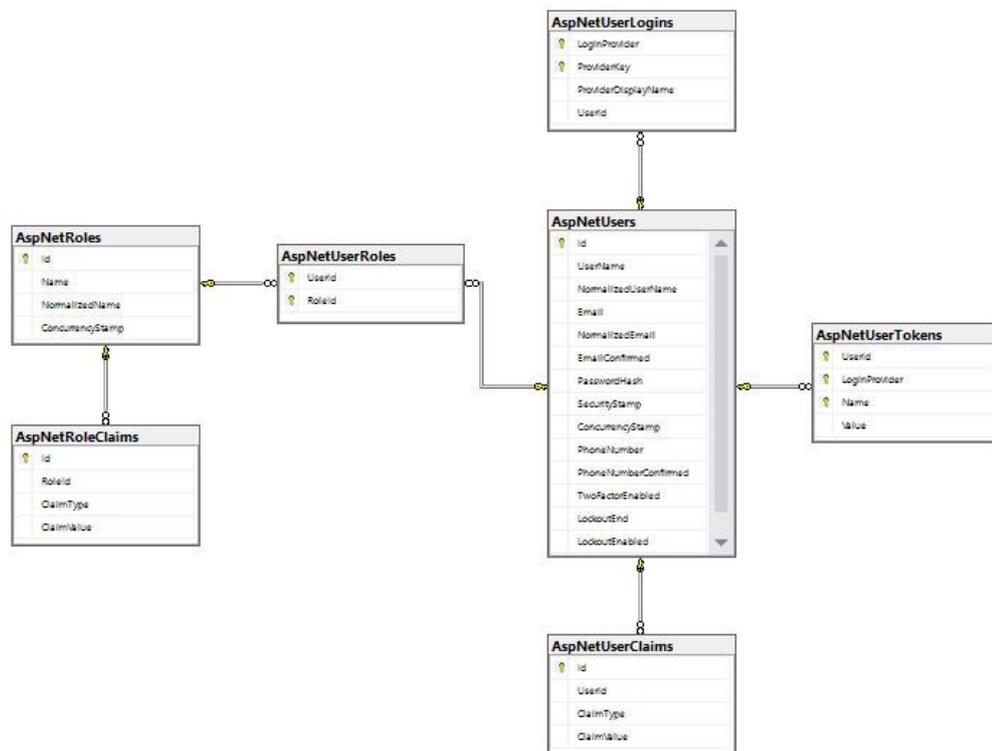


Figura 133: Tablas para la seguridad del aplicativo

1.2. Estructura para el procesamiento de información

Las tablas que a continuación se muestran son utilizadas para el registro de información que se realice desde el dispositivo móvil.

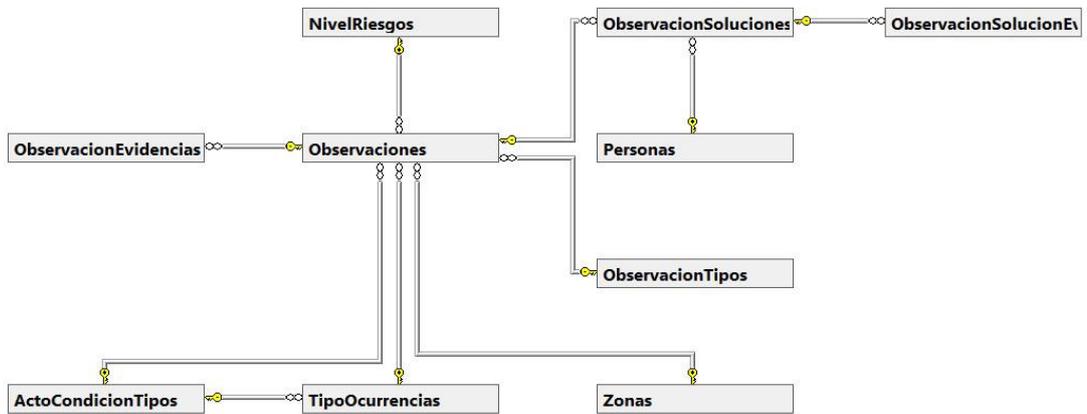


Figura 134: Tablas para el procesamiento de información

Anexo N° 17. Servidor Amazon LightSail

SERVIDOR AMAZON LIGHTSAIL

ALCANCE

Este documento muestra los servicios de Web API y Base de Datos desplegados en un servidor virtual de Amazaon LigthSail, este servicio brinda la posibilidad de virtualizar un servidor de Windows con un gestor de base de datos Sql Server 2016 los cuales son publicados para que el aplicativo móvil pueda consumirlos en cualquier momento del día.

1. AMAZON LIGHTSAIL

Se considera utilizar esta plataforma en la nube ya que ofrece crear un servidor virtual de Windows Server 2016 con acceso a escritorio remoto para una mejor configuración, en este servicio ha sido desplegada tanto la Web API como la base de datos.



Figura 135: Servidor creado en Amazon LightSail

A continuación, se muestran ambos servicios desplegados en el servidor virtual los cuales son consumidos por el aplicativo móvil en cualquier momento del día:

1.1. Servicio de base de datos

La base de datos se encuentra desplegada en SQL Server 2016 Express.

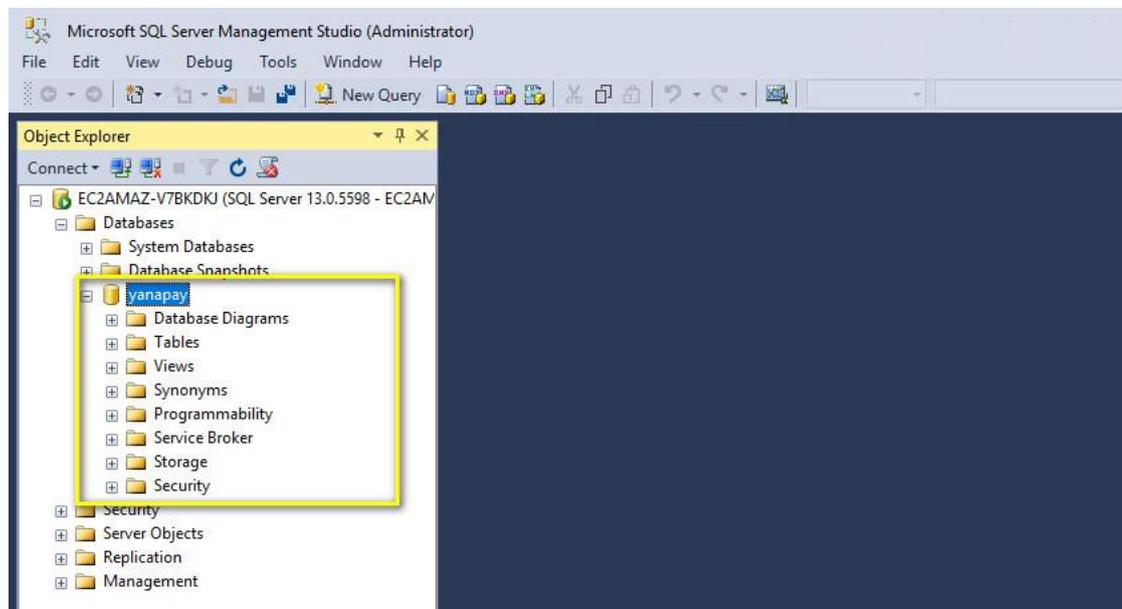


Figura 136: Base de datos Yanapay

1.2. Web API

Este servicio se encuentra desplegado en el IIS (Internet Information Services) del servidor virtual.

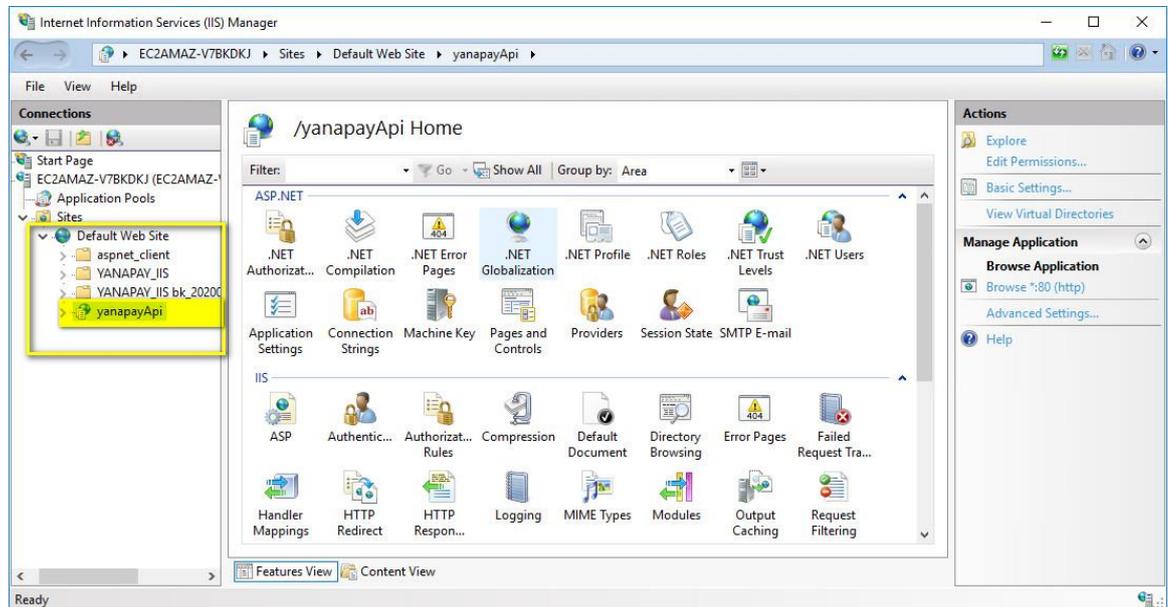


Figura 137: Servicio Web API desplegado

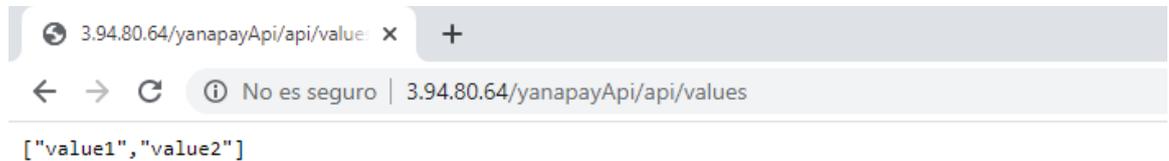


Figura 138: Web API publicado

Anexo N° 18. Resultados tiempo medio de respuesta

1. Login o inicio de sesión

Las muestras han sido tomadas cuando el usuario presiona en el botón Ingresar luego de digitar sus credenciales.

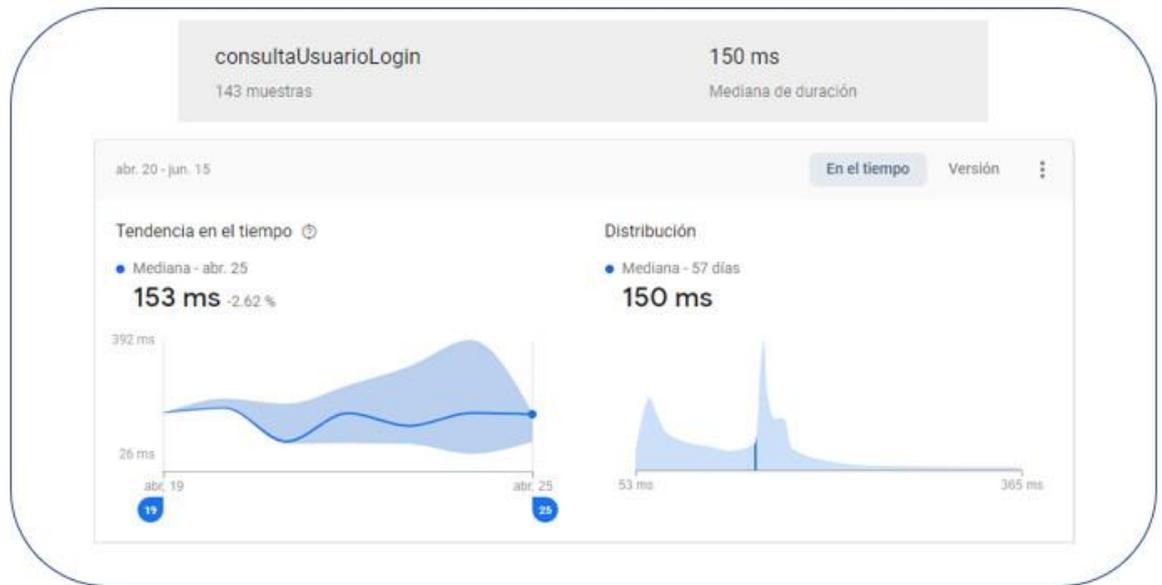


Figura 139: Tiempos de respuesta en iniciar sesión

2. Listar observaciones

Las muestras han sido tomadas cuando el usuario presiona la opción Observaciones ubicada en la barra de menú.



Figura 140: Tiempos de respuesta de listar observaciones

3. Listar categorías

Las muestras han sido tomadas cuando el usuario presiona en el botón de crear una nueva observación, esta vista adquiere una lista de Categorías con la finalidad que el usuario pueda escoger y seleccionar una de ellas.

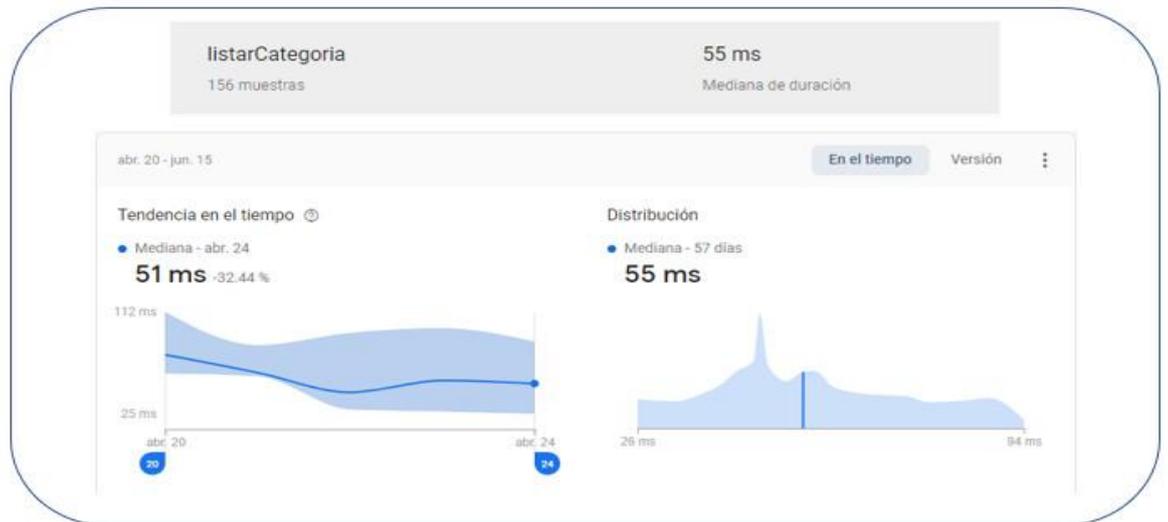


Figura 141: Tiempos de respuesta en listar categorías

4. Listar zonas

Las muestras han sido tomadas cuando el usuario presiona en el botón de crear una nueva observación, esta vista obtiene una lista de Zonas para que el usuario pueda escoger y seleccionar una de ellas.

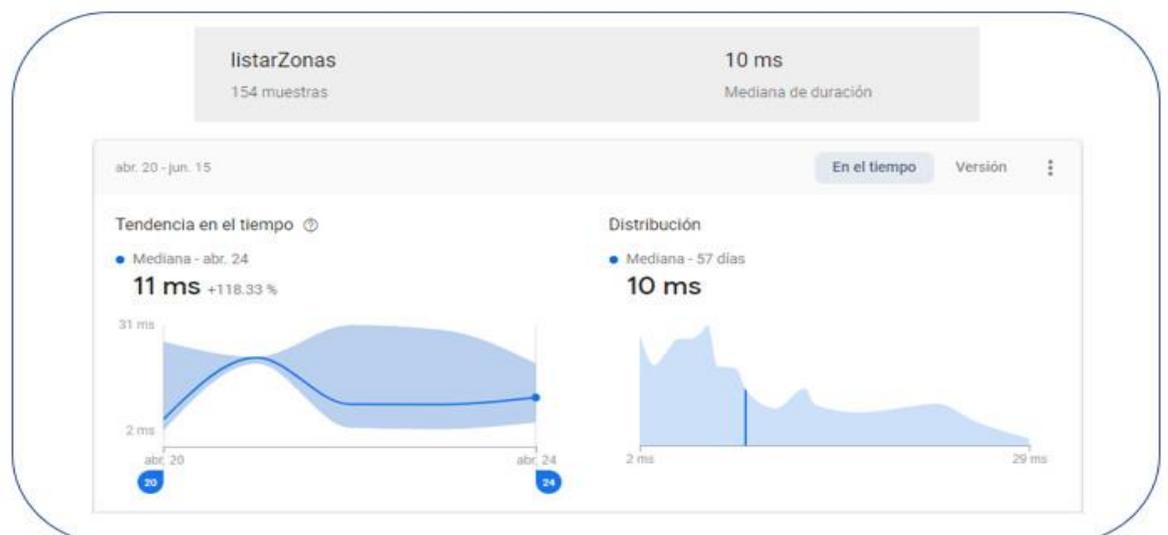


Figura 142: Tiempos de respuesta en listar zonas

5. Listar tipo de ocurrencias

Las muestras han sido tomadas cuando el usuario selecciona una Categoría, el aplicativo obtiene una lista de Ocurrencias para que el usuario pueda escoger y seleccionar una de ellas.

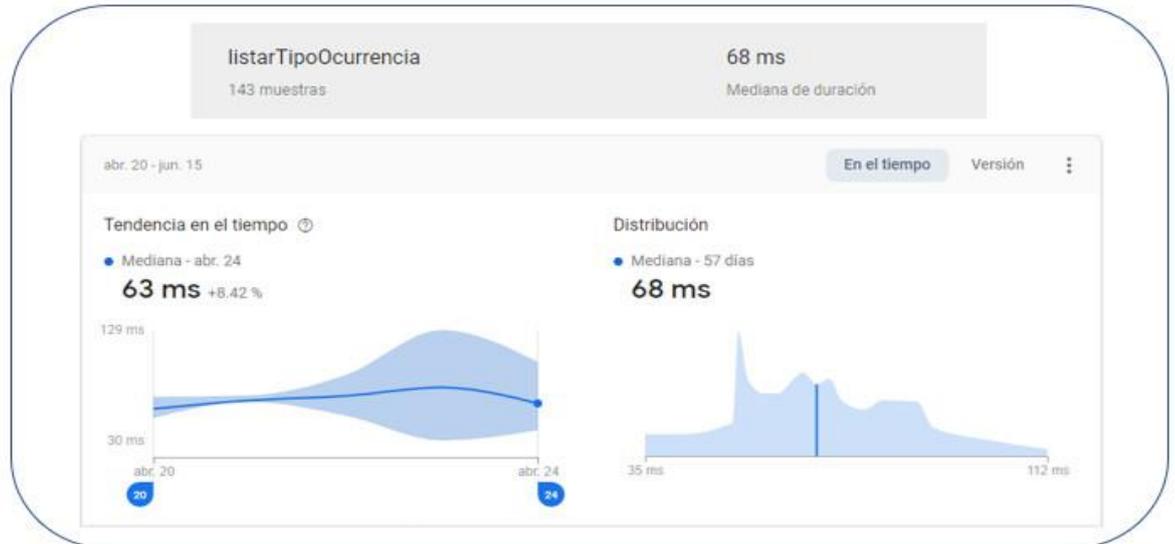


Figura 143: Tiempos de respuesta en listar tipos de ocurrencia

6. Listar nivel de riesgo

Las muestras han sido tomadas cuando el usuario presiona en el botón de crear una nueva observación, esta vista obtiene una lista de Nivel de Riesgo para que el usuario pueda escoger y seleccionar una de ellas.

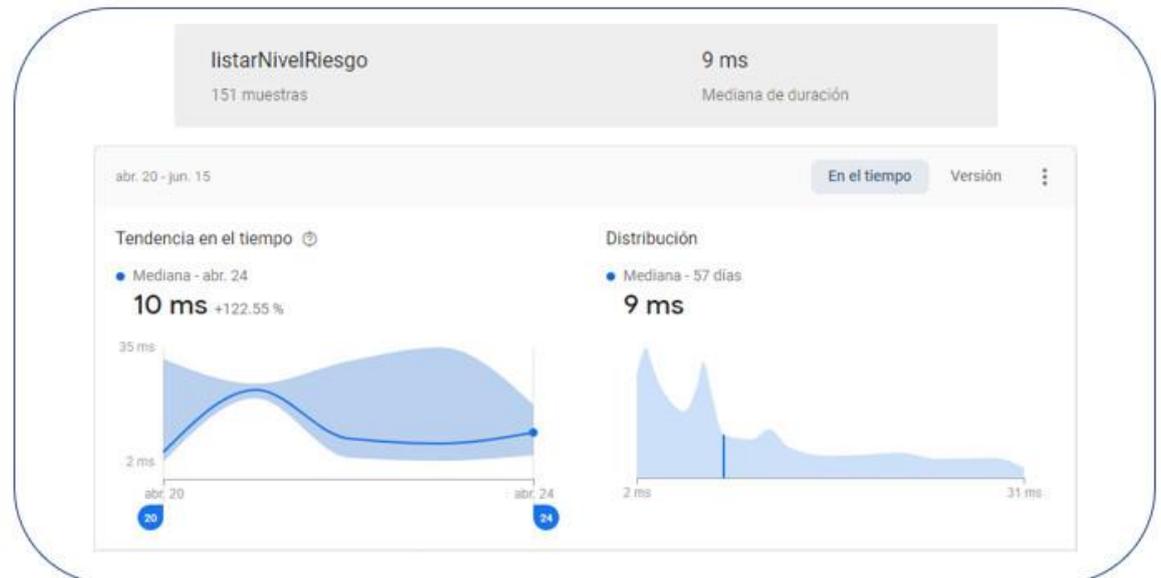


Figura 144: Tiempos de respuesta en listar niveles de riesgo

7. Listar personas

Las muestras han sido tomadas cuando el usuario presiona en el botón de crear una nueva observación, esta vista obtiene una lista de Personas para que el usuario pueda escoger y seleccionar una de ellas.

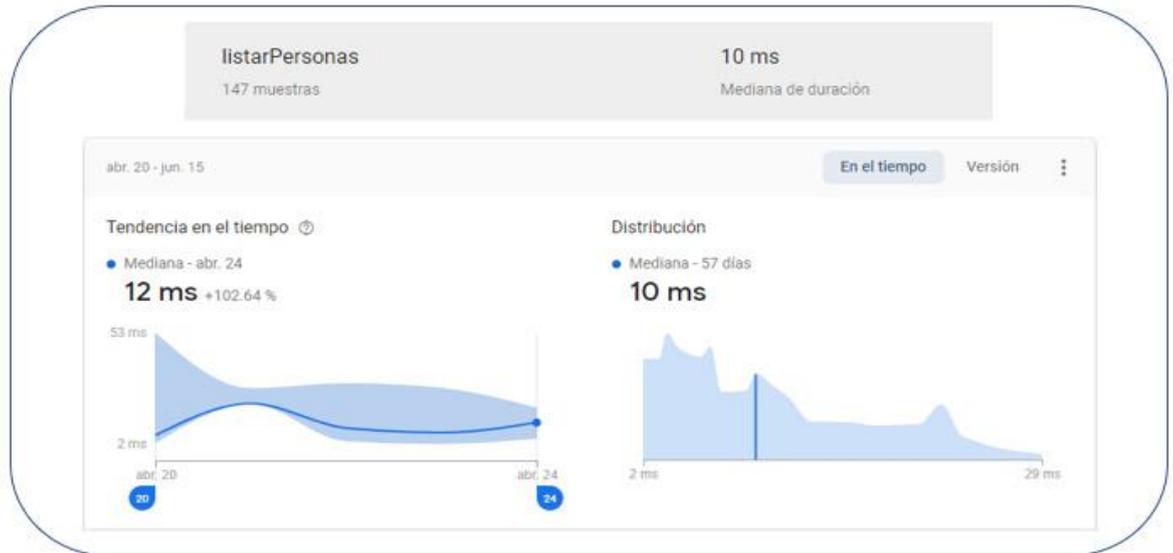


Figura 145: Tiempos de respuesta en listar personas

8. Guardar observación

Las muestras han sido tomadas cuando el usuario presiona en el botón de Guardar luego de haber completado los datos requeridos para crear una observación.

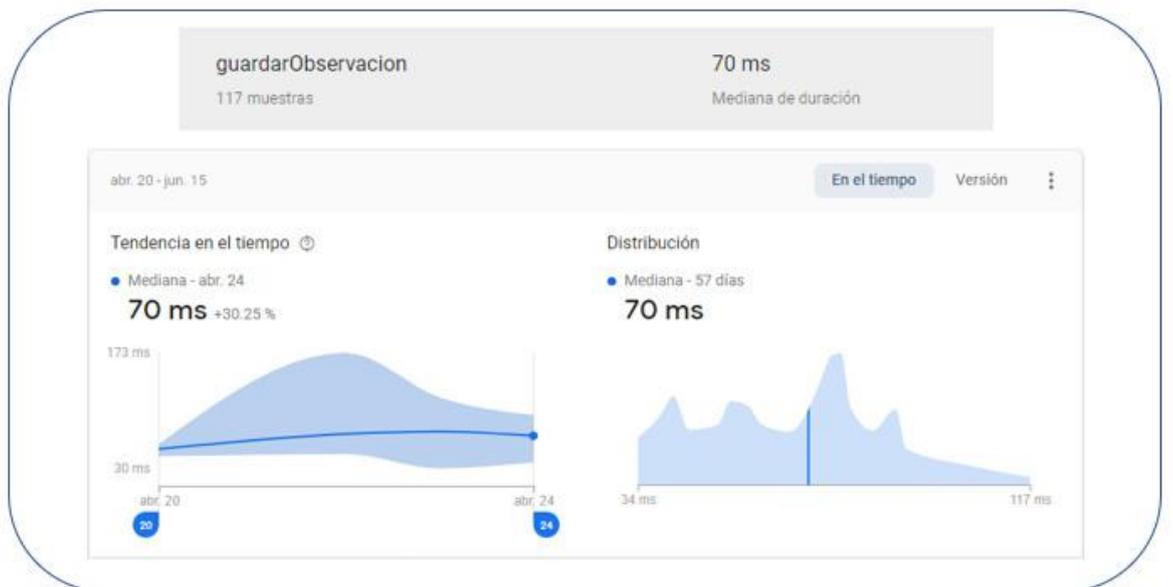


Figura 146: Tiempos de respuesta en guardar una observación

9. Guardar evidencia

Las muestras han sido tomadas cuando el usuario guarda una evidencia para las observaciones que tiene registradas.

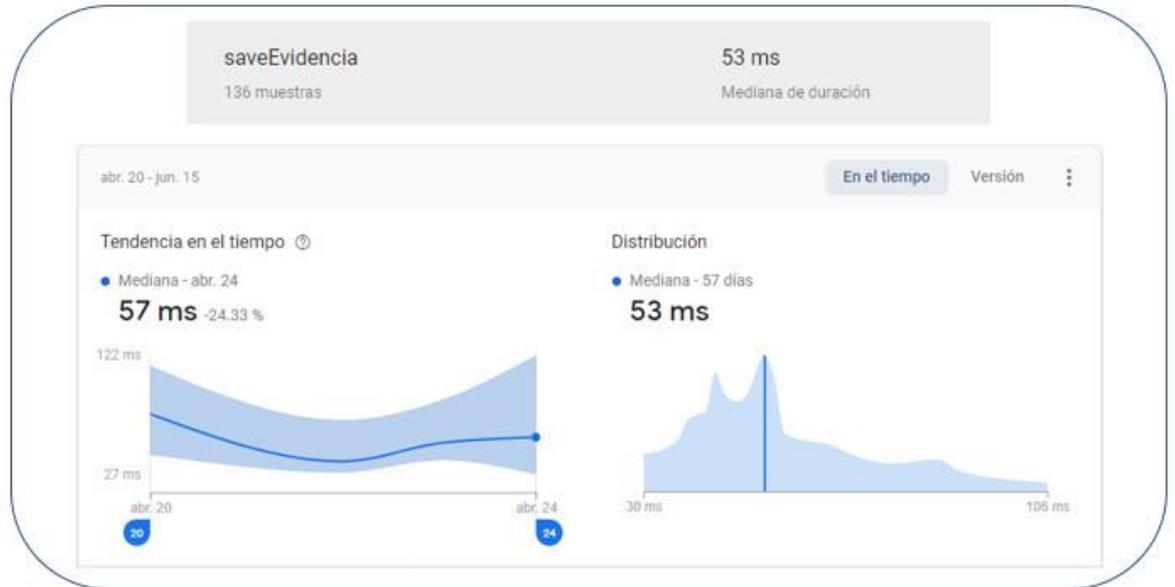


Figura 147: Tiempos de respuesta en guardar una evidencia

Anexo N° 19. Resultados rendimiento de la interfaz de usuario

1. Vista de presentación

Las muestras fueron tomadas cuando se inicia el aplicativo.



Figura 148: Rendimiento en vista presentación

2. Login o inicio de sesión

Las muestras fueron tomadas cuando el aplicativo solicita el ingreso de las credenciales del usuario.

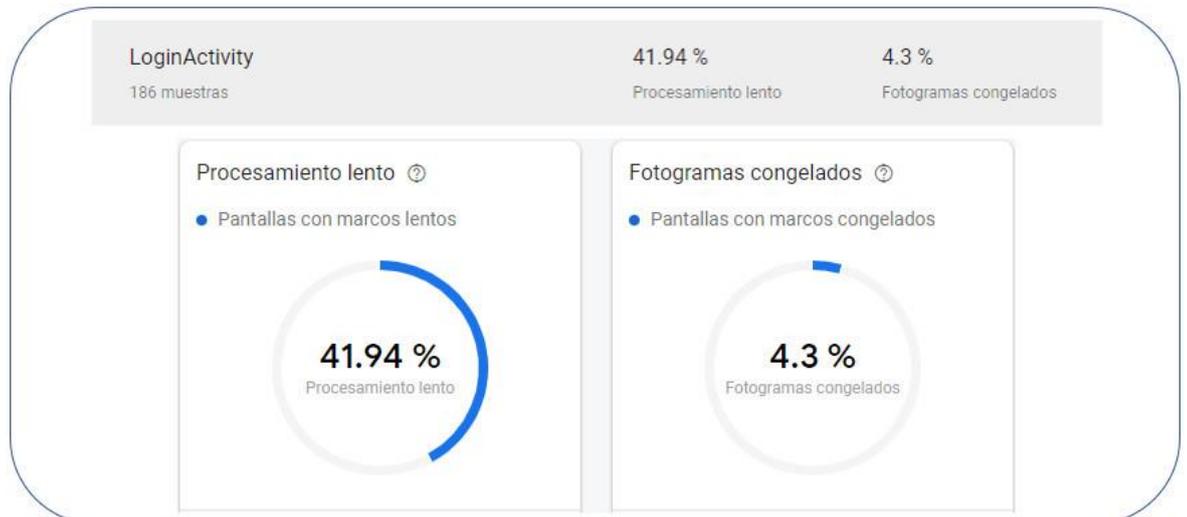


Figura 149: Rendimiento en vista login

3. Menú

Las muestras fueron tomadas cuando el usuario interactúa entre las vistas del menú.

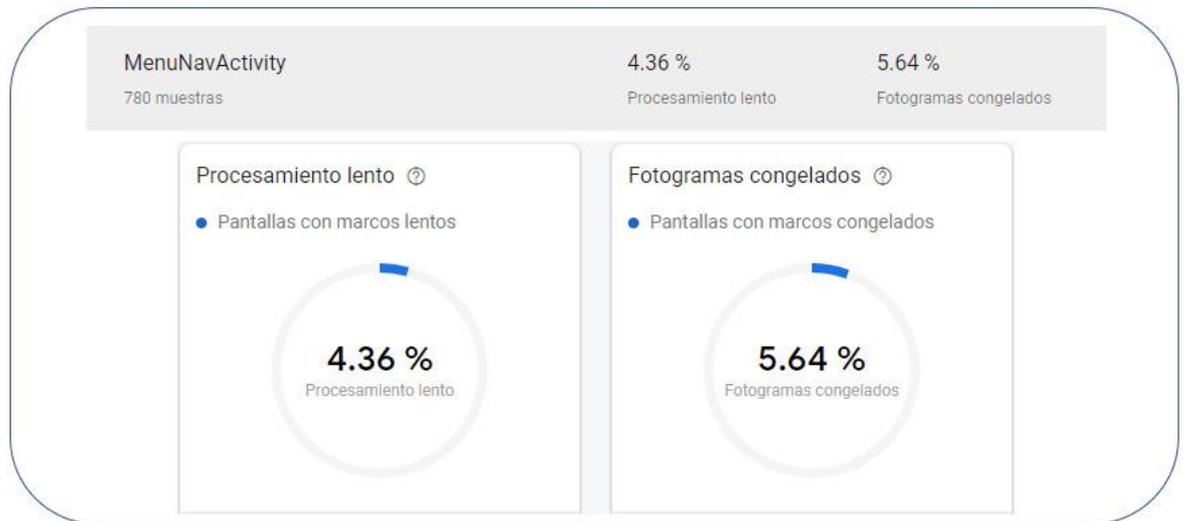


Figura 150: Rendimiento en vista menú

4. Registrar observación

Las muestras fueron tomadas cuando el usuario se encuentra en la vista para registrar una observación.



Figura 151: Rendimiento en vista registrar observación

5. Registrar evidencia

Las muestras fueron tomadas cuando el usuario adjunta las evidencias, en la vista del detalle de observación.

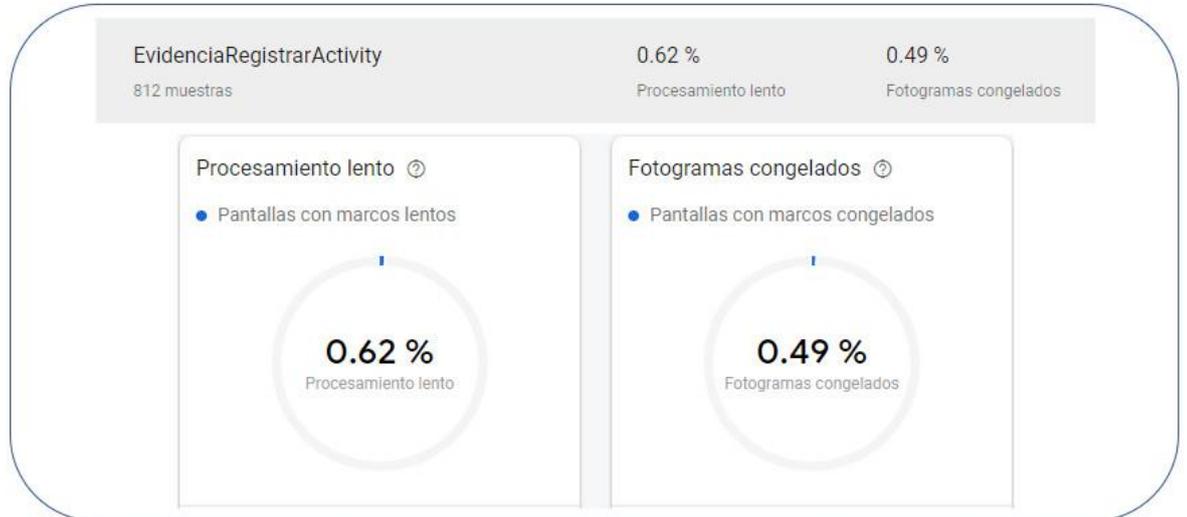


Figura 152: Rendimiento en vista registrar evidencia

Anexo N° 20. Resultados utilización máxima de memoria

1. Login o inicio de sesión

Las muestras fueron tomadas cuando los usuarios interactúan en la vista del inicio de sesión.

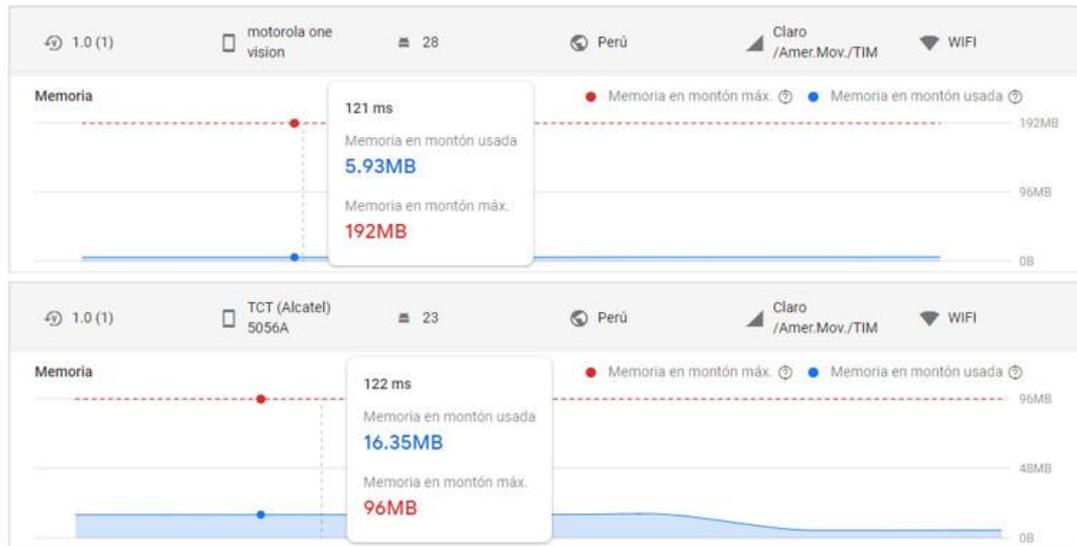


Figura 153: Memoria utilizada en iniciar sesión

2. Listar observaciones

Las muestras han sido tomadas cuando los usuarios interactúan en la vista del listado de observaciones.

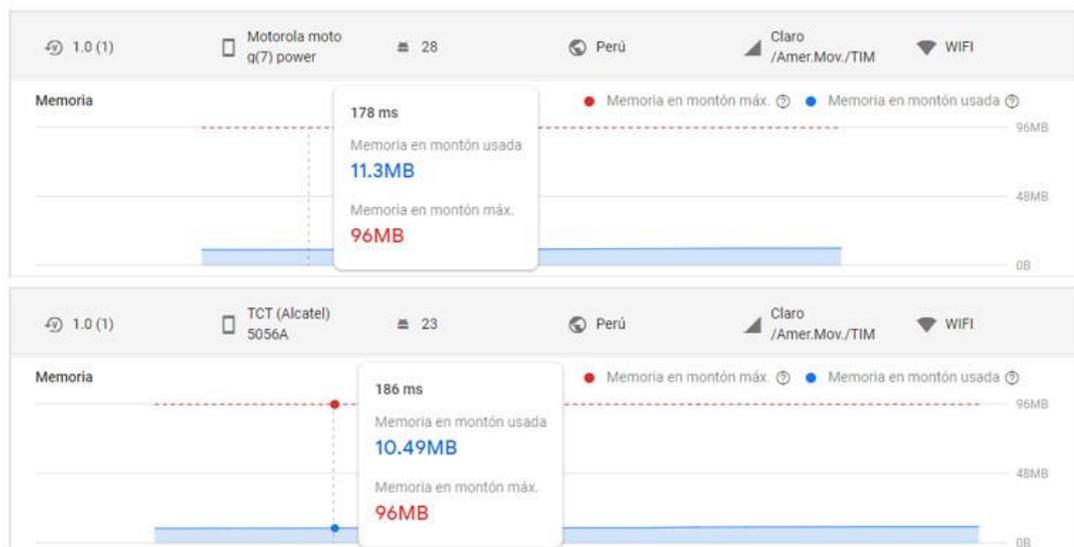


Figura 154: Memoria utilizada en listar observaciones

3. Listar categorías

Las muestras fueron tomadas cuando los usuarios interactuaban en la vista: crear una nueva observación, esta vista obtiene una lista de Categorías para que el usuario pueda escoger y seleccionar una de ellas.

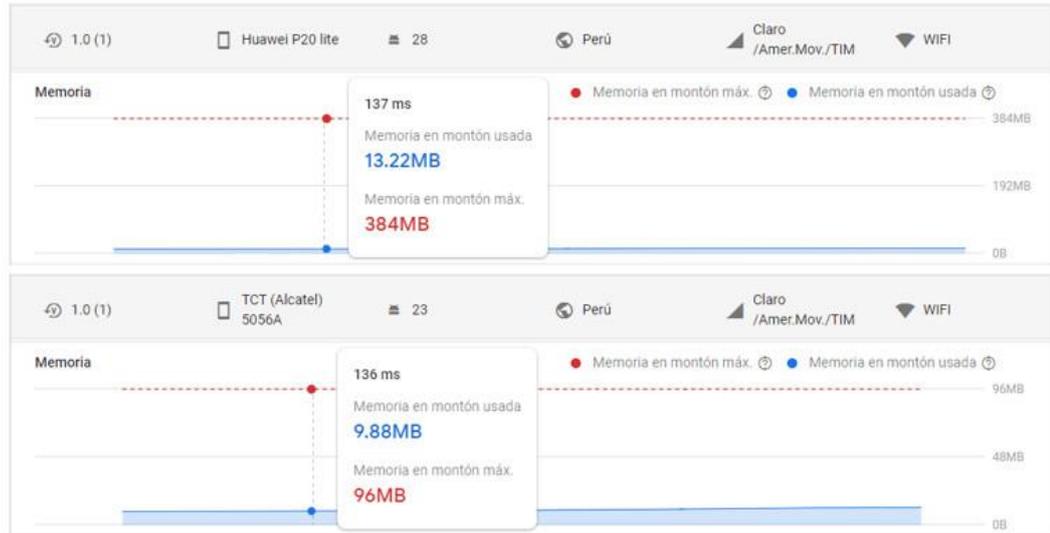


Figura 155: Memoria utilizada en listar categorías

4. Listar zonas

Las muestras han sido tomadas cuando los usuarios interactúan en la vista crear una nueva observación, esta vista obtiene una lista de Zonas para que el usuario pueda escoger y seleccionar una de ellas.

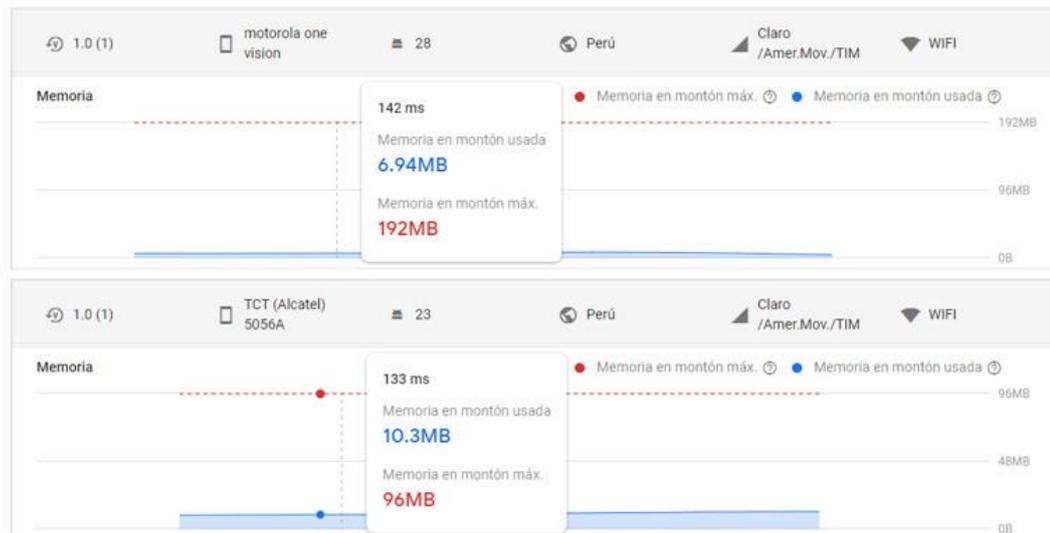


Figura 156: Memoria utilizada en listar zonas

5. Listar tipo de ocurrencias

Las muestras han sido tomadas cuando los usuarios interactúan en la vista crear una nueva observación y seleccionan una Categoría, el aplicativo obtiene una lista de Ocurrencias relacionadas a la categoría para que el usuario pueda escoger y seleccionar una de ellas.

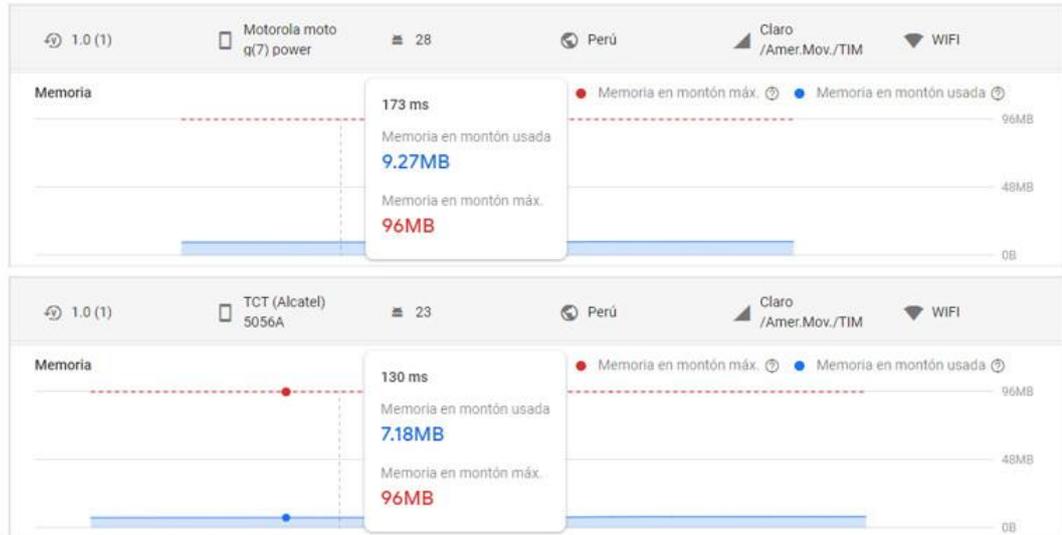


Figura 157: Memoria utilizada en listar tipos de ocurrencia

6. Listar nivel de riesgo

Las muestras han sido tomadas cuando los usuarios interactúan en la vista crear una nueva observación, esta vista obtiene una lista de Nivel de Riesgo para que el usuario pueda escoger y seleccionar una de ellas.

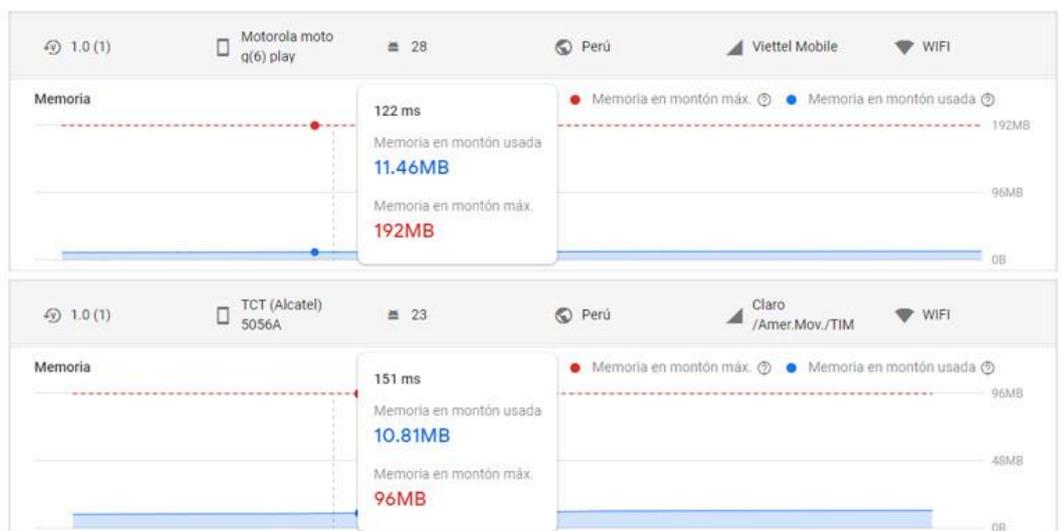


Figura 158: Memoria utilizada en listar niveles de riesgo

7. Listar personas

Las muestras han sido tomadas cuando los usuarios interactúan en la vista: crear una nueva observación, esta vista obtiene una lista de Personas para que el usuario pueda escoger y seleccionar una de ellas.

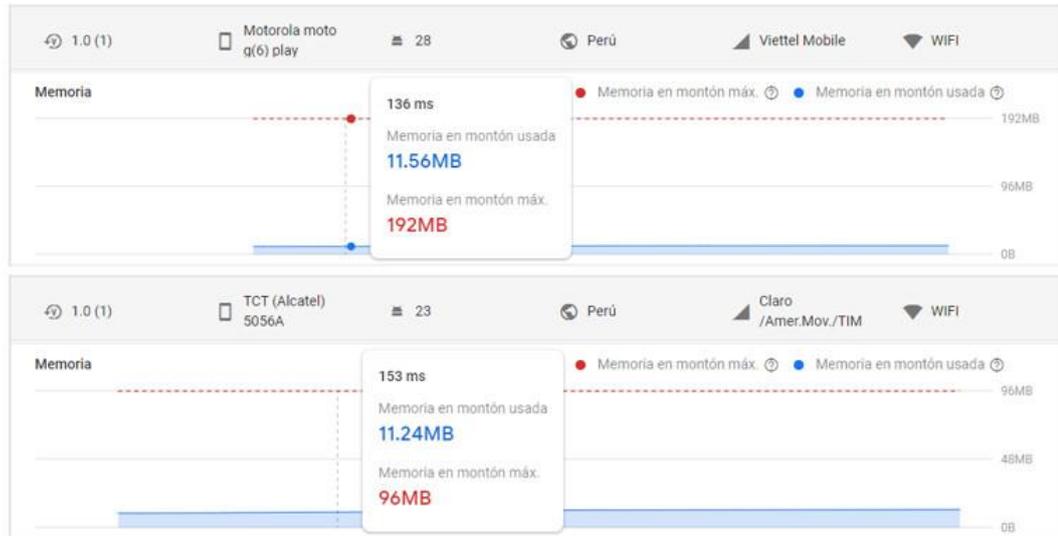


Figura 159: Memoria utilizada en listar personas

8. Guardar observación

Las muestras han sido tomadas cuando los usuarios interactuaban en la vista: crear una nueva observación y presionan en el botón de Guardar luego de haber completado los datos requeridos.

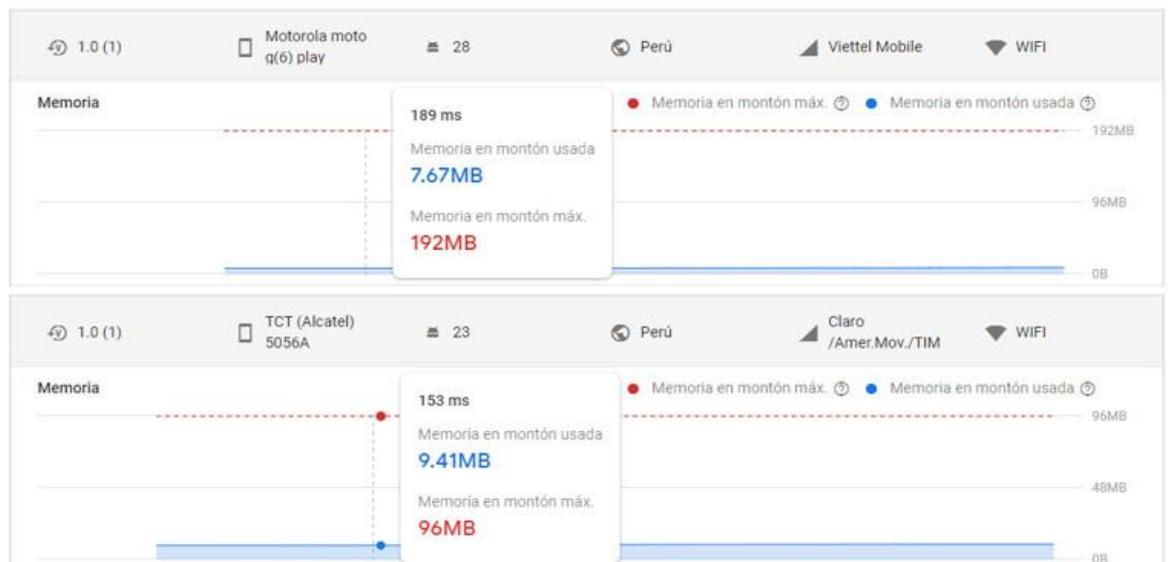


Figura 160: Memoria utilizada en guardar una observación

9. Guardar evidencia

Las muestras han sido tomadas cuando los usuarios interactúan en la vista: guardar una evidencia para las observaciones que tienen registradas.

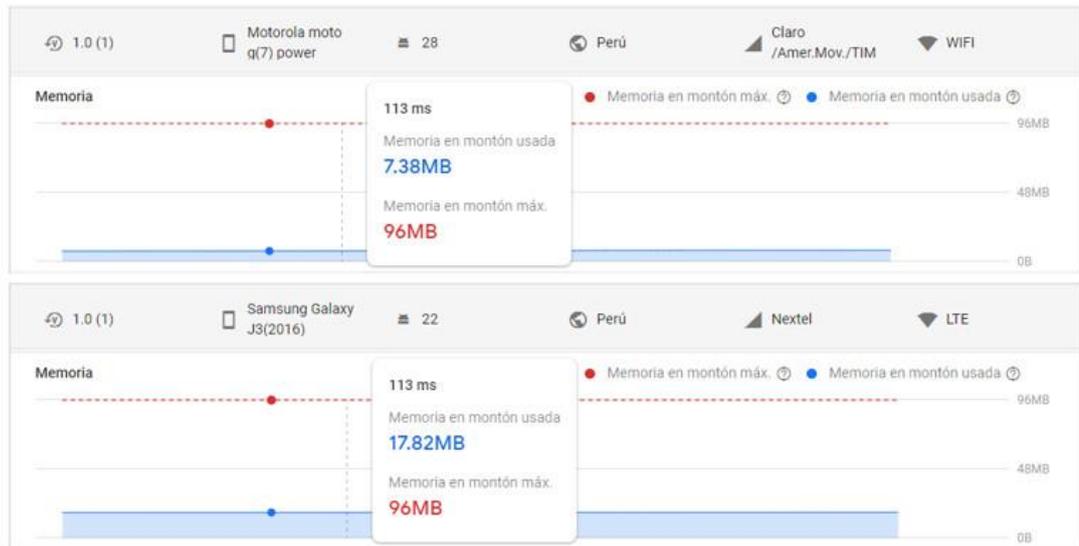


Figura 161: Memoria utilizada en guardar una evidencia

10. Cargar evidencia a Firebase Storage

Las muestras han sido tomadas en segundo plano antes que el aplicativo envíe a guardar una evidencia para las observaciones que tiene registradas.

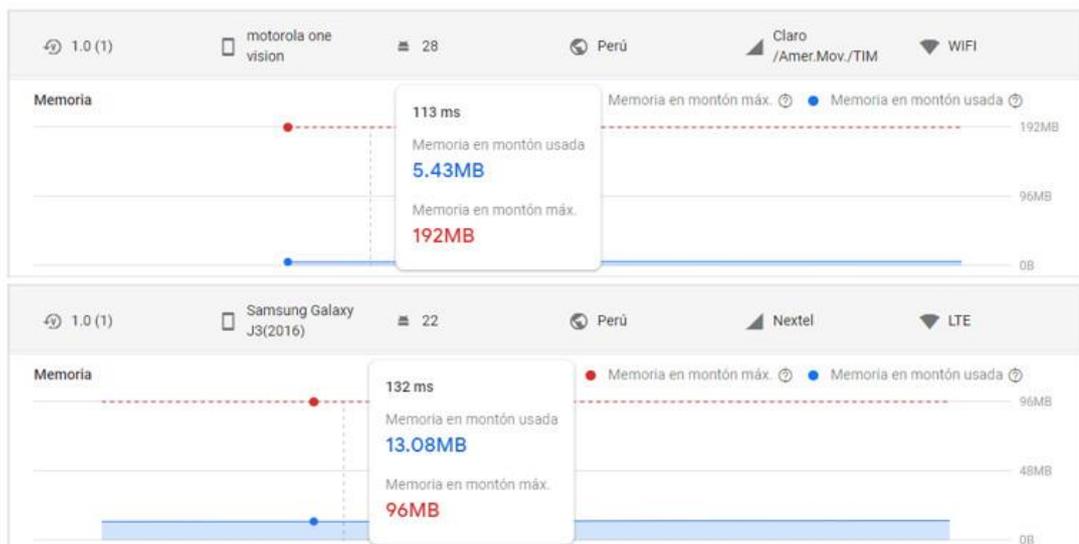


Figura 162: Memoria utilizada en cargar una evidencia a Firebase Storage

11. Descargar evidencia desde Firebase Storage

Las muestras han sido tomadas cuando el usuario descarga una de las evidencias que anteriormente han sido guardadas.

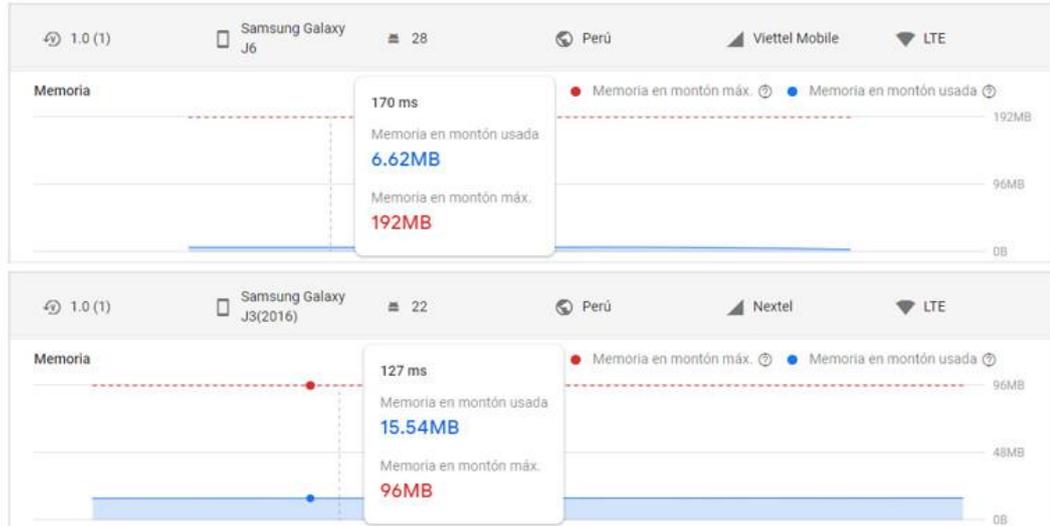


Figura 163: Memoria utilizada en descargar una evidencia desde Firebase Storage