

**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**



**Facultad de Ingeniería en Electricidad y Computación**

**“DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN PARA  
RECOLECCIÓN MÓVIL DE DATOS DE MEDICIONES DE CORRIENTES  
MARINAS Y CARACTERIZACIÓN COSTERA”**

**TRABAJO DE TITULACIÓN**

**PREVIO A LA OBTENCIÓN DEL TÍTULO DE  
MAGISTER EN SISTEMAS DE INFORMACIÓN GERENCIAL**

Autor:

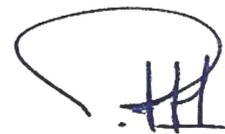
**MARCELO GIOVANNY CANDELL MONCADA**

Guayaquil – Ecuador

2020

## AGRADECIMIENTO

A la vida por darme tantas cosas buenas, a mis padres por su guía y esfuerzo, a mi familia y otras personas que, con paciencia, palabras de ánimo y consejos estuvieron para apoyarme cuando lo requerí.

A handwritten signature in blue ink, consisting of a large, rounded loop on the left and several vertical strokes on the right, resembling the letters 'H' or 'HH'.

## DEDICATORIA

A Papichu y Mamichi que me dieron la vida, los principios y valores que rigen mis actos, por sus consejos de siempre luchar y alcanzar mis objetivos.

A mi esposa Diana, por ser compañía en todos los momentos y más aún cuando sentía flaquear, a mi pequeño Matías por todos sus besos, abrazos y ocurrencias, a Diamar, que traerás más alegrías a nuestras vidas. A ustedes, por ser lo más importante en mi existencia, sin su presencia, apoyo y amor, esto no hubiera sido posible.

## TRIBUNAL DE SUSTENTACIÓN



**Msig. Lenin Freire Cobo**

**COORDINADOR MSIG**



**Msig. Omar Maldonado Dañín**

**DIRECTOR DEL TRABAJO DE TITULACIÓN**



**Msig. Ronny Santana Estrella**

**MIEMBRO DEL TRIBUNAL**

## RESUMEN

El presente trabajo de titulación fue desarrollado para un instituto de investigación científica, y se fundamentó en el diseño e implementación de una aplicación para recolección móvil de datos de mediciones de corrientes marinas y caracterización costera, que reemplace el uso de formularios de papel en el levantamiento de información en actividades de campo. La solución que se planteó para el diseño de la aplicación, fue mediante el uso de la plataforma Open Data Kit (ODK), la cual es una herramienta libre y de código abierto que ayuda a recopilar datos con dispositivos móviles, permitiendo diseñar formularios, recolectar datos y enviarlos a un servidor que se encarga de su gestión y almacenamiento, para su posterior interoperabilidad con otras aplicaciones y/o herramientas.

El trabajo concluyó con el diseño de formularios para la recolección móvil de datos que minimiza el error humano y la pérdida los mismos, la implementación de un visor geográfico que permite la presentación y búsqueda de información en una línea de tiempo de acuerdo a parámetros seleccionados y la generación de reportes de los datos recolectados y almacenados, tareas que antes se realizaban manualmente y demoraban en ser ejecutadas.

## ÍNDICE GENERAL

AGRADECIMIENTO .....	II
DEDICATORIA .....	III
TRIBUNAL DE SUSTENTACIÓN.....	IV
RESUMEN .....	V
ÍNDICE GENERAL .....	VI
ABREVIATURAS Y SIMBOLOGÍA.....	XI
ÍNDICE DE FIGURAS.....	XII
ÍNDICE DE TABLAS.....	XVII
INTRODUCCIÓN.....	XVIII
CAPÍTULO 1.....	1
1.1. Antecedentes.....	1
1.2. Descripción del problema.....	4

1.3. Solución propuesta.....	8
1.4. Objetivo general .....	11
1.5. Objetivos específicos .....	11
1.6. Metodología.....	12
CAPÍTULO 2.....	15
2.1. Modelo de desarrollo basado en componentes.....	15
2.1.1. Contexto.....	17
2.1.2. Definiciones básicas .....	19
2.1.3. Concepto de componente .....	20
2.2. Arquitectura de app móviles.....	24
2.2.1. Introducción .....	24
2.2.2. Información Geográfica Voluntaria y Sistemas de Información ..... Geográfica Participativa.....	27
2.2.3. Uso de dispositivos móviles para recolección de datos.....	28

2.2.4.	La Plataforma Open Data Kit (ODK) .....	35
2.2.5.	ODK Build y ODK XLSForm .....	41
2.2.6.	ODK Collect .....	43
2.2.7.	ODK Aggregate .....	45
2.3.	Arquitectura de aplicaciones web .....	48
2.3.1.	Infraestructura de Datos Espaciales y Geo Servicios Web.....	48
2.3.2.	GeoServer.....	52
2.3.3.	Geodatabase .....	53
2.3.4.	PostgreSQL .....	53
2.3.5.	PostGIS.....	54
2.4.	Metodología de desarrollo.....	56
CAPÍTULO 3	.....	58
3.1.	Definición de la situación actual.....	58

3.2. Levantamiento de información de los procesos actuales .....	62
3.2.1. Medición de corrientes marinas.....	62
3.2.2. Caracterización costera .....	67
3.3. Definición de actores .....	69
3.4. Definición de requerimientos.....	70
3.5. Alcance del proyecto .....	73
CAPÍTULO 4 .....	76
4.1. Análisis de requerimientos .....	76
4.2. Diseño de formularios .....	102
4.3. Modelado de base de datos .....	113
4.4. Desarrollo de formularios .....	116
4.5. Desarrollo de sitio web de visualización .....	117
4.6. Plan de pruebas e implementación de formularios.....	128

CAPÍTULO 5 .....	134
5.1. Evaluación de la solución implementada .....	134
5.2. Resultados de la solución implementada.....	139
CONCLUSIONES Y RECOMENDACIONES .....	144
BIBLIOGRAFÍA.....	146
ANEXO .....	151

## ABREVIATURAS Y SIMBOLOGÍA

<b>CSV:</b>	Comma Separated Values.
<b>ETL:</b>	Extract, Transformation and Load.
<b>GPS:</b>	Global Positioning System.
<b>IDE:</b>	Infraestructura de Datos Espaciales.
<b>INVEOCE:</b>	Instituto de Investigación Oceanográfica.
<b>GeoJSON:</b>	Geographical features JavaScript Object Notation.
<b>ODK:</b>	Open Data Kit.
<b>OGC:</b>	Open Geospatial Consortium.
<b>OWS:</b>	OGC Web Services.
<b>WCS:</b>	Web Coverage Service.
<b>WFS:</b>	Web Feature Service.
<b>WMS:</b>	Web Map Service.
<b>XML:</b>	eXtensible Markup Language.

## ÍNDICE DE FIGURAS

Figura 2.1: Smartphones vendidos a nivel mundial 2011 - 2018.....	30
Figura 2.2: Formulario tradicional en papel (Características litorales).....	33
Figura 2.3: Formulario tradicional en papel (Planillas para veleteo).....	33
Figura 2.4: Flujo de trabajo en recolección de datos con ODK.....	40
Figura 2.5: Ejemplo de pantalla de ODK Build.....	42
Figura 2.6: Ejemplo de pantalla de ODK Collect.....	45
Figura 2.7: Ejemplo de pantalla de ODK Aggregate .....	48
Figura 2.8: Arquitectura tecnológica propuesta.....	55
Figura 3.1: Sistema de información del instituto .....	60
Figura 3.2: Carga de archivos de medición de corrientes marinas.....	61
Figura 3.3: Fases del procedimiento de medición de corrientes (veleteo) ....	65
Figura 4.1: Inicio de instalación de ODK Aggregate .....	78

Figura 4.2: ODK Aggregate, selección de plataforma.....	79
Figura 4.3: ODK Aggregate, configuración Apache Tomcat.....	80
Figura 4.4: ODK Aggregate, configuración PostgreSQL.....	81
Figura 4.5: Interfaz web de ODK Aggregate .....	82
Figura 4.6: ODK Aggregate, configuración de acceso al sitio.....	83
Figura 4.7: Descarga de ODK Collect desde Google Play Store.....	84
Figura 4.8: Configuración de URL del servidor en ODK Collect.....	85
Figura 4.9: URL del servidor ODK Aggregate configurada .....	85
Figura 4.10: Ingreso de usuario en ODK Collect.....	86
Figura 4.11: Ingreso de contraseña en ODK Collect .....	86
Figura 4.12: Cambio de clave de usuario en ODK Aggregate .....	87
Figura 4.13: Añadir nuevo usuario en ODK Aggregate.....	90
Figura 4.14: Añadir datos de nuevo usuario en ODK Aggregate.....	90

Figura 4.15: Actualizar datos de usuario en ODK Aggregate .....	91
Figura 4.16: Eliminar usuario en ODK Aggregate .....	92
Figura 4.17: Carga de formulario en ODK Aggregate .....	93
Figura 4.18: Eliminar formulario en ODK Aggregate.....	94
Figura 4.19: Confirmación de eliminación de formulario.....	94
Figura 4.20: Configurar envío de formulario empleando WiFi (1).....	95
Figura 4.21: Configurar envío de formulario empleando WiFi (2).....	96
Figura 4.22: Configurar envío de formulario empleando WiFi (3).....	96
Figura 4.23: Configurar envío de formulario empleando WiFi (4).....	97
Figura 4.24: Presentación de datos por ubicación y tiempo .....	98
Figura 4.25: Búsqueda de datos en visor geográfico.....	101
Figura 4.26: Exportación de datos a archivo PDF.....	102
Figura 4.27: Documento Estándar XLSForm en Excel .....	107

Figura 4.28: Convertidor XLSForm online .....	112
Figura 4.29: Descarga del formulario en formato XML .....	112
Figura 4.30: Previsualización del formulario en Enketo .....	113
Figura 4.31: Estructura de la tabla de formulario .....	115
Figura 4.32: Estructura de la tabla de archivos.....	116
Figura 4.33: Pantalla principal del GeoServer.....	119
Figura 4.34: Origen de datos en almacén de GeoServer .....	120
Figura 4.35: Origen de datos vectoriales PostGIS .....	120
Figura 4.36: Creación de vista en la base de datos .....	121
Figura 4.37: Creación de capa en GeoServer.....	122
Figura 4.38: Previsualización de capas en GeoServer .....	123
Figura 4.39: Respuesta de servicio en formato GeoJSON .....	123
Figura 4.40: Vista general de plantilla web .....	125

Figura 4.41: Control de línea de tiempo en visualizador geográfico .....	127
Figura 4.42: Tabla de datos y botones de exportación .....	128
Figura 4.43: ODK Collect menú principal .....	130
Figura 4.44: Listado de formularios disponibles.....	131
Figura 4.45: Formulario para datos de veleteo .....	132
Figura 4.46: Formulario para datos de caracterización costera.....	133
Figura 5.1: Gráfico pregunta 1 .....	141
Figura 5.2: Gráfico pregunta 2.....	141
Figura 5.3: Gráfico pregunta 3.....	142
Figura 5.4: Gráfico pregunta 4.....	142
Figura 5.5: Gráfico pregunta 5.....	143
Figura 5.6: Gráfico pregunta 6.....	143

## ÍNDICE DE TABLAS

Tabla 1: Hoja de trabajo de encuesta, formulario veleteo.....	108
Tabla 2: Hoja de trabajo de encuesta, formulario caracterización.....	109
Tabla 3: Resultados tabulados de cuestionario a técnicos.....	140

## INTRODUCCIÓN

La recolección de datos en actividades de campo, resulta ser fundamental en todo proceso de investigación, y consiste en recopilar información dentro de ciertas circunstancias y entorno, siendo la condición previa para obtener un conocimiento científico. Dada la importancia y criticidad de esta tarea, en la actualidad los técnicos e investigadores que la ejecutan, requieren cada vez contar con mejores herramientas e implementos que permitan la eficiente consecución de sus objetivos, reduciendo la probabilidad de errores, pérdida de datos, así como minimizar el tiempo total empleado.

Las aplicaciones móviles resultan ser una solución orientada a solventar los inconvenientes encontrados por los técnicos del Instituto de Investigación Oceanográfica, INVEOCE, los cuales necesitan recolectar, preservar, almacenar y analizar los datos provenientes de la recolección en campo, evidenciándose su malestar con dos procesos en particular: mediciones de corrientes marinas y caracterización costera. Con esto se entiende que el empleo de esta tecnología informática, facilitará y simplificará en ciertos casos el trabajo de los usuarios, eliminando el levantamiento de información

mediante el uso de formularios físicos, los cuales tienden a perderse y deteriorarse, y qué decir del tiempo empleado para su digitalización.

La plataforma Open Data Kit (ODK) resulta ser una alternativa idónea para resolver la situación actual, siendo una herramienta gratuita, de código abierto, con una curva de aprendizaje mínima, que no requiere software adicional para su funcionamiento, ni un entorno de desarrollo complejo, ni hardware e alta prestación que soporte su implementación.

Capítulo 1, describe brevemente la situación actual del instituto, el problema existente en la recolección de datos en campo, objetivo general y objetivos específicos a lograr con el diseño e implementación de la aplicación.

Capítulo 2, en donde se detalla el marco teórico que fundamenta el desarrollo del presente trabajo, presentando definiciones sobre componentes, sistemas, la plataforma ODK, aplicaciones móviles y aplicaciones web.

Capítulo 3, definición de la situación actual y requerimientos, levantamiento de los procesos actuales, definición de requerimientos funcionales y requerimientos no funcionales, así como el alcance del proyecto.

Capítulo 4, detalla el análisis, diseño e implementación de la aplicación para recolección móvil de datos de mediciones de corrientes marinas y caracterización costera.

Capítulo 5, análisis de resultados, evaluación y resultados de la aplicación.

## **CAPÍTULO 1**

### **GENERALIDADES**

#### **1.1. Antecedentes**

Ecuador se encuentra a orillas del océano Pacífico, el más grande de la Tierra, que ocupa la tercera parte de su superficie; nuestro país tiene un territorio marítimo 5.3 veces más extenso que el territorio continental, con una superficie de 1'092.140 km<sup>2</sup>, siendo esta zona altamente potencial para generar mayores posibilidades de investigación, ciencia, tecnología y desarrollo en el mar.

En los inicios de la vida republicana en nuestro país existían cartas náuticas incompletas, las mismas que provocaban varamientos y otros accidentes marítimos, sin que exista mayor ayuda y referencias para la

navegación segura en esteros, ríos navegables y canales de acceso a puertos marítimos. Con respecto a la señalización náutica, a mediados del siglo XIX se contaba con pocos faros y fanales, siendo brindado al Estado este servicio de carácter público desde sus inicios por la Armada.

En la década de 1930 se crea el Servicio Hidrográfico, con funciones que se circundaban a la realización de levantamientos hidrográficos, los mismos que eran empleados en la elaboración de las primeras cartas náuticas de la costa del país.

En el año de 1972, el Servicio Hidrográfico es elevado a categoría de Instituto de Investigación Oceanográfica, encargándose de proporcionar seguridad en la navegación, realizar investigación oceanográfica y compilar la cartografía náutica del Ecuador, además de la representación del Estado ante organismos internacionales. Con esta nueva categoría alcanzada, se deriva la necesidad de establecer coordinación entre programas nacionales y de cooperación que se desarrollan a nivel internacional en el campo de la investigación oceanográfica, además de ser un organismo oficial permanente con autoridad en materia de desarrollo de las ciencias del mar.

El Instituto de Investigación Oceanográfica INVEOCE tiene sede en la ciudad de Guayaquil y jurisdicción nacional, con personería jurídica, patrimonio y fondos propios, siendo parte de su misión institucional entre otras, el monitoreo de condiciones oceanográficas, para lo cual obtiene, procesa, analiza, difunde y transfiere geoinformación marítima, contribuyendo así al conocimiento integral del mar, desarrollo marítimo, defensa y conservación del territorio marítimo nacional. Entre sus funciones generales se resaltan la realización, dirección, coordinación y control de las actividades de monitoreo y exploración oceanográfica, geofísica y de las ciencias del medio ambiente marítimo.

De acuerdo con su distributivo de personal, a septiembre de 2019 cuenta con 197 empleados, y en su orgánico se detalla un director, un subdirector general, un director administrativo financiero, y directores en las direcciones técnicas de: Hidrografía y Cartografía, Navegación, Oceanografía Naval, Plataforma Continental y Fondo Oceánico, Geoinformación Marítima, y de Fortalecimiento de Capacidades y Difusión de Información.

Dentro del contexto de la organización, se detallan como clientes de INVEOCE a la comunidad marítima, Gobiernos Autónomos Descentralizados, empresas navieras y pesqueras, Asociación de

Prácticos de Guayaquil, Autoridad Portuaria de Guayaquil y Armada del Ecuador. Estos clientes tienen perspectivas acerca de la institución, las cuales, de acuerdo al enfoque estratégico de la misma, están consideradas en el catálogo de objetivos mediante la implementación de temas estratégicos como: desarrollar estudios hidro-oceanográficos e implementar sistemas de hidro-oceanografía y de alertas de amenazas de origen oceánico, entre otros.

Entre los procesos agregadores de valor de INVEOCE, se encuentra el macroproceso de Gestión de Oceanografía, con procesos de Gestión de Observación del océano y Gestión de Oceanografía Operacional; y es justamente en este último proceso en donde se realizan trabajos de campo basados en observaciones y mediciones, en las cuales se ha detectado oportunidad de mejora para dejar de emplear formularios en papel para el registro de datos.

## **1.2. Descripción del problema**

Poco antes de terminar la primera década de este siglo, se contaba con poca información en cuanto a las características oceanográficas en el Ecuador, y la que existía no se encontraba en forma integrada, ordenada y actualizada, razón por la que se propuso ejecutar un monitoreo multidisciplinario que abarque la mayor parte de las componentes

ambientales que permitan caracterizar la zona costera con un enfoque amplio y de una manera integrada.

En los procesos dinámicos de la zona costera intervienen tanto factores naturales como artificiales, los cuales producen cambios en la morfología de las playas provocando procesos de erosión o sedimentación. Los parámetros principales establecidos para la determinación de los procesos costeros son corrientes, altura y dirección de olas, magnitud y fuerza del viento, tipo de sedimento, tipo de morfología costera, entre otros; además se propuso integrar en nuestras costas una red nacional de monitoreo oceanográfico, de tal forma que se pueda caracterizar nuestras costas desde el punto de vista físico, químico, biológico, geológico y meteorológico, a fin de contribuir con investigación oceanográfica, y que los organismos competentes puedan formular y promulgar respuestas y planes de manejo en los diferentes campos de acción.

En el año 2008 se presenta el proyecto macro “Caracterización oceanográfica de la costa ecuatoriana”, el mismo que tiene como objetivo contribuir con el conocimiento de los procesos oceanográficos de tal forma que permitan el manejo integrado de los recursos costeros; este proyecto se presenta para ser ejecutado con continuidad en el tiempo,

sin que sea necesario volver a presentar documentación al momento de ejecutar actividades dentro del mismo.

En el marco de este proyecto, actualmente la División de Oceanografía Física del instituto, tiene problemas en dos procesos que se realizan con relativa regularidad: veleteo y caracterización costera.

En la medición de corrientes en el Golfo de Guayaquil se aplica el método lagrangiano, también denominado veleteo, el cual se basa en la observación del movimiento de una parcela de agua, para lo cual, a bordo de un bote, se siembra, observa y recoge las veletas o derivadores cada cierto tiempo; la posición de un punto específico en un determinado tiempo se la obtiene con la lectura del GPS a bordo de la embarcación, para lo cual el personal designado debe acercarse al derivador, anotar su posición y hora.

La caracterización en campo por estación, es un monitoreo en el que se emplea otro método de obtención de datos, ya que se realiza en un solo lugar a diferencia del veleteo. Se trata de observaciones que se desarrollan cada hora en un mismo lugar con diferentes parámetros y características, teniendo que realizar actividades como: medición de la altura, periodo y dirección de las olas; medición del ancho de la playa en

pleamar, media y bajamar y medición de la corriente litoral. Tanto en el proceso de veleteo como de caracterización, se presentan los siguientes problemas:

- En cada monitoreo se debe realizar el registro manual de los parámetros obtenidos en las mediciones en formularios impresos en papel, complementado con el uso de dispositivos GPS para obtener la posición de la observación, cámara para el registro fotográfico de la actividad, tableros y protectores para documentos, portafolio y bolígrafo, lo cual supone que en muchas ocasiones el técnico no tenga mayor control de los implementos que lleva consigo, y genera problemas en la calidad y consistencia de los datos recolectados.
- Considerable inversión de tiempo, mayor disposición de recursos humanos capacitados que supervisen las actividades en campo, con la finalidad de disminuir la posibilidad de error en el proceso de recolección de datos.
- Demora en el registro de los datos recolectados, lo cual conlleva que se incremente el tiempo en que se concluye la actividad en cada punto de monitoreo.
- Error en la transcripción manual y posterior digitalización en hojas de cálculo de los datos recabados en los formularios.
- Desfase en el tiempo desde la realización del monitoreo hasta que se realiza la digitalización, procesamiento y análisis de los datos.

- El trabajo con los formularios genera inconvenientes como el registro incorrecto e incompleto de datos.
- Deterioro del material de registro en formato papel, por la manipulación en el campo, lo que provoca dificultad al momento de interpretar los datos.
- Pérdidas de los formularios y tableros de trabajo.

Es relevante para la División optimizar las actividades que se realizan en estos procesos, y mejorar así la forma de ejecutar los mismos, lo cuales son repetitivos dada su naturaleza. La no utilización de los datos recolectados y procesados, más un almacenamiento inadecuado en hojas de cálculo, conlleva un desperdicio de recursos de la organización, así como riesgo de pérdida de datos, lo cual puede afectar procesos de monitoreo y la ejecución de planes de acción a futuro.

### **1.3. Solución propuesta**

La solución que propone el presente trabajo de titulación, se refiere al diseño e implementación de una aplicación móvil basada en código abierto que permita a los usuarios la automatización de procesos que se realizan de manera manual, durante y después de la recolección de los datos, por lo que se plantea el desarrollo de formularios empleando la

plataforma ODK, y la implementación de un sitio web interno para la visualización, consultas históricas y generación de reportes de los datos recolectados y almacenados.

La plataforma ODK (Open Data Kit), es una herramienta de código abierto y gratuita que ofrece soluciones para recolección y manejo de datos móviles a las organizaciones, mediante el diseño e implementación de formularios totalmente personalizables, desarrollados con software libre y que funcionen en dispositivos móviles con sistema operativo Android.

La División de Oceanografía Física requiere contar con una herramienta versátil para las actividades de monitoreo en campo, que permita la recolección e ingreso de datos de manera más rápida, con formularios para ingreso de datos, módulo de seguridad, gestión de usuarios, así como la implementación de un sitio web interno que permita la visualización, consulta y exportación de datos, presentación georreferenciada de puntos de monitoreo y la generación de reportes.

Para el desarrollo de la aplicación móvil se emplearán los componentes de la plataforma ODK: ODK Builder para el diseño y creación de los formularios empleando la aplicación web propia de la plataforma; ODK

Aggregate que es un componente en el servidor, que se empleará para la administración y gestión de usuarios, almacenamiento y gestión de datos recolectados; y ODK Collect, para el ingreso y manipulación de datos, para luego ser enviados al servidor, teniéndose que los datos soportados van desde texto, imágenes, audio, videos y localización.

Para la visualización de datos, presentación de puntos de monitoreo y generación de reportes, en el backend se empleará Tomcat como servidor web y Geoserver para la publicación de datos espaciales en la web, el sistema de gestión y almacenamiento en base de datos se realizará con PostgreSQL en conjunto con su extensión espacial PostGIS para la administración y soporte de datos espaciales; y desde el lado del cliente o frontend, se implementará un sitio web interno basado en HTML5 que utilizará como componentes Open Layers, Leaflet, jQuery, Datatables (plugin de jQuery) y Bootstrap.

La implementación de la aplicación móvil con la plataforma ODK, presentará los siguientes beneficios para el instituto:

- Optimización del tiempo y recursos empleados en la recolección de datos.
- Evitará la pérdida y deterioro de los documentos físicos (formularios) empleados en cada monitoreo.

- Los formularios digitales son diseñados de manera dinámica y permiten cambios de manera rápida en caso de requerirse.
- Contar con una herramienta que permita la validación de campos obligatorios y tipos de datos requeridos en el monitoreo.
- Disminución del error humano en los formularios, ya que la sistematización se realiza de manera automática.
- Centralización de los datos en un sistema, para posterior visualización, consulta y análisis, sea de tipo estadístico o espacial.

#### **1.4. Objetivo general**

Implementar una aplicación móvil que permita en campo, la recolección y almacenamiento de datos de mediciones de corrientes marinas y caracterización costera, permitiendo reducir los problemas de pérdida de datos, automatización en el envío de información y la generación de reportes en menor tiempo.

#### **1.5. Objetivos específicos**

Los objetivos específicos definidos para el proyecto son los siguientes:

1. Realizar el levantamiento de información de las actividades de monitoreo.

2. Analizar los requerimientos planteados por los técnicos e investigadores.
3. Diseñar formularios para recolección de datos empleando una herramienta gratuita de código abierto.
4. Desarrollar un sitio web para visualización de datos y generación de reportes, que brinde soporte para estudios posteriores.
5. Evaluar los resultados obtenidos posterior a la implementación de la aplicación.

#### **1.6. Metodología**

Para la planificación y desarrollo de aplicaciones en general, existen algunas metodologías o marcos de trabajo que pueden ser aplicadas, todo depende del alcance y tiempo de ejecución que tiene un proyecto. Entonces, la metodología será el marco de trabajo para la estructuración, planificación y control del proceso de desarrollo de sistemas de información. En un proyecto de desarrollo de software, la metodología ayuda a definir: quién debe hacer, qué, cómo y cuándo debe hacerlo.

El presente trabajo de titulación será realizado empleando la Metodología de Desarrollo en Cascada, la cual ha sido elegida en función del proyecto y sus necesidades, considerando que este modelo se ajusta bien a pequeños proyectos donde los requisitos se entienden bien, que

comprenden actividades que fluyen de manera lineal, donde las etapas están claramente definidas, y en el que se puede comenzar con el desarrollo con bastante rapidez, viéndose también la utilidad de su aplicación en proyectos como este, en los cuales no existe todo un equipo de trabajo que deban esperar la finalización de tareas interdependientes.

Por tal razón, este proyecto será determinado por la ejecución de las siguientes etapas:

#### **Análisis de requerimientos**

Comprende la realización de reuniones con técnicos de la División de Oceanografía en las instalaciones del instituto, en donde se escuchan sus experiencias, y se anotan los requerimientos que tienen para que la aplicación ayude a mejorar la toma de datos en campo.

#### **Diseño del sistema**

Descripción de la estructura relacional global de la aplicación, y relaciones entre las entidades que la componen, analizando además las herramientas que se usarán en la etapa de desarrollo.

#### **Desarrollo**

En esta etapa los requisitos especificados por el usuario pasan a ser programados, haciendo uso de las estructuras de datos definidas en el diseño del sistema; comprende la implementación del código fuente y prototipos evolutivos para corregir errores.

### **Implementación**

Comprende la instalación y configuración en el servidor del instituto de los componentes ODK para diseño de formularios y gestión de usuarios y datos; debiéndose también migrar los formularios desarrollados. Por otro lado, en el o los dispositivos móviles asignados, se configurará la aplicación ODK Collect para la recolección de datos empleando los formularios.

### **Pruebas**

Ejecutar pruebas para verificar que los diferentes componentes de la aplicación funcionen correctamente y cumplan con los requisitos del usuario; esta etapa comprende pruebas realizadas dentro de las instalaciones del instituto, así como en actividades de campo.

## **CAPÍTULO 2**

### **MARCO TEÓRICO**

#### **2.1. Modelo de desarrollo basado en componentes**

El continuo avance de la informática y telecomunicaciones ha producido cambios en el desarrollo de aplicaciones de software, aumentando el uso de sistemas distribuidos y abiertos, poniendo en manifiesto un desbordamiento de los modelos de programación existentes, los cuales no son idóneos para operar estos tipos de sistemas de acuerdo al nivel de complejidad que presentan [1].

De allí que se presentan nuevos paradigmas de programación, teniendo entre otros, la coordinación, programación orientada a componentes, o la movilidad, que pretenden generar mejoras en los procesos de desarrollo

de software, trabajando en extender los modelos actuales, así como en nuevos modelos, estandarización de interfaces y servicios, y la globalización de componentes software [1].

Uno de los enfoques actualmente utilizados es el conocido como Desarrollo de Software Basado en Componentes (DSBC), o simplemente Modelo de Desarrollo Basado en Componentes, el cual tiene por objetivo el sentar bases para el diseño y desarrollo de aplicaciones distribuidas basadas en componentes reutilizables de software [1].

Los componentes comerciales de software general, desarrollados por vendedores, ofrecidos como productos, y que son conocidos como COTS (Commercial Off-The-Shelf), ofrecen una funcionalidad que se busca mediante la implementación de interfaces bien definidas, lo que permite que este componente se integre de manera adecuada con el software que se va a desarrollar. Este modelo de desarrollo concentra muchas de las características que están presentes en el modelo de espiral, mostrando un entorno evolutivo, además de un enfoque iterativo para el desarrollo de software [2].

Las actividades de modelado y construcción empiezan identificando los posibles componentes a utilizar, pudiendo estos diseñarse como

módulos de software tradicional, clases orientadas a objetos o paquetes de clases. Cualquiera sea la tecnología que se emplee para crear componentes, este modelo de desarrollo incorpora las siguientes etapas, que se implementan con el uso de un enfoque evolutivo [2]:

1. Investigación y evaluación de productos disponibles basados en componentes, de acuerdo al tipo de aplicación.
2. Considerar aspectos de integración de los componentes.
3. Diseñar la arquitectura del software para integrar los componentes.
4. Integrar los componentes en la arquitectura.
5. Realizar pruebas íntegras para garantizar una apropiada funcionalidad.

#### **2.1.1. Contexto**

En el desarrollo de software, el concepto de componentes no resulta nuevo, ya que muchos autores lo consideran una evolución de la metodología orientada a objetos, tal es así, que de hecho muchas características de los componentes para el desarrollo, provienen de esta [3].

El desarrollo tradicional de aplicaciones suele incurrir en altos costos, así como en una inversión de tiempo extensa. El iniciar el desarrollo de software desde cero supone un reto muy grande aun para grandes empresas, y se lo sesgaría únicamente a nivel de estas, dejando fuera a pequeñas y medianas empresas que deseen adquirir tecnología o desarrollar soluciones propias; esto cambió a inicios de la década de los 90, en donde estas comenzaron a participar en este campo, buscando sus propios productos para no depender de terceros, requiriendo además reducción de costos en hardware, software e infraestructuras de comunicación [3].

DSBC tiene entre sus objetivos la reducción del tiempo de trabajo, del esfuerzo en implementación de aplicaciones, así también del costo del proyecto, aumentando la productividad del equipo de desarrollo y reduciendo los riesgos globales sin incurrir en gastos descomunales. Otra ventaja es la mejor integración de las tecnologías para desarrollar aplicaciones personalizadas, de acuerdo a las necesidades de los clientes [3].

Es así, que las empresas y equipos de desarrollo pueden adquirir tecnología acorde a sus necesidades, sin pagar elevadas

cantidades de dinero por conceptos de licenciamiento, soporte y actualización de las grandes soluciones, en vista que mucha de la tecnología existente es gratis y existe bajo la premisa de Freeware y GNU (General Public License), lo que supone otra gran ventaja [3].

Por último, hay que indicar que el DSBC, es parte del paradigma de programación de sistemas abiertos, los mismos que son extensibles y mantienen interacción con componentes heterogéneos que ingresan o abandonan el sistema de manera dinámica, de tal manera que los componentes pueden ser reemplazados por otros independientemente de su arquitectura y desarrollo [3].

### 2.1.2. Definiciones básicas

**Sistema:** conjunto de mecanismos y herramientas que permiten el desarrollo e interconexión de componentes software, junto con una lista de servicios para facilitar las tareas de los componentes que se encuentran y ejecutan en él [1].

**Sistema independientemente extensible:** implica que puede ser extendido de manera dinámica, y admite combinar extensiones

desarrolladas independientemente por otras partes o entidades, sin conocimiento una de otras [1].

**Sistema abierto:** cuando es concurrente, reactivo, independientemente extensible, que permite el ingreso o salida de forma dinámica del sistema, de componentes heterogéneos; son sistemas inherentemente evolutivos, en donde el tiempo de vida de los componentes es menor a la del propio sistema [1].

**Programación orientada a objetos:** paradigma de programación que combina datos y acciones (métodos) en estructuras lógicas denominadas objetos, aumentando la capacidad de administrar la complejidad del software [4].

**Programación orientada a componentes:** extensión de la programación orientada a objetos para entornos abiertos, la cual defiende el desarrollo y uso de componentes reutilizables dentro de un mercado global de software [1].

### 2.1.3. Concepto de componente

Al momento de definir el DSBC, se debe diferenciar las características del mismo y el paradigma de orientación a objetos.

La programación orientada a objetos es cuando se desarrolla el software empleando modelamiento mental de un objeto real o imaginado, mientras que DSBC indica que el software se debe desarrollar a partir de componentes prefabricados [3].

Para nombrar algunas de las definiciones de componentes, expertos encargados de desarrollar esta metodología, han sentado bases en la metodología orientada a objetos y el modelado UML, teniendo [3]:

- Componente: parte no insustancial, casi independiente y reemplazable en un sistema, que cumple con una funcionalidad dentro de un contexto en una arquitectura claramente definida. Se conforma y provee la ejecución física por un conjunto de interfaces.
- Componente de Software (Definición 1): es un paquete vinculado a uno o varios programas de manera dinámica, operados como una unidad, que son accedidos mediante interfaces bien documentadas que pueden ser descubiertos en tiempo de ejecución.
- Componente de Software (Definición 2): unidad de estructura con interfaces contractualmente especificadas y explícitas, que

poseen dependencias únicamente dentro de un contexto. El componente puede ser desplegado independientemente, y es dependiente a la composición de terceros.

- **Componente de Negocio:** simboliza la implementación de software del concepto de un negocio autónomo, o de un proceso de negocio, el mismo que consiste de artefactos de software necesarios para formular, implementar y poner en producción el concepto de elemento reusable dentro de un sistema de negocios más grande.

Estas definiciones se complementan y construyen el significado no solo de componente sino del DSBC, y más allá de la definición, existen características primordiales para catalogar a un elemento como componente, las mismas que se detallan a continuación [3]:

- **Identificable:** debe poseer identificación para poder clasificarlo y acceder de manera sencilla a sus servicios.
- **Auto contenido:** el componente no necesitará usar otros, para concluir la función que tiene asignada.
- **Reemplazable:** podrá ser reemplazado por otro componente que mejore su función, o por una nueva versión de sí mismo.

- Accesibilidad: acceso solamente a través de su interfaz, motivo por el cual se asegurará que esta no cambie durante su implementación.
- Invariabilidad de servicios: la implementación de la interfaz podrá variar, por las funcionalidades ofrecidas por esta no lo harán.
- Documentado: el componente debe contar con buena documentación, para facilitar su búsqueda, actualización, integración con otros componentes, adaptación, entre otros.
- Genérico: varias aplicaciones pueden usar sus servicios.
- Reutilizable: puede ser cargado dinámicamente en una aplicación, en tiempo de ejecución.
- Independencia: autónomo del hardware, software y sistema operativo.

Existen modelos que permiten entender los componentes de software, para encontrar los necesarios para cada aplicación en particular. El modelo desarrollado por Anneliese Andrews y Sudipto Ghosh, en Colorado State University, propone entender un componente enfocándose en tres aspectos [3]:

- Dominio: representa todos los aspectos del problema del usuario, relacionado de manera directa con la funcionalidad que aporta el componente.
- Programa: aspecto que más varía entre componentes, ya que presenta al detalle la información técnica, como la estructura de archivos, definición de la base de datos, definición de interface de datos, tipos de parámetros, llamada de métodos del componente, entre otros.
- Situación: la información que se determina en este punto será de utilidad para equiparar los requerimientos con las capacidades funcionales y no funcionales del componente; determinando también la estructura de la arquitectura física y conceptual, flujos de información que el componente transfiere, usa y transforma, así como los tipos de algoritmos.

## **2.2. Arquitectura de app móviles**

### **2.2.1. Introducción**

La telefonía móvil ha presentado un creciente auge a nivel mundial, y de acuerdo a los datos ofrecidos en el informe anual Mobile Economy de la GSMA, la asociación que organiza el Mobile World Congress (MWC) del año 2018, al finalizar el 2017 la

cantidad de usuarios únicos de telefonía móvil alcanzó los 5.000 millones, aun cuando el número de tarjetas SIM empleadas por persona (sin incluir la de las máquinas entre sí), se incrementó a 7.800 millones, es decir, el 103% de los habitantes del planeta, superando la población mundial [5].

Para el año 2025 dos tercios de las conexiones móviles a nivel mundial se establecerán en redes 4G y 5G, de lo cual se espera que 4G represente el 53%, mientras que las redes 5G crecerán hasta representar un 15% adicional. La población que posee un dispositivo móvil inteligente (smartphone) alcanzó en 2017 el 57% del total.

Un valor agregado que permitió a la telefonía móvil un alto grado de aceptación por parte de los usuarios, y alta penetración en el mercado, es el poder ofrecer servicios alternos a los canales de voz. Se puede citar como ejemplo, la cuarta generación de telefonía móvil 4G con tecnologías como Long Term Evolution (LTE), LTE Avanzado, LTE Advanced (LTE-A), High Speed Packet Access (HSPA+) y el estándar IEEE 802.16, ofreciendo servicios basados completamente en el Protocolo de Internet (IP), con

velocidades de transferencia hasta de 100 Mbps y con Calidad de Servicios, Quality of Service (QoS) [6].

A la par del crecimiento de las redes, los teléfonos móviles han evolucionado integrando diversas tecnologías como WiFi, Bluetooth, GPS, infrarrojo, entre otras, lo cual permite que el teléfono celular sea compatible con diferentes tipos de dispositivos.

Esta disponibilidad tecnológica crea en las personas una demanda del uso de nuevos servicios, los cuales se basan en estas tecnologías para solucionar problemas en el terreno empresarial, comercial, académicos, de salud y social. Actualmente la mayoría de los servicios móviles están desarrollados en HTML 5, WAP, Java, 2 Micro Edición (J2ME), C#, Silverlight, .NET, entre otros.

Esto exige nuevas tendencias para el desarrollo de software móvil, considerando características especiales de los entornos móviles como el canal de radio, capacidad de los dispositivos, portabilidad, movilidad del usuario entre otras.

### **2.2.2. Información Geográfica Voluntaria y Sistemas de Información Geográfica Participativa**

Con el constante avance de tecnologías como Web 2.0 y GeoWeb 2.0 (extensión geográfica de Web 2.0) a inicios de la década anterior, se puso de manifiesto un caso especial del fenómeno web general, denominado Información Geográfica Voluntaria, conocida también por sus siglas en inglés como VGI (Volunteered Geographic Information), la cual evidencia una transformación profunda en cómo se produce y comparte el conocimiento geográfico [7]; es así que se halla que personas que no tienen conocimiento de Sistemas de Información Geográfica, Cartografía y ciencias similares, terminan siendo contribuyentes de información geográfica en diferentes áreas [8].

Goodchild, quien empleó por primera vez el término VGI [7], compara a la humanidad como “una gran red de sensores móviles inteligentes”, que registran una inmensa y valiosa cantidad de información geográfica.

Por otra parte, los Sistemas de Información Geográfica Participativa, o PPGIS (Public Participation GIS, por sus siglas en inglés) son un tipo de VGI, cuyo concepto surgió en el año de 1996

en el Centro Nacional de Información Geográfica y Análisis, y tiene como fin de llevar las prácticas de los SIG con el objetivo de promover el conocimiento y participación en toma de decisiones a nivel local, en otras palabras, participación ciudadana relacionada con sistemas de información geográfica.

Estos conceptos podrían aplicarse en una segunda fase de la aplicación, en la cual se involucre a la ciudadanía para reportar contaminación en playas, basura marina, muestreo de mesoplásticos, afloramiento de algas, entre otros, lo cual brindaría datos al instituto de investigación, para proponer y coordinar planes de acción por ejemplo a gobiernos autónomos descentralizados del margen costero.

### **2.2.3. Uso de dispositivos móviles para recolección de datos**

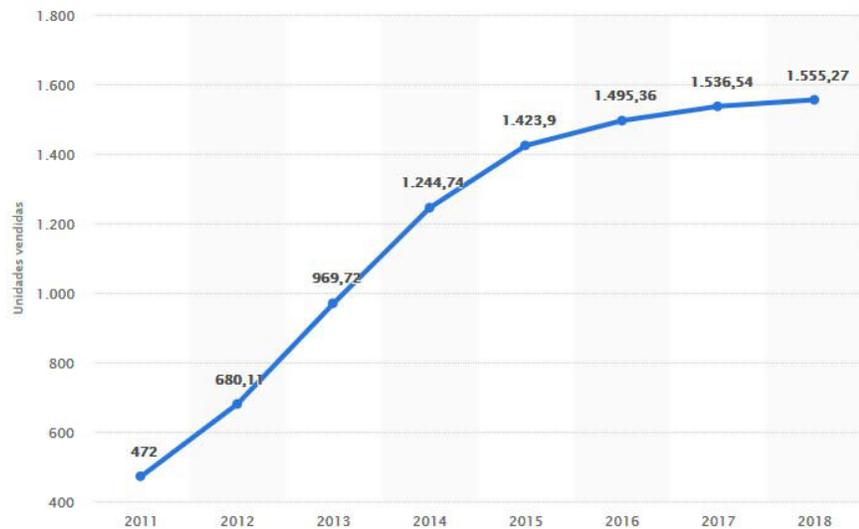
Los teléfonos celulares, dispositivos móviles inteligentes o smartphone, dejaron de ser simples objetos que permiten la comunicación telefónica, para convertirse en la herramienta de una gran red de aplicaciones colaborativas y sensores en constante movimiento; a todo esto, se suma la disminución de costos de los equipos y el aumento en la accesibilidad por parte

de los usuarios, lo cual lo convierte en un dispositivo idóneo para recolección móvil de datos.

La recolección de datos utilizando dispositivos móviles, ha abierto oportunidades para automatizar procesos de forma efectiva empleando mínimos recursos; es así, que la actividad de recolección es fundamental en varios contextos de la investigación como evaluaciones médicas, gestión de riesgos, coberturas de desastres naturales, búsqueda y rescate, catalogación medioambiental, observaciones medioambientales, encuestas socioeconómicas, y para el caso del instituto, sería de utilidad en muestreo de mesoplásticos (basura y desecho marino) por ejemplo, y como propone el presente trabajo, para medición de corrientes marinas y caracterización costera.

El número de smartphones vendidos al usuario final a nivel mundial de 2011 a 2018 ha tenido un crecimiento exponencial, como se puede observar en la Figura 2.1. Esta cifra no resulta extraña a la realidad del Ecuador, ya que según datos proporcionados por la Agencia de Regulación y Control de las Telecomunicaciones (Arcotel) en su boletín estadístico de julio de

2017, el 46.4% de usuarios del Servicio Móvil Avanzado (SMA) poseen un smartphone [9].



**Figura 2.1: Smartphones vendidos a nivel mundial 2011 - 2018**

En esta área se puede citar, que los mecanismos de recopilación de datos pueden conllevar mucho tiempo y esfuerzo; la recopilación de datos, georreferenciación, entrada de datos, procesamiento, depuración, análisis y generación de informes requiere de gran inversión tanto de tiempo y capacitación, sin que dejen de ser exentos de problemas de calidad [10].

La obtención de datos de manera rápida y georreferenciada, es un creciente problema que impacta sobre la oportunidad de realizar

un análisis pertinente. Es así que se evidencia un gran retraso entre la recopilación de datos y el informe de los resultados [10].

En un proceso tradicional de recolección de datos basado en formularios impresos, con uso de equipo GPS para obtener coordenadas del sitio de la observación y cámara fotográfica para registro visual de evidencias, existe mayor susceptibilidad de errores y consumo de tiempo, lo que representa problemas de calidad y consistencia de los datos, debido especialmente a la separación de actividades de recopilación e ingreso de datos [10].

Encuestadores, técnicos, investigadores y demás, escribirán los resultados de su muestreo en cuestionarios de papel, y al final los datos del proceso de recolección son entregados a un equipo para su ingreso en una base de datos, en el mejor de los casos. A esto se añade el esfuerzo extra que implica mantener relación entre el registro fotográfico y registro digital del formulario ingresado.

De esto, se anotan problemas adicionales como [10]:

- Los formularios tradicionales requieren el recurso del papel y la impresión, no resulta sencillo ni ágil en ciertas ocasiones realizar cambios a los mismos.

- Formularios que no llegan en óptimas condiciones (mojados o anotaciones borradas, por ejemplo) a su destino, dificultando la actividad del equipo de ingreso, debido a la mala interpretación de la escritura de los registros, provocando además fallas en la validación de los datos.
- Problemas con respecto a la ubicación, lo cual es solicitado por los formularios impresos, presentando ambigüedades e inconsistencias en el nombre de las localidades; el autor anota que el uso de GPS portátil no fue totalmente satisfactorio por complicaciones con el formato de coordenadas, y a nivel físico, por falla en conexiones de cable, por ejemplo.

El uso de formularios tradicionales, tal como lo muestran la Figura 2.2 y Figura 2.3, se ha dado en el instituto desde sus inicios, motivo por el cual las diferentes direcciones plantearon la necesidad de contar con un mecanismo para sistematizar estos formularios, y que se ajusten a las necesidades de los técnicos e investigadores en los trabajos en campo.



Estos problemas pueden ser solucionados mediante el uso de dispositivos móviles inteligentes en la recolección de datos en campo, integrándose todo en un solo procedimiento, teniendo como beneficios principales: calidad en los datos recolectados y reducción de recursos (tiempo y costos), gracias a la mejor disposición para ingreso de información mediante la pantalla táctil del dispositivo, sensor GPS para posicionamiento geográfico y cámara para registro visual.

Implementando una solución en dispositivos móviles inteligentes, la recolección de datos mejora y por consiguiente su calidad, pero se tiene también características asociadas a estos formularios, como se detalla a continuación:

- Permiten cambios de manera rápida y son diseñados dinámicamente.
- Admiten procesos de validación de campos obligatorios, así como de tipos de datos.
- El posicionamiento geográfico en la recolección de datos, es automáticamente registrado por el sensor GPS del dispositivo móvil.

- Además del registro fotográfico, se puede vincular automáticamente al formulario, audio o video desde el dispositivo.
- El error humano disminuye al mínimo, ya que los formularios se sistematizan.
- Los datos se pueden centralizar en un sistema, y emplearlos para su visualización, análisis y generación de reportes.

Aun cuando existen alternativas para el diseño e implementación de formularios para dispositivos móviles (desarrollo nativo en Android, Adobe Experience Manager Forms, Collector for ArcGis, entre otros) el presente trabajo se centrará en el uso de la plataforma ODK, que ha sido elegida por su facilidad de uso, además de ser un software open source.

#### **2.2.4. La Plataforma Open Data Kit (ODK)**

Con el paso del tiempo aumenta la necesidad de las organizaciones de recolectar datos empleando dispositivos móviles para su posterior presentación en Internet, siendo la facilidad de uso para la recolección de datos por parte del usuario

final y la rápida distribución a nivel global la principal razón de este incremento [11].

La comunidad que colabora con ODK genera software gratuito y de código abierto para recopilación, administración y utilización de datos en entornos con recursos limitados [12]; es así que permite la recopilación de datos en modo off line y enviar estos datos cuando exista conexión a Internet [13], brindando a las comunidades o usuarios agregar datos, con control total de lo recopilado así como de los servidores donde se almacenan estos datos.

ODK es una herramienta muy popular en zonas en desarrollo o sectores rurales al momento de realizar encuestas, y se ha adaptado a diferentes escenarios, como se anotan: funcionarios de gobierno que completan encuestas socioeconómicas sobre hogares en un distrito o área, prestadores de salud comunitaria gestionando visitas domiciliarias, aplicaciones para soporte a la toma de decisiones médicas en administración de pruebas, maestros que implementan actividades lúdicas con preguntas interactivas y tutoriales de respuestas y grabación automática de puntajes, funcionarios que tras terremotos se encuentran en zonas

críticas capturando imágenes y ubicaciones de zonas afectadas [12].

La plataforma ODK brinda a las organizaciones soluciones para la recolección y manejo de datos móviles. En una primera definición básica, se puede decir que permite el diseño y desarrollo de formularios o encuestas de manera sencilla, recolección de datos desde un dispositivo móvil para luego ser enviados a un servidor, extracción y consulta de estos datos desde el servidor, y por último exportación a diferentes formatos para su uso.

En un análisis efectuado a diferentes alternativas para la recolección de datos empleando dispositivos móviles, el autor menciona en primer lugar aplicaciones móviles precompiladas como Epic Collect, Arc Gis Mobile Solutions by Esri, pasando también por la discusión acerca de escenarios que contemplan el desarrollo de una aplicación nativa para Android e iOS, en donde se describen las fortalezas y debilidades de cada alternativa [11].

Se efectuó una reunión de acercamiento con la Dirección de Oceanografía Naval en donde participaron técnicos del área, luego de lo cual se eligió a la plataforma ODK para el desarrollo del

presente proyecto como una solución para la recolección de datos móviles, considerando las siguientes razones:

- Es una plataforma de tipo Open Source, ajustándose a la política pública de uso de software libre en instituciones públicas del Ecuador.
- Posee facilidad de instalación y uso.
- Es una solución completa que permite la creación de formularios, recolección y manejo de datos en el servidor.
- Se implementan los formularios de acuerdo a las necesidades del usuario, con relativa rapidez.
- Se puede implementar formularios que permiten atributos multimedia y geolocalización.
- Soporta recolección de datos fuera de línea.
- Ofrece diferentes escenarios de implementación.

Aplicado al presente proyecto, el uso de la plataforma ODK se ajusta de manera perfecta, ya que, de acuerdo a las necesidades, su implementación se puede realizar en diferentes escenarios y configuraciones, siendo una herramienta versátil que permite interoperabilidad mediante el uso de los componentes de la plataforma de forma separada, con costos de implementación

mínimos, presentando interfaces sencillas y amigables lo que supone además un proceso de aprendizaje menor para el usuario final.

Como se describe en un análisis efectuado, otras soluciones presentan factores limitantes similares, es decir, fueron construidos como monolíticos, siendo soluciones aisladas que utilizan formatos de datos e interfaces patentados, siendo algunos de ellos construidos para plataformas de hardware específicas que requieren un rediseño a gran escala para actualizar a nuevas tecnologías; de allí que ODK tiene como requisitos [12]:

- Componentes modulares, permitiendo de acuerdo a las necesidades específicas de un proyecto, que sea fácilmente extendido y modificado.
- Open Source, con utilización de software de código abierto e interfaces basadas en estándares abiertos, lo cual permite una mayor participación de la comunidad de desarrolladores.
- Tecnología de vanguardia, desarrollando aplicaciones en sistemas que persistan y evolucionen a largo plazo, adaptándose a nuevas capacidades.

Las herramientas de núcleo de la plataforma ODK que se emplearán en este proyecto son [14]:

- ODK Build y ODK XLSForm, para diseño y generación de formularios.
- ODK Collect, para recolección de datos en dispositivos móviles.
- ODK Aggregate, para manejo en el servidor de los datos recolectados.

En la Figura 2.4 se presenta el flujo de trabajo del diseño de encuestas y recolección de datos empleando la plataforma Open Data Kit.



**Figura 2.4: Flujo de trabajo en recolección de datos con ODK**

### 2.2.5. ODK Build y ODK XLSForm

ODK ofrece a los usuarios una aplicación web basada en HTML5 que usa Javascript y Ruby Rack, ODK Build, el cual es un diseñador de formularios que brinda una interfaz de “arrastrar y soltar” elementos hacia un canvas en blanco para luego exportarlo a formato XML, el mismo que funciona de mejor manera para el diseño de formularios sencillos [15]; los usuarios pueden reorganizar el orden de las indicaciones o agregar lógica personalizada a cada solicitud en el diseño de formularios.

Para el despliegue de los formularios en ODK y para garantizar que las herramientas se puedan utilizar de forma independiente, pero también entre sí, la plataforma emplea el estándar abierto XForms del W3C, el cual es un estándar de descripción de formulario basado en XML, que permite generar formularios con cualquier aplicación que implemente XForms [16].

Para el diseño e implementación de formularios en el estándar XForms se puede optar como alternativas: el desarrollar los formularios empleando un editor de texto siguiendo el estándar, mediante el uso de una herramienta gráfica de diseño de formularios como ODK Build o PurcForms, o utilizar XLSForm [17],

que es un estándar desarrollado para ayudar a simplificar la creación de formularios en hojas de cálculo Excel, permitiendo generar además formularios de tipos más complejos. De hecho, ODK lo recomienda como un diseñador de formularios más potentes [14].

Por último, para la personalización o cambios en los formularios, se puede emplear un editor de texto para editar los archivos XML generados con cualquiera de las herramientas mencionadas.

La Figura 2.5 muestra un ejemplo de pantalla de ODK Build. En el presente proyecto se utilizó ODK Build 0.3.5, y en modo prueba ODK XLSForm Offline v1.9.0.



Figura 2.5: Ejemplo de pantalla de ODK Build

### 2.2.6. ODK Collect

ODK Collect es una aplicación Android de código abierto, que permite reemplazar formularios de papel en encuestas y actividades de recolección de datos, permitiendo una diversa variedad de preguntas y respuestas, poseyendo como característica especial, el funcionar bien sin tener conectividad de red [18].

Esta aplicación se encarga del procesamiento de los formularios siguiendo una secuencia de solicitudes de entrada aplicando la lógica del formulario, subestructuras repetitivas, así como restricciones de entrada. Los usuarios proceden de acuerdo a las indicaciones, pudiendo almacenar el envío en cualquier momento; los envíos finalizados pueden reenviarse a (así como la descarga de nuevos formularios desde), un servidor.

ODK Collect admite formularios con una amplia variedad de tipos de preguntas. La funcionalidad exacta y el estilo de visualización de cada pregunta se especifican en la definición XLSForm; Collect admite ubicación, audio, imágenes, video, códigos de barras, firmas, opciones múltiples, texto libre y respuestas numéricas, e

inclusive puede aceptar respuestas de otras aplicaciones del dispositivo móvil [18].

Collect recibe la lógica XForm y presenta indicaciones al usuario en un formato de solicitud rápida y los usuarios navegan en la aplicación hacia adelante y hacia atrás moviendo el dedo por la pantalla, como pasando páginas de un libro. La recopilación admite también una función de salto a solicitud que permite al usuario saltar hacia adelante o hacia atrás rápidamente dentro de la aplicación.

Los avisos se pueden especificar como uno de los siguientes tipos: texto libre, entero, decimal, selección individual (botones de radio), selección múltiple (casillas de verificación), imagen, audio, video, código de barras, o ubicación geográfica mediante GPS [12].

Una vez descargado e instalado, debe ser configurado en un sencillo paso, el cual consiste en especificar la dirección URL del servidor ODK Aggregate y credenciales del usuario (de ser necesario), luego se procede a descargar los formularios, e iniciar con el proceso de recolección de datos.

La Figura 2.6 muestra un ejemplo de pantalla de ODK Collect. En el presente proyecto se utilizó ODK Collect v1.19.0.

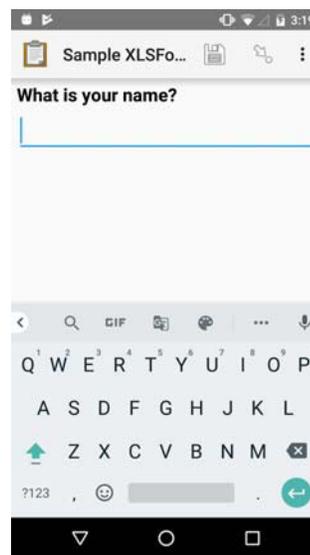


Figura 2.6: Ejemplo de pantalla de ODK Collect

### 2.2.7. ODK Aggregate

ODK Aggregate provee de un servidor listo para despliegue, el cual puede realizar la carga, almacenamiento y transferencia de datos a servidores locales o a la nube. ODK Aggregate es una aplicación de código abierto desarrollada en entorno Java, la cual almacena, analiza y muestra datos de encuestas XForm recopilados utilizando ODK Collect u otras aplicaciones compatibles con OpenRosa. Soporta variados tipos de datos y

está diseñado para funcionar bien en cualquier entorno de alojamiento [19].

Proporciona un servidor de repositorio para administrar los datos recopilados, proporciona interfaces estándar para extracción de datos (hojas de cálculo, consultas, entre otros) e integrarse con los sistemas existentes a través de solicitudes web HTTP. Así también, Aggregate admite el retorno de datos en diversos formatos estándar, incluidos CSV (para análisis estadístico), KML (del acrónimo en inglés Keyhole Markup Language) y JSON (del acrónimo en inglés JavaScript Object Notation) para transporte a otros servicios web [12].

Aggregate está diseñado para ser un servicio genérico de almacenamiento de datos que se ejecutará en la plataforma informática elegida por el usuario, siendo importante destacar que puede recibir datos de cualquier teléfono y servidor que sea compatible con OpenRosa.

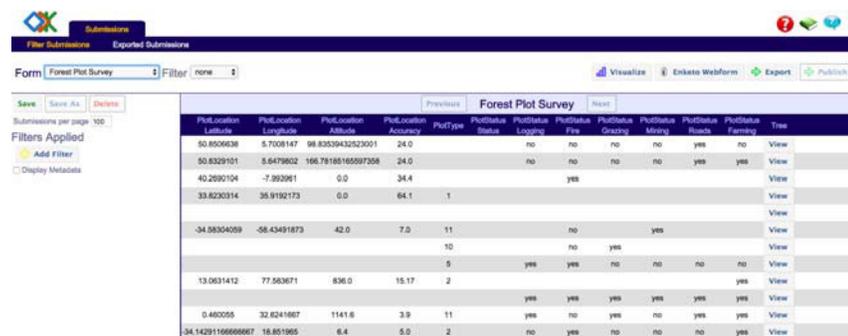
Como se indicó anteriormente, ODK Aggregate tiene un diseño listo para ser desplegado en un servidor de aplicaciones Java estándar, permitiendo su arquitectura que se ha implementado en

varios escenarios, los cuales dependerán exclusivamente de las necesidades de cada proyecto, como se detalla a continuación:

- Desplegar la aplicación mediante servicios basados en la nube, modalidad Plataforma como Servicio (PaaS), pudiéndose emplear Google App Engine o Amazon Elastic Compute Cloud (Amazon EC2), los cuales brindan un ambiente ideal para organizaciones que no cuentan con infraestructura tecnológica ni de recursos humanos que se encarguen de la administración tanto de la plataforma como de la base de datos.
- Desplegar la aplicación en infraestructura propia, en un ambiente local, para lo cual se deberá contar con un equipo dedicado en el cual se instalará un servidor de aplicaciones Java, servidor de base de datos (MySQL o PostgreSQL), una conexión estable y de buena velocidad a Internet, lo cual permitirá brindar una arquitectura ideal a organizaciones con políticas de privacidad de datos, las cuales tendrán total control del acceso a la plataforma y los datos. Esta arquitectura será la que se empleará en este proyecto, en vista que el instituto, facilitará un servidor de pruebas en donde se instalará la plataforma ODK y demás componentes necesarios para la implementación de la aplicación.

Por último, se comenta que existe en el sitio web oficial de ODK una máquina virtual preconfigurada disponible para descarga de los usuarios, la cual permite desplegar de manera fácil y en menor tiempo, un servidor con la aplicación instalada, la cual puede ser ejecutada en cualquier equipo o infraestructura.

La Figura 2.7 muestra un ejemplo de pantalla de ODK Aggregate. En el presente proyecto se utilizó ODK Aggregate v1.4.10.



The screenshot shows the ODK Aggregate interface for a survey titled 'Forest Plot Survey'. The interface includes a header with the ODK logo, 'Submissions' tab, and 'Exported Submissions' link. Below the header, there are options to 'Visualize', 'Enketo Webform', 'Export', and 'Publish'. A sidebar on the left contains 'Save', 'Save As', 'Delete', 'Filters Applied', 'Add Filter', and 'Display Metadata'. The main area displays a table of submissions with columns for Plot Location (Latitude, Longitude, Altitude, Accuracy), Plot Type, and various Plot Statuses (Logging, Fly, Grazing, Mining, Roads, Farming). Each row includes a 'View' link.

Plot Location Latitude	Plot Location Longitude	Plot Location Altitude	Plot Location Accuracy	Plot Type	Plot Status Status	Plot Status Logging	Plot Status Fly	Plot Status Grazing	Plot Status Mining	Plot Status Roads	Plot Status Farming	View
50.800638	5.7008147	98.8303943202001	24.0		no	no	no	no	no	yes	no	View
50.8329101	5.6479602	166.79185165097358	24.0		no	no	no	no	no	yes	yes	View
40.2990104	-7.892981	0.0	34.4			yes						View
33.8230314	35.9192173	0.0	64.1	1								View
-34.58304059	-58.43491873	42.0	7.0	11		no		yes				View
				10		no	yes					View
				5	yes	yes	no	no	no	no	no	View
13.0631412	77.563671	836.0	15.17	2	yes	yes	yes	yes	yes	yes	yes	View
0.460255	32.8241667	1141.6	3.9	11	yes	no	yes	no	yes	yes	yes	View
34.1429118666667	18.851965	6.4	5.0	2	no	yes	no	no	no	no	yes	View

Figura 2.7: Ejemplo de pantalla de ODK Aggregate

## 2.3. Arquitectura de aplicaciones web

### 2.3.1. Infraestructura de Datos Espaciales y Geo Servicios Web

Con la introducción de la Web 2.0 y GeoWeb 2.0 la forma en que la información geográfica es generada, compartida y consumida ha cambiado de manera acelerada [8], y en ambos ámbitos

abiertos y participativos, ha aumentado la necesidad de integración de información de carácter geográfico con aplicaciones web llamadas Web Mapping Applications.

En una breve reseña, se puede anotar que la entrega de información geográfica a través de la Web empezó en el año de 1993 con el desarrollo del Xerox PARC Map Viewer, el cual presentaba un mapa mundial que permitía ampliarlo a escalas predefinidas, cambiar la proyección y controlar la visibilidad de ríos y características fronterizas; tecnología muy vanguardista que simplemente consistía en una página web HTML, en la cual se incrustaba un archivo de imagen representando el mapa, y la interacción del usuario con el mapa se implementó mediante un script CGI (Common Gateway Interface), escrito en Perl, que se ejecutó en el servidor web [8].

Este modelo de interacción era típico de la denominada primera generación de aplicaciones de mapeo web (Web Mapping Applications), que han aumentado rápidamente después del experimento Xerox PARC Map Viewer y se mantuvo en uso hasta fines de la década de 1990.

Es aquí donde las Infraestructuras de Datos Espaciales (IDE) tienen gran importancia, ya que son “un conjunto de normas, políticas y estándares cuyo fin es la publicación y representación en Internet de datos, metadatos y servicios de forma estándar, garantizando la interoperabilidad de estos permitiendo el acceso a dicha información de una forma rápida y oportuna” [20].

Uno de los componentes de las IDE son los estándares tecnológicos que permiten compartir y acceder a información geográfica, basados en la interoperabilidad; dichos estándares son implementados en el desarrollo de software o servicios, abiertos o privados, y son creados a través de especificaciones de documentos técnicos.

Gran parte de las IDE adoptan los estándares establecidos por el Open Geospatial Consortium (OGC), fundado en 1994 para hacer de la información geográfica una parte integral de la infraestructura mundial de información [21]. Estos estándares facilitan el uso, visualización y descarga de información geográfica, independiente de la tecnología empleada por el usuario.

Estos estándares son referidos colectivamente como OGC Web Services (OWS), y serán empleados a través de los servicios web implementados en el presente proyecto; son accedidos para su operación mediante protocolo HTTP (HyperText Transfer Protocol), mediante una llamada URL (Uniform Resource Locator):

- **Web Map Service:** proporciona una interfaz HTTP para la solicitud de imágenes digitales georreferenciadas desde una o más bases de datos, las cuales son devueltas en formato JPEG, PNG y otros, para su visualización en buscadores y aplicaciones de escritorio [22].
- **Web Feature Service (WFS):** provee interfaz de entrega de datos, definiendo las interfaces de acceso y manipulación de datos, pudiendo los usuarios y servicios consultar, editar y descargar datos geográficos en formatos como GML, ESRI Shape File, KML, GeoJSON, CVS, entre otros [23].
- **Web Coverage Service (WCS):** define el estándar de interface y operaciones permitiendo el acceso interoperable a “coberturas” geoespaciales, es decir, en formato raster con datos espacio/tiempo de un fenómeno asociado; se refiere típicamente a contenidos del tipo imágenes de satélite, fotos aéreas digitales y datos digitales de elevación [24].

### **2.3.2. GeoServer**

Servidor web de código abierto que permite publicar mapas y datos de diversos formatos a clientes estándar, tales como navegadores web y programas GIS de escritorio, de tal manera que se pueden almacenar datos espaciales en casi cualquier formato, y el usuario final no necesita saber nada sobre datos GIS, necesitando únicamente un navegador web para desplegar exactamente los mapas.

Mediante interfaces basadas en estándares, como WMS, WFS, WCS, WPS, Tile Caching y más, los datos son publicados, y GeoServer presenta una interfaz de administración basada en un navegador web, pudiendo conectarse a múltiples fuentes de datos en el back-end [25].

Como características principales se puede anotar: es un servidor de datos de una diversidad de almacenamiento de datos, los datos son atendidos como imágenes rápidas y seguras, empleando protocolos WMS y WMTS, datos vectoriales completos pueden ser enviados a un cliente usando el protocolo WFS, datos raster pueden ser enviados a un cliente utilizando el protocolo de la WCS y servicios web de procesamiento (WPS) [25].

### **2.3.3. Geodatabase**

Ofrece la posibilidad de almacenar, gestionar y mantener datos espaciales, facilitando el uso de relaciones espaciales entre los tipos de datos básicos de información como vector, raster y atributos, en vista que ofrecen la posibilidad de representación geográfica.

En una definición más formal de acuerdo a Esri: “En su nivel más básico, una geodatabase es un conjunto de datasets geográficos de distintas clases que están almacenados en una carpeta común del sistema de archivos o en un sistema de administración de bases de datos relacionales (como Oracle, Microsoft SQL Server, PostgreSQL, IBM Informix o IBM Db2)” [26].

### **2.3.4. PostgreSQL**

De acuerdo al sitio web oficial de PostgreSQL [27], “PostgreSQL es un potente sistema de base de datos relacional de objetos de código abierto que usa y amplía el lenguaje SQL combinado con muchas características que almacenan y escalan de manera segura las cargas de trabajo de datos más complicadas”. Los inicios de PostgreSQL se remontan a 1986, como parte del

proyecto POSTGRES en la Universidad de California en Berkeley y tiene más de 30 años de desarrollo activo en el núcleo de la plataforma.

PostgreSQL posee como características: la herencia de tablas (clases), un conjunto amplio de tipos de datos que incluyen arreglos, Binary Large Object Block (BLOB), tipos geométricos y direcciones de red, incluyendo también, el procesamiento de transacciones, integridad referencial y procedimientos almacenados. Se ejecuta en todos los principales sistemas operativos, cumple con ACID (del acrónimo en inglés Atomicity, Consistency, Isolation, Durability) desde 2001 y tiene complementos poderosos como el extensor de base de datos geoespaciales PostGIS [27].

### **2.3.5. PostGIS**

De acuerdo al sitio web oficial de PostGIS [28], “PostGIS habilita espacialmente la popular base de datos relacional de objetos PostgreSQL, lo que le permite ser utilizada como una base de datos back-end para sistemas de información geográfica (GIS) y aplicaciones de mapeo web”.

PostGIS está publicada bajo licencia GNU General Public License, y desde el 2006 está certificada por el Open GeoSpatial Consortium (OGC), con lo cual se garantiza interoperabilidad con otros sistemas.

PostGIS es estable, rápida, compatible con los estándares, con cientos de funciones espaciales y actualmente es la base de datos espacial de código abierto más utilizada [28].

En la Figura 2.8 se muestra la arquitectura tecnológica propuesta, la misma que está dividida en componentes del lado del servidor, y componente en el lado del cliente.

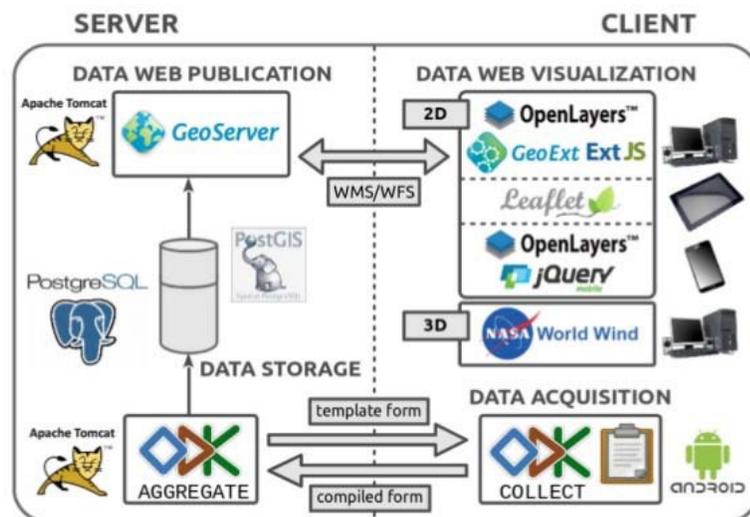


Figura 2.8: Arquitectura tecnológica propuesta

## 2.4. Metodología de desarrollo

Para desarrollo de este proyecto se empleó el modelo en cascada, el cual es un proceso de desarrollo secuencial, siendo un conjunto de etapas que se ejecutan una a continuación de otra:

- **Análisis de requerimientos:** consiste en realizar el análisis de las necesidades de los usuarios, presentar formulario de preguntas, escuchar sus expectativas y experiencias de trabajo en campo, lo cual permitirá determinar las características de la aplicación a desarrollar, especificando también las funcionalidades que presentará sin ahondar en mayores detalles técnicos. Al emplear esta metodología, se debe ser precisos en el registro de requerimientos, ya que no se aceptan nuevos requisitos en la fase de desarrollo.
- **Diseño de la aplicación:** descripción de la aplicación y entidades que la conforman, se organiza y descompone todos los elementos que componen el sistema (en un ambiente grupal, se asigna a cada miembro uno de estos elementos). Así también se desarrolla el diseño arquitectónico, donde se define la estructura de la solución, y diseño detallado en donde se definen algoritmos y organización de código.
- **Desarrollo:** se programan todos los requisitos especificados previamente, comprende la implementación del código fuente y prototipos evolutivos para corregir errores.

- **Implementación:** se realiza la instalación y configuración en el servidor del instituto de los componentes ODK; debiéndose también migrar los formularios desarrollados, configurar en el o los dispositivos móviles asignados la aplicación ODK Collect para la recolección de datos empleando los formularios.
- **Pruebas:** consiste en verificar que los componentes de la aplicación funcionen de manera adecuada, y cumplan con los requisitos del usuario. Con esto se obtiene información de la aplicación desarrollada que permite: encontrar defectos, aumentar la calidad de la aplicación, optimizar el código, entre otros.

## **CAPÍTULO 3**

### **DEFINICIÓN DE LA SITUACIÓN ACTUAL Y REQUERIMIENTOS**

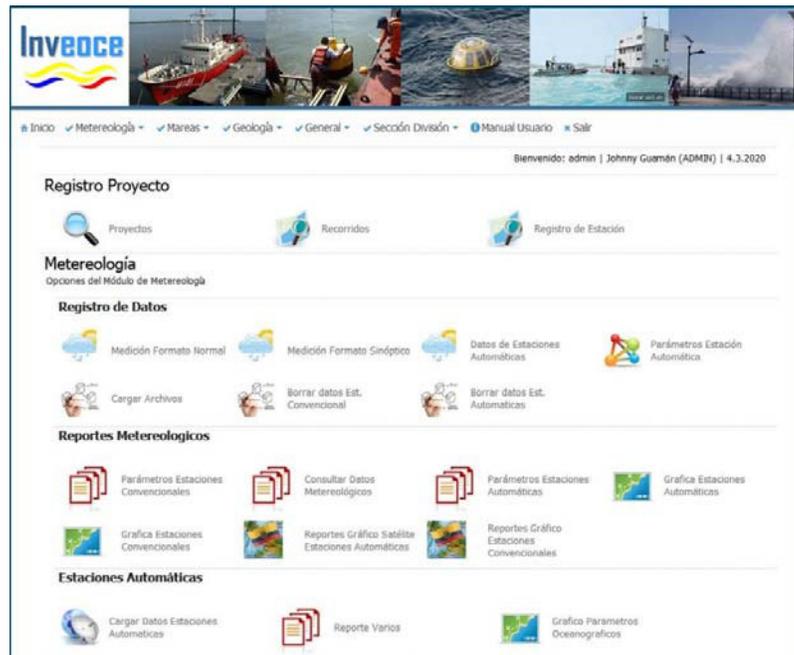
#### **3.1. Definición de la situación actual**

El instituto de investigación oceanográfica, es una entidad pública, razón por la cual para cumplir con el decreto que recomienda el uso de estándares abiertos, mantiene cierto porcentaje de sus sistemas informáticos empleando software libre, ya sean desarrollados en su totalidad, o integrando diferentes componentes.

Para la gestión y administración de datos técnicos - científicos, posee un sistema de información hidrográfico y oceanográfico, desarrollado por personal de la institución; la concepción de este sistema data del año 2001, y fue desarrollado basándose en la necesidad de recolectar y

almacenar los datos que disponían los diferentes departamentos técnicos (hoy direcciones), sean en archivos, en hojas de cálculo, en papel, sin aplicación de métodos que aseguren la calidad de los datos, aislados por almacenarse en archivos y no en un esquema de base de datos.

Está desarrollado en plataforma y lenguaje Java, con servidor de aplicaciones Java EE de código abierto JBoss y almacenamiento centralizado en una base de datos Oracle 11g. En sus inicios se empleaba Oracle Developer 6i y Oracle 9i, teniendo en la actualidad una interfaz web (repotenciación del sistema, año 2014), como se presenta en la Figura 3.1. Cuenta con diferentes módulos para la obtención, carga, procesamiento y consulta de datos de proyectos multidisciplinarios, equipos de medición, estaciones y sensores remotos y trabajos en campo de las diferentes direcciones técnicas, una de ellas la de Oceanografía Naval.



**Figura 3.1: Sistema de información del instituto**

De este sistema diariamente se genera una determinada cantidad de información procesada, la misma que de acuerdo al requerimiento del usuario, será visualizada por pantalla o servirá para la generación de reportes de los diferentes parámetros medidos, siendo la fase final el análisis por parte del técnico o científico.

Con respecto al ámbito del presente proyecto, se evidenció que en el sistema no se ha implementado un módulo que gestione datos de caracterización costera, sin embargo, existe un módulo como lo presenta la Figura 3.2 que permite la carga de archivos de medición de corrientes

marinas (veleteo) en formato Excel, pero no se tiene otra opción asociada a esta actividad y parámetro, es decir, no existe visualización de los datos, y los mismos permanecen en el repositorio central, sin que se les de uso (visualización o reporte).

The screenshot shows the 'Carga de Archivos al Sistema' (Upload Files to System) interface. At the top, there is a navigation menu with options: Inicio, Metereología, Mareas, Geología, General, Sección División, Manual Usuario, and Salir. Below the menu, the title 'Carga de Archivos al Sistema' is displayed, followed by the instruction 'Seleccione el proyecto donde vas a almacenar los datos'. The form contains the following fields:

- Proyecto:** CO-01-2000 20000001 PRIMER CRUCERO OCEANOGRAFICO DEL 2000 EN PRODUCCION. A 'buscar proyecto' button is located to the right.
- Forma Recolección:** 2000-05-08 CO DATOS METEREOLÓGICOS SUPERFICIAL.
- Estación:** -3.1 -80.1. Below this, there are two coordinate input fields: 2000-05-09, 03° 06' 00.00" SUR, and 080° 06' 00.00" OESTE.
- Tipo de Carga:** Corriente Veleta.
- Archivo:** Examinar... No se ...chivo.
- Fecha Desde:(Solo Globo):** An empty date input field.
- Superficial/Fondo:(adcp):** Superficial.

At the bottom of the form, there are two buttons: 'Cargar Archivo' and 'Cancelar'.

**Figura 3.2: Carga de archivos de medición de corrientes marinas**

El sistema permite automatizar tareas de procesamiento de datos de las diferentes direcciones técnicas, y su gran objetivo es contar con un módulo automático y completamente dinámico que reconozca los diferentes formatos de archivos, para obtener datos de múltiples dispositivos y permitir su carga en la base centralizada, pero esto no ha sido posible durante el paso del tiempo, por falta de recursos económicos y humanos, que permitan la implementación y mejora del sistema.

En muchas de las actividades cotidianas, se encuentra a funcionarios utilizando hojas de cálculo y formado con estas (según sus propias palabras), “bases de datos locales” en la estación de trabajo, sin considerarse ningún tipo de control de calidad, con una considerable probabilidad de pérdidas de información por daños del dispositivo de almacenamiento en el computador, sin que exista tan siquiera una política de respaldo de archivos, en procura de salvaguardar los datos contenidos en los mismos.

Como se explica más adelante, en ambos procesos, el personal encargado de la recolección de datos en actividades de campo, realiza el ingreso y procesamiento de datos empleando mayoritariamente Microsoft Excel, implementando en el mismo las fórmulas necesarias, usando luego estos datos para la generar tablas y gráficos que serán sustento de los informes técnicos emitidos por la Dirección.

### **3.2. Levantamiento de información de los procesos actuales**

#### **3.2.1. Medición de corrientes marinas**

Desde tiempos inmemorables. los marineros han requerido conocer la velocidad y dirección de las corrientes oceánicas o marinas para conducir sus embarcaciones dentro de los puertos,

así como a lo largo de las rutas comerciales y de exploración; de tal manera que un navegante debe ser capaz de medir la velocidad de las corrientes observando la distancia, el tiempo y la dirección [29].

Las mediciones de corrientes marinas se pueden efectuar con objetos a la deriva y midiendo el camino seguido por los fluidos, lo cual se denomina "mediciones lagrangianas", nombradas en honor al matemático Joseph Louis Lagrange, o también empleando "mediciones Eulerianas", por el matemático suizo Leonhard Euler, las cuales implican describir el flujo del fluido midiendo la velocidad y la dirección del fluido en un solo punto [29].

La medición de corrientes marinas en el instituto, se realiza empleando el también denominado veleteo, que es una metodología que se aplica para medir corrientes a nivel superficial, y es conocida por el método lagrangiano o Lagrange. Esta metodología se aplica cuando el área no es propicia para medir las corrientes con un equipo electrónico, además de realizarse las mediciones en áreas cercanas a la costa, y no en alta mar porque los cambios son mínimos.

Este método consiste en seguir la trayectoria del movimiento de una parcela de agua, y para aplicarla se utilizan flotadores o derivadores, y en el caso del instituto, las veletas (de ahí el nombre de veleteo), las que se diferencian en la estructura y color para hacer mediciones propiamente superficiales, y mediciones con cierta profundidad en ciertos casos.

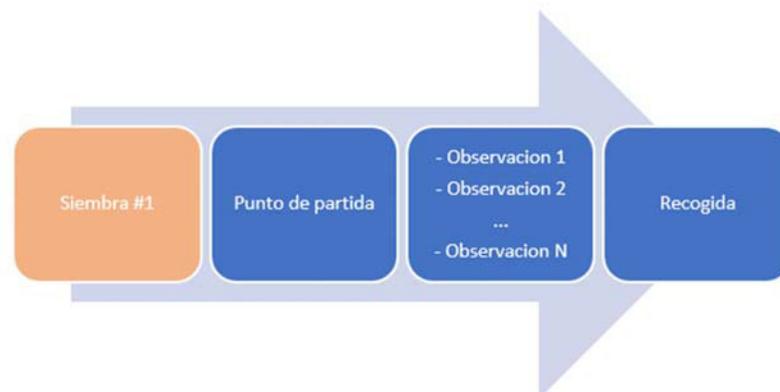
Se coloca una de estas veletas o derivadores sobre la superficie del agua (mar, estero o río), se marca el tiempo (hora, minutos y segundos) en la que se la soltó, y se toma la información del punto de ubicación, en coordenadas UTM (este y norte), y se la deja que viaje sola por un tiempo determinado, el cual puede ser entre 5 o 20 minutos lo cual dependerá de la velocidad en la que viaje esta veleta.

Transcurridos estos minutos, se marca otra vez el tiempo y se toma la nueva posición, y con estos datos: tiempo 1, tiempo 2, y posición 1, posición 2, se puede calcular la velocidad (expresada en m/s) con la que se movió la veleta del punto inicial al punto final, pudiéndose además calcular el ángulo que siguió, lo que vendría a ser la dirección de la corriente; la magnitud de la corriente puede

ser expresada como vector con dirección, o mediante sus componentes X y Y.

Este método se debe realizar con por lo menos 6 veletas, y durante un periodo de mínimo de 8 horas de monitoreo, lo cual produce un buen estimativo de las corrientes superficiales de la zona que se quiere estudiar.

Dicho esto, el veleteo como procedimiento tiene 3 fases que se pueden repetir, las que se identifican como: punto de partida, observaciones y recogidas. Cada fase está relacionada a una siembra que es un ciclo que contiene las 3 fases y que se va contando dependiendo del tiempo de muestreo en el día, como se representa en la Figura 3.3.



**Figura 3.3: Fases del procedimiento de medición de corrientes (veleteo)**

Una vez que se ha finalizado con el trabajo de campo, todos los datos se los pasa a una hoja de cálculo de Excel, separando las veletas por colores, se procede a calcular la velocidad y dirección si se tiene coordenadas UTM, y en caso de tener coordenadas geográficas, se debe primero pasar a coordenadas UTM (metros), cálculo de velocidad y distancia (usando las posiciones), calculando además cada la magnitud de cada tramo de la veleta. La dirección es calculada empleando el Arcotangente, realizando este proceso con cada tramo de veleta y cada una de las veletas.

Se procede a generar los gráficos correspondientes empleando software especializado como Matlab u ODV (Ocean Data View) para observar las trayectorias seguidas por cada veleta, separando flujo (marea subiendo) y reflujos (marea bajando). Con los gráficos listos, se realizan tablas indicando las máximas y mínimas magnitudes con su dirección, y los promedios tanto en flujo y reflujos, para concluir el análisis con la redacción del informe empleando datos de mareas, vientos y batimetría (medición de las profundidades marinas) del área de estudio.

Es de anotar también, que obtenida información de las corrientes, esta es graficada en ArcMap, utilizando como mapa base una carta batimétrica digital del área.

### **3.2.2. Caracterización costera**

Estos estudios tienden a conocer la variabilidad espacio-temporal de diferentes parámetros que comprenden la dinámica y los procesos biológicos, ecológicos, físicos y químicos que alteran la columna de agua, de tal manera que se puedan desarrollar modelos que expliquen y permitan conocer sus efectos sobre el clima y los recursos.

En el proceso de caracterización costera también se realizan actividades de campo, para lo cual se determina previamente el área de estudio, y se definen estaciones (posiciones geográficas) en donde se ejecutarán las mediciones de los diversos parámetros especificados por las divisiones técnicas.

La caracterización tiene otro método de obtención de datos, ya que se ejecuta en un solo lugar, siendo observaciones en cada hora con diferentes parámetros y características.

En este proceso de caracterización intervienen las diferentes divisiones técnicas de la Dirección de Oceanografía, tal es el caso de Meteorología, Oceanografía Física, Oceanografía Química, Oceanografía Biológica, teniendo cada una de ellas participación y ejecución de actividades en determinados tipos de estudio.

Entre otros, los objetivos de estos estudios son:

- Meteorología: determinar las características climatológicas del área de estudio en función del régimen de viento, temperatura del aire, precipitaciones y presión atmosférica.
- Oceanografía Física: determinar el patrón de circulación superficial y subsuperficial, régimen de oleaje en función de su altura significativa, dirección y periodo, a fin de determinar la influencia del mismo en los procesos costeros en el área de estudio, levantamiento de información oceanográfica, para determinar los procesos costeros que afectan en función de: la altura y dirección de las olas de rompientes, dirección y magnitud de las corrientes litorales.

Una vez concluidas las actividades de campo, se realiza el procesamiento de los datos obtenidos de los equipos sensores, empleando Microsoft Excel y diferentes softwares de acuerdo a la

necesidad, por ejemplo, WRPLOT (para gráficos meteorológicos de viento), Matlab, ODV, ArcMap, entre otros.

Con las tablas de datos, y gráficos, los técnicos proceden al análisis y redacción de informes por división técnica, para luego compilar el informe de caracterización costera que es presentado a la Dirección General de Instituto, el mismo que servirá luego de fuente de consulta interna y punto de partida para posteriores estudios en el área, o circundante a ella.

En ambos procesos, las actividades de registro de datos se realizan empleando formularios impresos y de forma manual, conllevando esto deterioro y/o pérdida del medio de registro, errores en las anotaciones, entre otros problemas, tal como se explicó con anterioridad.

### **3.3. Definición de actores**

En los procesos de medición de corrientes marinas y caracterización costera se tiene como actores principales:

- Técnicos: personal que realiza las actividades de campo, posiciona y monitorea los equipos, ejecuta las mediciones, registra y analiza los datos, así como la elaboración de informes técnicos.
- Investigadores científicos: personal que posee más años de experiencia, el mismo que puede o no ejecutar actividades de campo, y que realiza además funciones de análisis e interpretación de datos, redactor y revisor de informes técnicos. Utilizarán estos datos para generar conocimiento.
- Director de Oceanografía: generará reportes que serán presentados en reuniones técnicas con la Dirección General del Instituto.
- Usuarios generales: personal interno de la misma Dirección u otras, que podrán visualizar los datos recolectados por la aplicación.

### **3.4. Definición de requerimientos**

En el presente proyecto se han definido los siguientes requerimientos, los cuales han sido consultados a la jefatura de la División de Oceanografía Física, previa autorización de la Dirección:

#### **Requerimientos no funcionales**

- Como requerimiento organizacional, la aplicación debe ajustarse a la política pública de uso de software libre en instituciones públicas del

Ecuador, no debe tener costos asociados en la adquisición de software de terceros para su funcionamiento, ni implicar cualquier otro desembolso económico para el instituto.

- La aplicación deberá contar con una interfaz de usuario intuitiva de fácil uso, que permita el registro de los datos de manera rápida y eficiente.
- La aplicación deberá ser capaz de receptar y procesar datos operando de manera adecuada en cada sesión de usuario.
- La aplicación debe ser capaz de funcionar en dispositivos con sistema operativo móvil Android versión 6.0 (Marshmallow) o superior, ya que, al ser una aplicación para uso interno de la institución, se efectuó una revisión de los dispositivos de los funcionarios de la División, y se encontró que en su totalidad poseían este sistema operativo, y así también, se evidenció que esta es la menor versión con la que contaban los equipos.
- La aplicación debe poseer el principio de portabilidad, que le permita su adaptación de manera efectiva y eficiente en diversos entornos de hardware, en este caso en dispositivos móviles Android exclusivamente.
- El producto final debe de contar con la facilidad para ser instalado o desinstalado de manera rápida y exitosa en un determinado entorno.

- La aplicación debe contar con seguridades que permitan el acceso a los formularios, y los permisos de acceso podrán ser modificados únicamente por un administrador central, garantizándose la protección de los datos.
- Como requerimiento de hardware: se debe contar con un servidor de base de datos y servidor web, este último con conectividad a Internet.
- Requerimiento de software: se debe contar con la plataforma ODK para gestión y administración de los formularios, PostgreSQL como sistema gestor de base de datos, PostGIS extensión GIS de PostgreSQL para la gestión de datos espaciales, Apache Tomcat como servidor web, Leaflet biblioteca de código abierto para implementar mapas interactivos en entorno web, jQuery para interacción con documentos HTML para la implementación del visor geográfico en formato web.
- Los dispositivos móviles para la recolección de datos deberán tener instalada la aplicación ODK Collect para acceder a los formularios.

### **Requerimientos funcionales**

- Autenticar usuarios a la aplicación.
- Modificar la clave de acceso de los usuarios a la aplicación.
- Crear nuevos usuarios y perfil de acceso a la aplicación.

- Actualizar datos de los usuarios de la aplicación.
- Eliminar usuarios de la aplicación.
- Crear nuevos formularios de recolección de datos.
- Modificar formularios de recolección de datos.
- Agregar campos multimedia a los formularios en caso de requerirse.
- Eliminar formularios de recolección de datos.
- Permitir la operación de la aplicación sin depender de conexión a Internet para la carga de datos.
- Crear visor geográfico de acuerdo a las necesidades de la Dirección.
- Presentar datos de acuerdo a ubicaciones y periodo de tiempo.
- Permitir búsqueda de datos en el visor geográfico.
- Crear reportes personalizados de los datos recolectados.

### **3.5. Alcance del proyecto**

El presente proyecto tiene como finalidad el diseño e implementación de una aplicación para recolección móvil de datos de mediciones de corrientes marinas y caracterización costera, y como alcance del mismo se tiene:

- La aplicación se ha desarrollado empleando la plataforma Open Data Kit de tipo Open Source, el servidor web, el sistema gestor de base de datos, la implementación de los formularios, visor geográfico y

reportes, no representan una inversión económica a tiempo presente para el instituto, ya que el equipo que se facilitó para la puesta en marcha de este proyecto corresponde al inventario de la institución.

- Es una solución completa que permite la creación de formularios, recolección y manejo de datos en el servidor.
- La aplicación ha sido desarrollada exclusivamente para dispositivos móviles con sistema operativo Android versión 6.0 o superior.
- La aplicación permite la recolección móvil de datos, reemplazando en las actividades de campo, a medios de registro análogos empleados actualmente, como son los formularios impresos.
- La aplicación implementa formularios que permiten atributos multimedia y geolocalización.
- La aplicación soporta la recolección de datos fuera de línea, y configurada correctamente permite enviar la información automáticamente al conectarse a una red WiFi.
- La aplicación permite el almacenamiento centralizado de los datos.
- El acceso y visualización de los datos se da en tiempo cuasi real, pudiendo ser monitoreados en el instituto por un operador.
- La aplicación no mantendrá conexión con la base de datos Oracle que posee el instituto, ni tampoco se interconectará con el sistema de información interno.

- La aplicación no se interconectará con ningún sistema o aplicación externa al instituto, ni se implementará ningún tipo de servicio web que provea de datos a terceros.
- No se implementará ninguna tabla en la base de datos centralizada del instituto.
- No se instalará ninguna herramienta, o desarrollará código personalizado que permita implementar un proceso ETL (del inglés, Extract, Transform and Load), para mover datos entre las bases de datos Postgre empleada por la aplicación, y Oracle usada por el sistema de información.
- La aplicación no pretende sustituir a algún módulo existente actualmente en el sistema de información, y su propuesta e implementación obedece únicamente a cumplir con los objetivos planteados con anterioridad.

## **CAPÍTULO 4**

### **ANÁLISIS, DISEÑO E IMPLEMENTACIÓN**

#### **4.1. Análisis de requerimientos**

El presente proyecto será diseñado e implementado empleando la plataforma Open Data Kit (ODK), que es un conjunto de herramientas basado en código abierto que ayuda en la creación formularios de encuestas, recopilación de datos en un dispositivo móvil y envío del formulario a un servidor agregando los datos recopilados, para su posterior extracción en formatos útiles.

ODK Aggregate provee de un servidor listo para despliegue, el cual puede realizar la carga, almacenamiento y transferencia de datos a servidores locales o a la nube; es una aplicación de código abierto desarrollada en entorno Java, la cual almacena, analiza y muestra datos de encuestas XForm recopilados utilizando ODK Collect u otras aplicaciones compatibles con OpenRosa. Soporta variados tipos de

datos y está diseñado para funcionar bien en cualquier entorno de alojamiento.

Esta herramienta proporcionará entre otras funcionalidades:

- Suministrar los formularios a la aplicación instalada en los dispositivos móviles.
- Cargar y gestionar los datos recolectados por dispositivos móviles.
- Visualizar los datos recolectados a través de gráficos simples y mapas.
- Exportar los datos recolectados a archivos de formatos CSV, KML y JSON.
- Publicar los datos recolectados a aplicaciones externas como Google FusionTables y Google Spreadsheet.

Teniendo en cuenta estas consideraciones, se procedió a instalar de manera local en un equipo servidor correspondiente al inventario tecnológico del instituto, el componente ODK Aggregate. El inicio del proceso de instalación se muestra en la Figura 4.1.

```

Aplicaciones Lugares Terminal
root@collect/opt/tomcat/webapps

Archivo Editar Ver Buscar Terminal Ayuda
total 4
-rw-r--r-- 1 root root 1388 mar  6 10:32 anaconda-ks.cfg
[root@collect ~]# mkdir odk
[root@collect ~]# cd /opt/tomcat/webapps/
[root@collect webapps]# ll
total 276400
drwxr-x--- 14 1001 1001   4096 nov 11  2017 docs
drwxr-x---  6 1001 1001    83 nov 11  2017 examples
drwxr-x---  5 1001 1001    67 dic 14  2017 geodk
-rw-r--r--  1 root root 89616750 nov 21  2017 geodk.war
drwxrwxrwx. 16 1001 1001   4096 nov 11  2017 geonetwork
drwxr-xr-x.  6 1001 1001    181 nov 11  2017 geoserver
drwxr-xr-x.  6 1001 1001    101 dic 14  2017 geoserver2
drwxr-x---  5 1001 1001    87 nov 11  2017 host-manager
drwxr-x---  5 1001 1001    163 nov 11  2017 manager
drwxr-x---  9 1001 1001   4096 mar 19  2019 ODKAggregate
-rwxr-xr-x.  1 root root 141778048 mar 19  2019 ODKAggregate1.4.13Linux-x64-installer.run
-rw-r--r--  1 root root 51617334 mar 19  2019 ODKAggregate.war
drwxr-x---  3 1001 1001   4096 nov 11  2017 ROOT
[root@collect webapps]# ./ODKAggregate1.4.13Linux-x64-installer.run
.....
Welcome to the ODK Aggregate Setup Wizard.
.....
Please read the following License Agreement. You must accept the terms of this
agreement before continuing with the configuration wizard.

Press [Enter] to continue :
Copyright (C) 2009 Google Inc.
Copyright (C) 2010-2016 University of Washington

Licensed under the Apache License, Version 2.0 (the "License"); you may
not use this software except in compliance with the License. You may
obtain a copy of the License at LICENSE-2.0
<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

This software includes the following jar files (libraries):
http://www.oracle.com/technetwork/java/javaf11-139815.html   /standard
extension/
activation-1.1.jar
http://aopalliance.sourceforge.net                          /Public domain/
aopalliance-1.0.jar
https://cloud.google.com/appengine/docs/java/release-notes /Apache 2.0/
appengine-api-1.0-sdk-1.9.48.jar
Press [Enter] to continue :

```

**Figura 4.1: Inicio de instalación de ODK Aggregate**

Continuando el proceso, se solicita elegir una plataforma, teniendo opciones entre Google App Engine (servicio basado en la nube), o instalación local basada en el servidor web Apache Tomcat 8, y servidores de base de datos como MySQL, PostgreSQL y SQL, como se muestran en la Figura 4.2 y Figura 4.3. En el caso de este proyecto, se trabajará con PostgreSQL por familiaridad con la administración del motor de base de datos y posterior uso de la librería PostGIS en la arquitectura de la aplicación. En la Figura 4.4 se presentan más detalles de configuración de la base datos PostgreSQL.

En este paso es importante indicar, que ODK Aggregate actúa como una caja negra en conjunto con el servidor de base de datos, ya que automáticamente genera la estructura de las tablas que se requieren para el núcleo de los formularios, así como para el manejo de los archivos multimedia asociados a los mismos.

```

Aplicaciones Lugares Terminal
root@collect/opt/tomcat/webapps

Archivo Editar Ver Buscar Terminal Ayuda
https://github.com/FasterXML/woodstox /Apache 2.0/
woodstox-core-5.0.3.jar
https://github.com/codehaus/woodstox /Apache 2.0/
woodstox-core-asl-4.4.1.jar

Press [Enter] to continue :
Do you accept this license? [y/n]: y

-----
Select an output parent directory

This "installer" does not actually install anything. Instead, it guides you
through configuring ODK Aggregate. As the last step of the installer, it will
either launch the upload tool for uploading to Google's AppEngine cloud services
or open a README.html containing the final installation instructions for MySQL
or PostgreSQL deployments.

Please select the parent directory under which an 'ODK Aggregate' directory will
be created to contain the configured software.

[]: /root/odk/

-----
Choose Platform

Choose the data storage and execution environment on which you would like ODK Aggregate to run. If you choose MySQL, PostgreSQL or
tall that database server and an Apache Tomcat 8 webserver.

NOTE: If you want to use the Microsoft Azure SQL Server, you must configure Tomcat 8 to use Java 8.

[1] Google App Engine: Uses Google's cloud-based data storage and webserver services.
[2] MySQL: Uses a MySQL database and an Apache Tomcat 8 webserver that you install.
[3] PostgreSQL: Uses a PostgreSQL database and an Apache Tomcat 8 webserver that you install.
[4] SQL Server: Uses a Microsoft SQL Server or Azure SQL Server database and an Apache Tomcat 8 webserver that you install.
Please choose an option [1] : 3

-----
Pre-Installation Requirements - Apache Tomcat

Before continuing, you should download and install an Apache Tomcat 8 webserver
within which you will run ODK Aggregate (instructions on installing your chosen
database will be presented later).

Apache Tomcat 8 can be downloaded from the link below.

If you modify any of the Apache Tomcat defaults, please make note of them.

NOTE: If you want to use the Microsoft Azure SQL Server, you must configure
Tomcat 8 to use Java 8.

http://tomcat.apache.org/download-80.cgi [Y/n]: 

```

**Figura 4.2: ODK Aggregate, selección de plataforma**

```

Aplicaciones  Lugares  Terminal
root@collect:/opt/tomcat/webapps

Archivo  Editar  Ver  Buscar  Terminal  Ayuda
[4] SQL Server: Uses a Microsoft SQL Server or Azure SQL Server database and an Apache Tomcat 8 webserver that you install.
Please choose an option [1] : 3

-----
Pre-Installation Requirements - Apache Tomcat

Before continuing, you should download and install an Apache Tomcat 8 webserver
within which you will run ODK Aggregate (instructions on installing your chosen
database will be presented later).

Apache Tomcat 8 can be downloaded from the link below.

If you modify any of the Apache Tomcat defaults, please make note of them.

NOTE: If you want to use the Microsoft Azure SQL Server, you must configure
Tomcat 8 to use Java 8.

http://tomcat.apache.org/download-80.cgi [Y/n]: y

-----
Apache Tomcat SSL configuration

SSL certificates allow a client to verify the identity of a webserver and to
establish secured (encrypted) communications with that webserver (https). SSL
certificates are typically purchased from Verisign or another issuing authority;
that process can take weeks to complete.

If you do have an SSL certificate, please refer to the Apache Tomcat
documentation for how to install the certificate on your system. Within the
Tomcat 8.0 documentation, refer to the User Guide section 12: SSL.

http://tomcat.apache.org/tomcat-8.0-doc/ssl-howto.html [Y/n]: y

-----
Apache Tomcat SSL configuration

Do you have an SSL certificate?

[1] No, I do not have an SSL certificate: - User logins ARE secure.
- Submitting and/or viewing form data MAY NOT BE secure.
- Changing passwords MAY NOT BE secure.
Security breaches are particularly likely over unsecured WIFI hotspots.

[2] Yes, I have an SSL certificate: - User logins ARE secure.
- Submitting and/or viewing form data IS secure.
- Changing passwords IS secure.

Please choose an option [1] : 1

```

**Figura 4.3: ODK Aggregate, configuración Apache Tomcat**

```

Aplicaciones Lugares Terminal
root@collect/opt/tomcat/webapps

Archivo Editar Ver Buscar Terminal Ayuda
We have developed and tested against PostgreSQL 9.0

Download and install PostgreSQL from the link below.

Make note of any non-default values you've entered during the installation
process.

http://www.postgresql.org/ [Y/n]: y

-----
Database server settings

Specify the host and port number used to communicate with the database server.

If the database server is on the same host as the webserver, enter '127.0.0.1'

Database port number: [5432]: 5432

Database server hostname: [127.0.0.1]: 127.0.0.1

-----
ODK Aggregate database authentication settings

The database server must be configured with an account (username and password)
that ODK Aggregate will employ for accessing the database server. This is not
an account that you will give to anyone; it is only for use by the ODK Aggregate
webserver. Username and password should be alphanumeric strings beginning with a
letter (no spaces or punctuation please!).

The installer will generate a script that can be run on the database server to
establish the required configuration.

Database username: [odk_user]: odk_user

Database password: :
Retype password: :

-----
ODK Aggregate database datastore settings

The database server must be configured with a workspace identified by a database
name, and, if using Postgresql, a schema name. This workspace is where ODK
Aggregate will store uploaded forms, submissions and other information. The
database name (and schema name for Postgresql) can be any alphanumeric strings
beginning with a letter; underscores (_) are OK (no spaces or punctuation
please!).

The installer will generate a script that can be run on the database server to
create the workspace.

Database Name: [odk_prod]: odk_prod

```

**Figura 4.4: ODK Aggregate, configuración PostgreSQL**

Al concluir con el proceso, se comprueba el correcto funcionamiento del servicio, accediendo a la interfaz del ODK Aggregate, mediante un navegador web digitando la dirección: <http://10.130.22.110:8080/ODKAggregate/Aggregate.html>, como lo muestra la Figura 4.5.



Figura 4.5: Interfaz web de ODK Aggregate

### Autenticar usuarios a la aplicación

El acceso a los formularios por parte del personal que recolectará los datos será restringido, y para esto se crearán usuarios en el servidor ODK Aggregate, con lo cual se garantiza que mientras no exista una autenticación exitosa en la aplicación, no se podrá descargar o enviar formulario alguno al servidor.

ODK Aggregate administra las cuentas y permisos de los usuarios, así como también almacena las definiciones de los formularios, y permite que los clientes de recolección de datos como ODK Collect se conecten a ellos para descargarlos y cargarlos en el dispositivo móvil. Dentro del módulo de administración del sitio, se encuentra la configuración de acceso al sitio, tal como se presenta en la Figura 4.6.

Users access the site either

- anonymously (i.e., as the `anonymousUser`), or
- via an OAuth 2.0 token (requires an Email account (e.g., `user@gmail.com`) and an OAuth 2.0 token issued by Google with access to `userinfo.email`), or
- via an ODK account, with a username and password that a site administrator has configured for them.

Capabilities are as follows:

- Data Collector - able to download forms to ODK Collect and submit data from ODK Collect to ODK Aggregate. Only ODK accounts and the `anonymousUser` can be granted Data Collector rights. The `anonymousUser` must be granted Data Collector rights to accept submissions from unidentified sources (e.g., from ODK Collect 1.1.5 and earlier, or from ODK Collect 1.1.7 and later if not authenticating).
- Data Viewer - able to log onto the ODK Aggregate website, filter and view submissions, and generate `csv` or `kml` files for download.
- Form Manager - all the capabilities of a Data Viewer plus the abilities to upload a form definition, delete a form and its data, and upload submissions manually through the ODK Aggregate website.
- Synchronize Tables - able to download, upload and alter data records within ODK Tables as restricted by table-access privileges granted to the user.
- Tables Super-user - able to administer row-level and table-access privileges.
- Administer Tables - able to administer tables and row-level and table-access privileges.
- Site Administrator - all the capabilities of a Form Manager plus the ability to add users, set passwords, and grant these capabilities to other users.

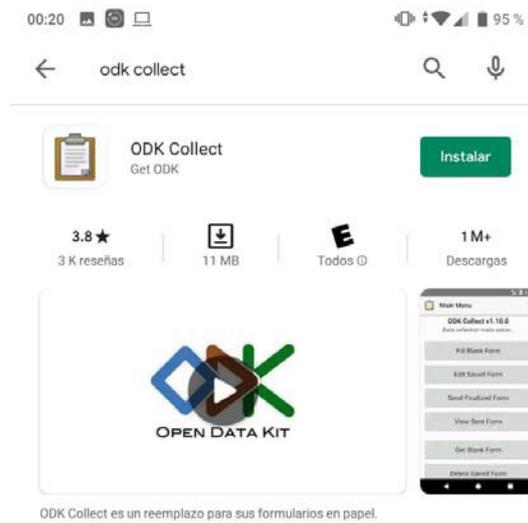
Configure Users and Capabilities using .csv file

[Upload Users and Capabilities .csv](#)

[Download current UsersAndCapabilities.csv](#)

Figura 4.6: ODK Aggregate, configuración de acceso al sitio

El proceso de creación de usuarios se detallará más adelante, pero para ingresar al servidor y proceder con la descarga de los formularios, el técnico deberá tener instalado en su dispositivo móvil la aplicación ODK Collect, la cual se obtiene desde Google Play Store (en este caso por ser empleado para equipos basados en Android exclusivamente), como lo muestra la Figura 4.7.



**Figura 4.7: Descarga de ODK Collect desde Google Play Store**

Instalado el ODK Collect en el dispositivo móvil, se procede a configurar la dirección URL del servidor ODK Aggregate, para realizar la conexión y generar la solicitud de los formularios al equipo, como lo muestran la Figura 4.8 y Figura 4.9.

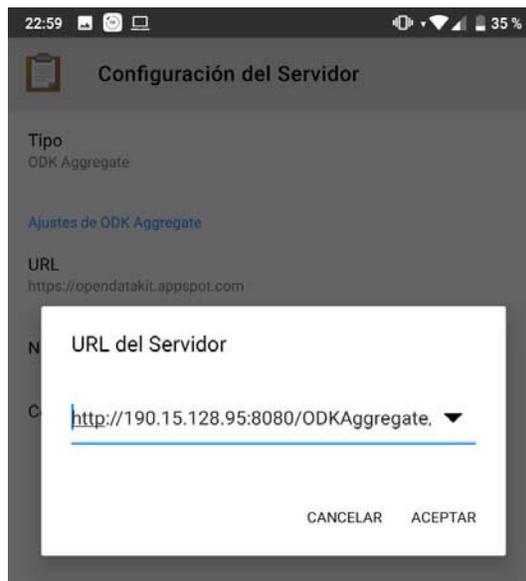


Figura 4.8: Configuración de URL del servidor en ODK Collect



Figura 4.9: URL del servidor ODK Aggregate configurada

Realizada la configuración de la URL del servidor, se ingresa el nombre de usuario y contraseña para establecer la conexión, y con esto disponer

de los formularios y revisar sus contenidos en la aplicación, como lo presentan la Figura 4.10 y Figura 4.11.

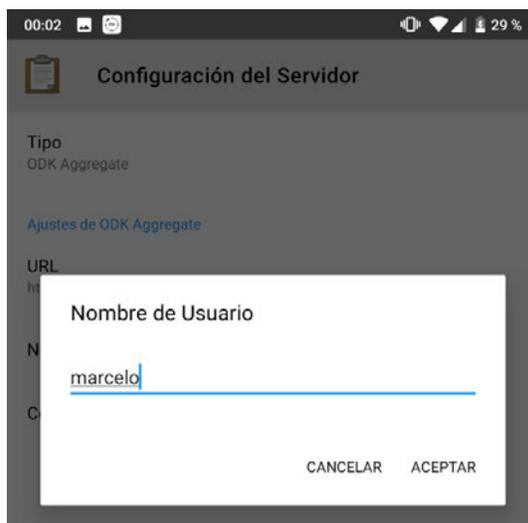


Figura 4.10: Ingreso de usuario en ODK Collect

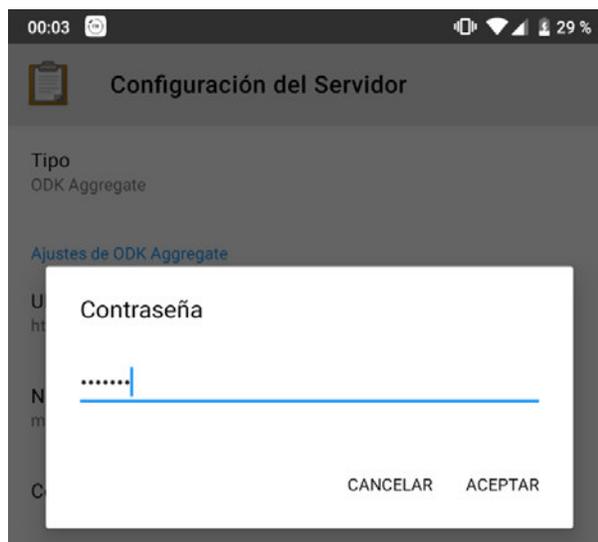


Figura 4.11: Ingreso de contraseña en ODK Collect

## Modificar la clave de acceso de los usuarios a la aplicación

En el módulo de administración del sitio, dentro del ODK Aggregate, se selecciona el usuario al cual se requiere modificar la contraseña de ingreso, y se presiona el botón Change Password, luego de lo cual se presenta una ventana flotante en donde se ingresa la nueva contraseña, como se muestra en la Figura 4.12.

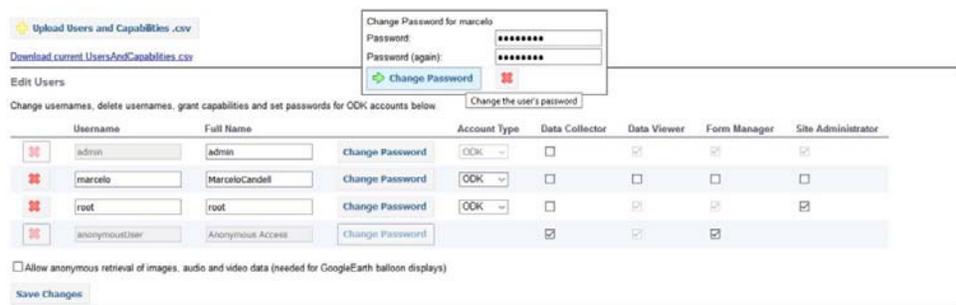


Figura 4.12: Cambio de clave de usuario en ODK Aggregate

## Crear nuevos usuarios y perfil de acceso a la aplicación

Dentro del módulo de administración del sitio, en la configuración de acceso, se detalla cómo los usuarios acceden al sitio para descarga de formularios:

- De manera anónima, empleando el usuario *anónimo*, la cual no es aconsejada por motivos de seguridad.

- A través de un token Oauth 2.0, para lo que se requiere una cuenta de correo electrónico, por ejemplo: usuario@gmail.com, y un token Oauth 2.0 emitido por Google con acceso a la cuenta de correo del usuario en dicha plataforma. Las credenciales de la API de Google se utilizan al publicar en los servicios de Google.
- A través de una cuenta ODK, con un nombre de usuario y contraseña que un administrador del sitio ha configurado para ellos. Este será el método empleado en este proyecto para la gestión de acceso a la aplicación.

Por otro lado, en este mismo apartado se detallan los perfiles de acceso a la aplicación, teniendo los siguientes:

- Recopilador de datos: puede descargar formularios en ODK Collect y enviar datos de ODK Collect a ODK Aggregate. Solo las cuentas ODK y el usuario anónimo pueden tener derechos de recopilador de datos.
- El usuario anónimo debe tener derechos de recopilador de datos para aceptar envíos de fuentes no identificadas (por ejemplo, de ODK Collect 1.1.5 y anteriores, o de ODK Collect 1.1.7 y posteriores si no se autentica).
- Visor de datos: puede iniciar sesión en el sitio web de ODK Aggregate, filtrar y ver envíos y generar archivos CSV o KML para descargar.

- Administrador de formularios: todas las capacidades de un visor de datos más las capacidades para cargar una definición de formulario, eliminar un formulario con sus datos y cargar envíos manualmente a través del sitio web de ODK Aggregate.
- Sincronizar tablas: puede descargar, cargar y modificar registros de datos dentro de las tablas ODK según lo restringido por los privilegios de acceso a la tabla otorgados al usuario.
- Superusuario de tablas: capaz de administrar privilegios de nivel de fila y acceso a la tabla.
- Administrar tablas: capaz de administrar tablas y privilegios de nivel de fila y acceso a la tabla.
- Administrador del sitio: todas las capacidades de un Administrador de formularios más la capacidad de agregar usuarios, establecer contraseñas y otorgar estas capacidades a otros usuarios.

ODK Aggregate permite manejar esta administración de usuarios mediante la carga de un archivo CSV, en donde se detallan datos y las capacidades de gestión que tendrá cada usuario ante el servidor.

Como se indicó anteriormente, para la gestión de acceso a la aplicación se emplearán cuentas ODK únicamente, y el formulario para ingreso del nombre de usuario es similar al presentado en la Figura 4.13.

Add Users

Enter the usernames (for ODK accounts) or e-mail addresses (for Google Gmail accounts) to add to the list of users in the above table. Interactive logins with Google accounts are no longer supported.

usrPrueba

Add

**Figura 4.13: Añadir nuevo usuario en ODK Aggregate**

Añadido el usuario, pasa automáticamente a la sección de edición usuarios de cuentas ODK, en donde se puede cambiar o eliminar nombres de usuario, otorgar capacidades de gestión y modificar sus contraseñas, como se presenta en la Figura 4.14. Para este ejemplo se ha ingresado el nombre completo del usuario usrPrueba, y se ha otorgado las capacidades de colector de datos, visualizador de datos y administrador de formularios.

En esa sección se indica además que se puede permitir la recuperación anónima de imágenes, datos de audio y video, necesarios para pantallas de GoogleEarth, pero dicha opción no ha sido seleccionada.

Edit Users

Change usernames, delete usernames, grant capabilities and set passwords for ODK accounts below.

Username	Full Name	Account Type	Data Collector	Data Viewer	Form Manager	Site Administrator
admin	admin	ODK	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
marcelo	MarceloCandell	ODK	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
root	root	ODK	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
usrPrueba	UsuarioPrueba	ODK	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
anonymousUser	Anonymous Access		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Allow anonymous retrieval of images, audio and video data (needed for GoogleEarth balloon displays)

Save Changes

**Figura 4.14: Añadir datos de nuevo usuario en ODK Aggregate**

## Actualizar datos de los usuarios de la aplicación

Anteriormente se indicó que las cuentas ODK pasaban a la sección de edición de usuarios, y siendo el formulario sencillo, se puede actualizar nombre de usuario, nombre completo, el tipo de cuenta (en este caso no se realizará, ya que se trabajará con la de tipo ODK únicamente), y las capacidades asignadas al usuario en cuestión, siendo todo ejecutado al presionar el botón **Save Changes**, como lo muestra la Figura 4.15, en la cual se actualizaron datos del usuario usrPrueba1.

Edit Users

Change usernames, delete usernames, grant capabilities and set passwords for ODK accounts below.

	Username	Full Name		Account Type	Data Collector	Data Viewer	Form Manager	Site Administrator
	admin	admin	Change Password	ODK	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	marcelo	MarceloCandell	Change Password	ODK	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	root	root	Change Password	ODK	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	usrPrueba1	UsuarioPrueba1	Change Password	ODK	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	anonymousUser	Anonymous Access	Change Password		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Allow anonymous retrieval of images, audio and video data (needed for GoogleEarth balloon displays)

Save Changes

Figura 4.15: Actualizar datos de usuario en ODK Aggregate

## Eliminar usuarios de la aplicación

En la sección de edición de usuarios, se presenta un ícono con forma de X y color rojo, el cual permite la eliminación del usuario de la fila en cuestión. Una vez presionado el botón, se presenta un cuadro de diálogo con un mensaje de confirmación, donde se pregunta si realmente se desea eliminar permanentemente al usuario del servidor ODK Aggregate,

ejecutándose la acción al presionar el botón **Delete User**, como lo presenta la Figura 4.16.

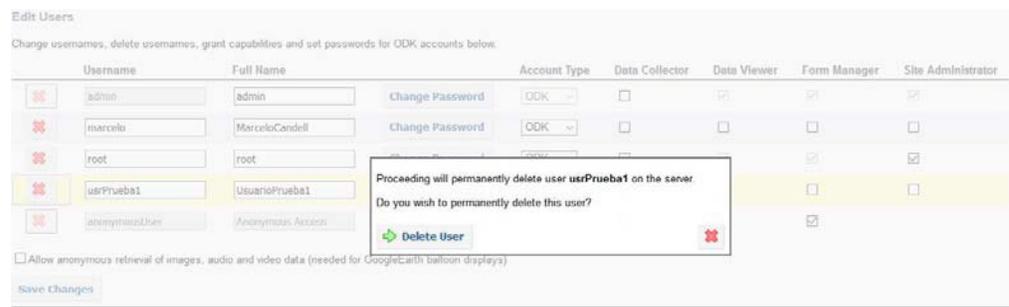


Figura 4.16: Eliminar usuario en ODK Aggregate

### Crear nuevos formularios de recolección de datos

El diseño y desarrollo de formularios se describirán con mayor detalle en las secciones 4.2 y 4.4 respectivamente, y en este apartado únicamente se indicará que en el servidor ODK Aggregate se encuentra la opción **Form Management**, en donde se presenta el listado de todos los formularios cargados y ciertas acciones que se pueden realizar sobre ellos, además de existir el botón **Add New Form**, que permite la carga de un nuevo formulario a la plataforma, como lo muestra la Figura 4.17.



Figura 4.17: Carga de formulario en ODK Aggregate

### Modificar formularios de recolección de datos

La modificación de formularios se realiza cargando una nueva versión del formulario diseñado, lo cual se detalla en la sección 4.2.

### Agregar campos multimedia a los formularios en caso de requerirse

Se realiza cargando una nueva versión del formulario diseñado, aumentando el o los campos multimedia requeridos, lo cual se detalla en la sección 4.2.

### Eliminar formularios de recolección de datos

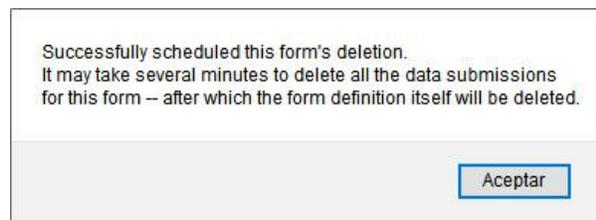
En el servidor ODK Aggregate, dentro de la opción **Form Management**, se presenta un listado de los formularios cargados a la plataforma, y entre otras opciones, se muestra la de eliminar el formulario en cuestión, mediante el botón **Delete**, y al presionarlo aparece un cuadro de diálogo,

en el cual se pide confirmar la acción, eliminando la data recolectada y el formulario conjuntamente, como lo muestra la Figura 4.18.



**Figura 4.18: Eliminar formulario en ODK Aggregate**

Al presionar el botón, se presenta un cuadro de diálogo donde se indica “que se ha programado con éxito la eliminación del formulario, pudiendo tomar varios minutos eliminar todos los envíos de datos, para este formulario, después de lo cual se eliminará la definición del formulario.”, como se muestra en la Figura 4.19.



**Figura 4.19: Confirmación de eliminación de formulario**

### **Permitir la operación de la aplicación sin depender de conexión a Internet para la carga de datos**

La configuración de esta opción se realiza desde el dispositivo móvil, en la aplicación ODK Collect. Se ingresa al menú **Cambiar la configuración**, luego a **Manejo de formularios**, seleccionando **Envío Automático**, encontrando en este submenú, las opciones: Desactivado, Solo con WiFi, Solo con Datos Móviles, y WiFi y Datos Móviles. En el presente proyecto se trabajará con la opción **Solo con WiFi**, ya que será más práctica y económica para el personal que labora en campo, el cual no tendrá siempre acceso al paquete de datos móviles, ni los consumirá de su plan de telefonía contratado, de tal manera que el envío de los formularios, se efectuará una vez que regresen a la estación base. Todos estos pasos se muestran en las Figuras 4.20, 4.21, 4.22 y 4.23.



**Figura 4.20: Configurar envío de formulario empleando WiFi (1)**

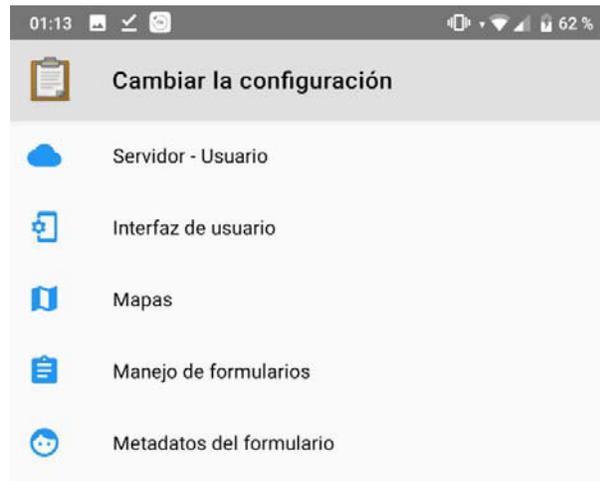


Figura 4.21: Configurar envío de formulario empleando WiFi (2)

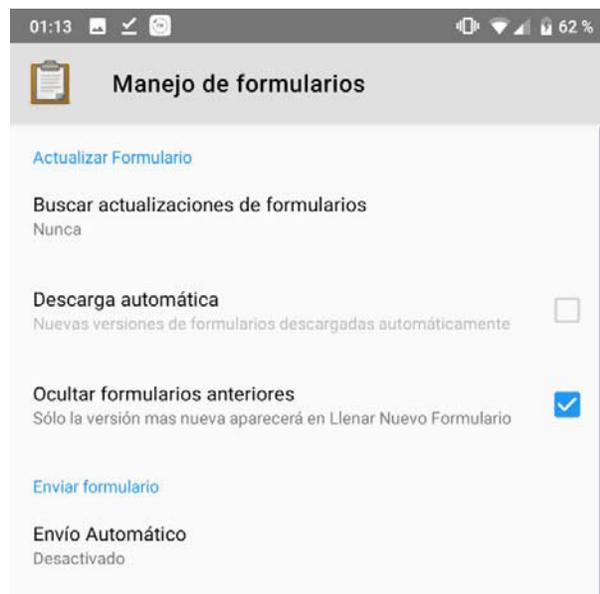


Figura 4.22: Configurar envío de formulario empleando WiFi (3)

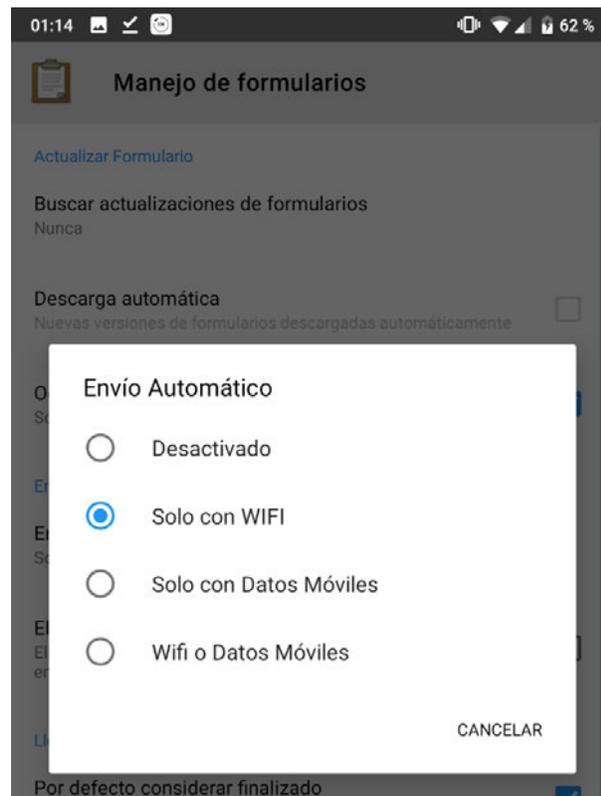


Figura 4.23: Configurar envío de formulario empleando WiFi (4)

### **Crear visor geográfico de acuerdo a las necesidades de la Dirección**

La creación del visor geográfico en entorno web donde se presenten los datos recolectados mediante la aplicación móvil, se detallará en la sección 4.5.

### **Presentar datos de acuerdo a ubicaciones y periodo de tiempo**

La creación del visor geográfico en entorno web en donde se presenten los datos recolectados mediante la aplicación móvil, se detallará en la

sección 4.5, pero se indica que los datos pueden ser presentados seleccionando la ubicación geográfica donde se realizó la recolección, y el periodo será presentado en una línea de tiempo dentro del visor, como se muestra en la Figuras 4.24.

### Visor de Veleteo

#### Observaciones en el mapa

(Click en el nombre la observacion para Zoom):

<b>P2</b> [Longitud: -80.05580641 Latitud: -3.25516024] 2020/03/05 05:12:00 <b>azul</b>
<b>P2</b> [Longitud: -80.05729627 Latitud: -3.25312229] 2020/03/05 05:23:00 <b>amarillo</b>
<b>P3</b> [Longitud: -80.05867375 Latitud: -3.25019477] 2020/03/05 05:35:00 <b>negro</b>
<b>P4</b> [Longitud: -80.06010907 Latitud: -3.24828885] 2020/03/05 05:40:00 <b>verde</b>

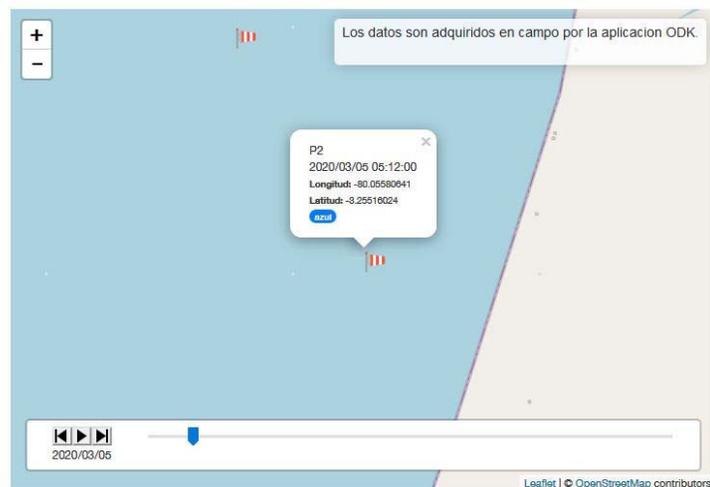


Figura 4.24: Presentación de datos por ubicación y tiempo

### Permitir búsqueda de datos en el visor geográfico

Dentro del visor geográfico en entorno web, se adjuntó una tabla para presentación de los datos recolectados y almacenados en la base de datos, la cual se implementó empleando DataTables, un complemento para jQuery, la librería multiplataforma de Javascript, la cual es una herramienta altamente flexible, construida sobre las bases de la mejora

progresiva, que agrega estas características avanzadas a cualquier tabla implementada en HTML.

Este complemento o plug-in, implementa automáticamente todas las funciones heredadas de jQuery, contando entre otras, con las siguientes características:

- Búsqueda instantánea filtrando resultados por ingreso de texto.
- Mostrar cantidad de registros.
- Paginación de resultados: navegación anterior, siguiente y de página.
- Ordenar datos por múltiples columnas a la vez.
- Utilizar casi cualquier fuente de datos: DOM, Javascript, Ajax y procesamiento del lado del servidor.
- Facilidad de personalizar temas-estilos: mediante creador de temas DataTables, uso de Bootstrap 3.x – 4.x y Semantic UI.
- Amplia variedad de extensiones: editor, botones, diseño responsivo y más.
- Amigable para presentación en dispositivos móviles, ya que basados en el diseño responsivo, las tablas se adaptan al tamaño de la ventana gráfica.

En la implementación de la tabla se descargó la extensión **Botones de exportación HTML5** (HTML5 export buttons), que permiten realizar las siguientes acciones:

- Copy: copia el contenido de la tabla de datos al portapapeles, considerando el ordenamiento actual se tenga en la presentación.
- Excel: exporta los datos a un archivo de formato de hoja de cálculo Excel (XLS).
- CSV: exporta los datos a un archivo de formato separado por comas (CSV).
- PDF: exporta los datos a un archivo de formato de documento portátil (PDF).

Con las características y funcionalidades detalladas anteriormente, la tabla de datos constará con la búsqueda instantánea, ordenamiento entre columnas, y botones HTML5, y la búsqueda de los datos se puede realizar ingresando alguna palabra clave en el cuadro de texto y empleando jQuery realizará el filtrado de los datos previamente cargados, como lo muestra la Figura 4.25.

TODOS LOS DATOS

Copy Excel CSV PDF - VELETEO

Search: amarillo

HORA MUESTREO	TIPO	SIEMBRA	CARACTERISTICA	COLOR	FECHAHORA_LANZADA
2020/03/05 08:13:00	Recogida	Siembra #5	superficial	amarillo	2020-03-05T18:13:00Z
2020/03/05 10:50:00	Recogida	Siembra #2	superficial	amarillo	2020-03-05T20:50:00Z
2020/03/05 09:55:00	Punto #5	Siembra #2	superficial	amarillo	2020-03-05T19:55:00Z
2020/03/05 10:22:00	Punto #4	Siembra #2	superficial	amarillo	2020-03-05T20:22:00Z
2020/03/05 05:23:00	Punto #2	Siembra #1	superficial	amarillo	2020-03-05T15:23:00Z
2020/03/05 06:20:00	Punto #2	Siembra #2	superficial	amarillo	2020-03-05T16:20:00Z
2020/03/05 07:06:00	Punto #2	Siembra #3	superficial	amarillo	2020-03-05T17:06:00Z
2020/03/05 07:44:00	Punto #2	Siembra #4	superficial	amarillo	2020-03-05T17:44:00Z

Showing 1 to 8 of 8 entries (filtered from 36 total entries)

Previous 1 Next

**Figura 4.25: Búsqueda de datos en visor geográfico**

### Crear reportes personalizados de los datos recolectados

El plug-in DataTables con su extensión Botones de exportación HTML5, permite exportar los datos cargados y presentados en la tabla, a un archivo en formato PDF como lo muestra la Figura 4.26. La personalización de los reportes, dependerá exclusivamente de los campos que en la fase de desarrollo se agreguen a la tabla, pudiendo elegir cuál de ellos serán presentados.



### Reporte de veleteo



HORA MUESTREO	TIPO	SIEMBRA	CARACTERISTICA	COLOR	FECHA HORA LANZADA
2020/03/05 05:23:00	Punto #2	Siembra #1	superficial	amarillo	2020-03-05T15:23:00Z
2020/03/05 06:20:00	Punto #2	Siembra #2	superficial	amarillo	2020-03-05T16:20:00Z
2020/03/05 07:06:00	Punto #2	Siembra #3	superficial	amarillo	2020-03-05T17:06:00Z
2020/03/05 07:44:00	Punto #2	Siembra #4	superficial	amarillo	2020-03-05T17:44:00Z
2020/03/05 08:13:00	Recogida	Siembra #5	superficial	amarillo	2020-03-05T18:13:00Z
2020/03/05 09:55:00	Punto #5	Siembra #2	superficial	amarillo	2020-03-05T19:55:00Z
2020/03/05 10:22:00	Punto #4	Siembra #2	superficial	amarillo	2020-03-05T20:22:00Z
2020/03/05 10:50:00	Recogida	Siembra #2	superficial	amarillo	2020-03-05T20:50:00Z

**Figura 4.26: Exportación de datos a archivo PDF**

## 4.2. Diseño de formularios

Para el diseño de formularios, la plataforma ODK requiere la definición de los mismos, más no un objeto como cualquier desarrollo en algún IDE, y se necesita que esta definición sea cargada en el servidor ODK Aggregate en formato XML.

Remitiéndose a la documentación oficial de formularios ODK en internet [30], se detalla que las herramientas de construcción de formularios, son XLSForm (modo online u offline) y ODKBuild (modo online u offline) la

cual es una aplicación web HTML5 que funciona muy bien para diseñar formularios simples.

Para el presente proyecto se empleará XLSForm, y tal como lo describe su sitio web oficial [31], XLSForm es un formulario estándar que ayuda con la simplificación en el proceso de creación de formularios en Excel, empleando para esto un formato legible por humanos. XLSForms provee un estándar práctico para compartir y colaborar en formularios de autor, ya que son sencillos de comenzar, y permiten la creación de formularios complejos por parte de alguien familiarizado con la sintaxis que emplea esta herramienta.

El formulario XLSForm luego se convierte en uno del tipo ODK XForm, que es un popular estándar de formulario abierto, que permite crear un formulario con funcionalidades complejas en varias plataformas web y móviles de recolección de datos [31]. ODK XForms es utilizado por herramientas en el ecosistema ODK. Es un subconjunto de la especificación W3C XForms 1.0 mucho más grande y también contiene ciertas características agregadas que no se hallan en la especificación W3C XForms.

Para realizar el diseño de los formularios de recolección de datos, XLSForm presenta un formato básico, en el que cada libro de Excel generalmente contiene dos hojas de trabajo denominadas **encuesta** y **opciones**, pudiendo existir una tercera hoja opcional llamada **configuración**, en donde se puede agregar especificaciones adicionales al formulario. Este formato está disponible para la descarga en el sitio web de XLSForm, y es el que se utilizó como documento de referencia para el diseño de formularios del proyecto.

Como se indicó de manera previa, estos libros de Excel contienen las siguientes hojas de trabajo:

**Hoja de trabajo de la encuesta:** es la hoja de trabajo que le da al formulario su estructura general, y comprende la mayor parte del contenido del formulario. Contiene el listado completo de las preguntas e información acerca de cómo deberían presentarse en el formulario. Por lo general cada fila representa una pregunta; sin embargo, existen otras características que se pueden agregar al formulario para mejorar la experiencia del usuario.

**Hoja de trabajo de opciones:** en esta hoja de trabajo se especifican las opciones de respuesta para preguntas de opción múltiple, en donde cada

fila representa una opción de respuesta, y las opciones de respuesta que tienen el mismo nombre de lista son consideradas parte de un conjunto relacionado de opciones, apareciendo juntas para una pregunta. Esto también permite reutilizar un conjunto de opciones para múltiples preguntas (por ejemplo, preguntas sí / no).

Estas hojas de trabajo contienen un conjunto de columnas que obligatoriamente deben estar presentes en el formulario para que este funcione. Así también, cada hoja de trabajo tiene un conjunto de columnas opcionales que permitirán un mayor control sobre el comportamiento que debe tener cada entrada en el formulario, pero es importante decir, que no es esencial tenerlas. Lo que si debe ocurrir es que cada entrada debe tener valores para cada una de las columnas obligatorias, pudiendo las columnas opcionales dejarse en blanco.

En la hoja de trabajo de la encuesta se observan 3 columnas obligatorias:

- La columna de tipo: especifica el tipo de entrada que se está agregando, las mismas que pueden ser del tipo entero, decimal, rango, texto libre, selección única y múltiple, selección única y múltiple desde un archivo (ejemplo: archivos XML y CSV), ranking, notas, geopunto (única coordenada con GPS), geotrazo (línea, o dos o más coordenadas con GPS), geoshape (polígono de múltiples

coordenadas GPS), fecha, hora, fecha y hora, imagen (tomar una foto o cargar un archivo de imagen), audio (tomar una grabación o subir un archivo de audio), video (tomar una grabación o subir un archivo de video), siendo estos campos de tipo multimedia, así como la entrada genérica de archivos (txt, pdf, xls, xlsx, doc, docx, rtf, zip), entre otros.

- La columna de nombre: especifica el nombre único de variable para esa entrada, sin que existan dos entradas que puedan tener el mismo nombre. Estos nombres tienen que empezar con una letra, o un guión bajo, pudiendo solo contener letras, dígitos, guiones, guiones bajos y puntos, siendo los nombres distinguidos entre mayúsculas y minúsculas.
- La columna de etiqueta: contiene el texto real que se ve en el formulario, pudiendo usarse de manera alternativa columnas de traducción de etiquetas.

Para este proyecto, se procedió con la descarga del archivo en formato XLXS denominado **Documento Estándar XLSForm** del sitio web oficial, para luego empezar con el diseño del formulario de acuerdo a las especificaciones solicitadas por este estándar, donde el usuario se puede guiar con respecto de los tipos de entrada específicos, encabezados de columna, entre otros que forman parte de la sintaxis empleada en XLSForms, como se puede ver en la Figura 4.27.

	A	B	C	D	E	F
1	se	Standard	Notes			
2	Form Properties		Form properties are the column headers on the settings page. Form properties can also be specified on the survey sheet, except for default language.			
3		form_title	if missing assigned to form_id			
4		form_id	if missing assigned to site name			
5		public_key				
6		submission_url				
7		default_language				
8	Phone Properties		According to the documentation here ( <a href="http://opendatakit.org/help/form-design/xls2form/">http://opendatakit.org/help/form-design/xls2form/</a> ) these are also question types, so they go in the survey sheet.			
9		start				
10		end				
11		today				
12		deviceid				
13		imei				
14		phonenumber				
15						
16	Question Types					
17		text				
18		integer				
19		decimal				
20		select_one [options] [or_other]				

Figura 4.27: Documento Estándar XLSForm en Excel

Para empezar el trabajo con los formularios, como primer paso se mantuvieron reuniones con el personal de la Dirección de Oceanografía Naval, para determinar qué campos deberían incluirse en los formularios para recolección de datos, tanto de mediciones de corrientes marinas (veleteo), como de caracterización costera, los cuales se ingresaron en el archivo de Excel, en las hojas de trabajo como lo requiere el XLSForm.

Considerando las necesidades de la Dirección, y los tipos de datos existentes en XLSForm, los formularios fueron diseñados de una manera sencilla, de tal manera que cumpla con el objetivo de reemplazar a los formularios de papel existentes en la actualidad, y quedaron conformados de la siguiente manera, como lo muestran la Tabla 1 y Tabla 2. Es

importante indicar que, en el formulario de caracterización costera, se solicitó incluir un campo multimedia, para evidenciar luego, una imagen de la zona en donde se realizará el estudio, siendo esta imagen almacenada de forma binaria en la base de datos.

### Formulario para mediciones de corrientes marinas (veleteo)

Tabla 1: Hoja de trabajo de encuesta, formulario veleteo

Tipo	Nombre	Etiqueta
select_one siembra	Siembra	Siembra:
select_one tipo	Tipo	Tipo (Siembra/Punto/Recolección):
select_one color	Color	Color de Veleta:
dateTime	fechahora_lanzada	Fecha y Hora:
geopoint	gps_lanzada	Grabar ubicación:
today	Today	(tipo de datos disponibles para la recopilación de metadatos)
deviceid	Deviceid	(tipo de datos disponibles para la recopilación de metadatos)

## Formulario para caracterización costera

Tabla 2: Hoja de trabajo de encuesta, formulario caracterización

Tipo	Nombre	Etiqueta
select_one hora	Hora	Hora de recolección:
select_one investigador	Investigador	Elegir investigador:
select_one localidad	Localidad	Elegir localidad:
date	fec_rec	Fecha de recolección:
select_one estacion	Estación	Elegir estación:
geopoint	Geopoint	Obtener Localización:
decimal	espacio	Espacio (m):
decimal	tiempo	Tiempo (s):
select_one direccion_1	direccion	Dirección:
calculate	calc_velocidad	
decimal	velocidad	Velocidad de la corriente litoral: \${calc_velocidad} (m/s)
select_one direccion_1	corriente_resaca	Corriente de resaca:
decimal	ancho_surf	Ancho de la zona de surf (m):

<b>Tipo</b>	<b>Nombre</b>	<b>Etiqueta</b>
decimal	ancho_rompiente	Ancho de la rompiente (m):
decimal	altura_rompiente	Altura de la rompiente (m):
decimal	distancia_lp_flotador	Distancia LP al flotador (m):
decimal	alineacion_playa	Alineación de playa (grados):
decimal	ancho_playa	Ancho de playa (m):
decimal	velocidad_vientos	Velocidad de vientos (m/s):
integer	direccion_vientos	Dirección de vientos (grados):
integer	ortogonal	Ortogonal:
select_one direccion_2	tipo	Tipo:
decimal	periodo_olas	Periodo de 10 olas:
decimal	obs_1	Observación #1:
decimal	obs_2	Observación #2:
decimal	obs_3	Observación #3:
decimal	obs_4	Observación #4:
decimal	obs_5	Observación #5:
decimal	obs_6	Observación #6:
decimal	obs_7	Observación #7:
decimal	obs_8	Observación #8:
decimal	obs_9	Observación #9:
decimal	obs_10	Observación #10:
calculate	calculate	

Tipo	Nombre	Etiqueta
decimal	promedio	Promedio de 10 olas (cm): \${calculate}
text	observaciones	Ingrese Observaciones:
text	caracter_estacion	Características de la estación:
image	imagen	Tome una foto de la observación
today	today	
deviceid	deviceid	

Es importante indicar que cuando se carga un nuevo formulario en el servidor ODK Aggregate, lo que solicita la plataforma es la definición del mismo en un archivo de formato XML (el cual es el único formato que reconoce), para lo cual se emplea el convertidor de ODK XLSForm en versión online u offline, que permitirá su carga posterior o una pre visualización temporal en un navegador de internet.

Para este proyecto se ha utilizado el convertidor de la versión online accediendo a su sitio web oficial [32], y cargando por cada formulario el archivo en formato Excel, basado en el Documento Estándar XLSForm como se ve en la Figura 4.28, obteniendo como respuesta del proceso de carga, la descarga del XForm (archivo de formato XML) ver Figura 4.29, o previsualizar el diseño del formulario (muy similar a como se lo vería en el dispositivo móvil donde ese ejecutará) en el navegador de

internet, para lo cual emplea la herramienta Enketo, como se presenta en la Figura 4.30.

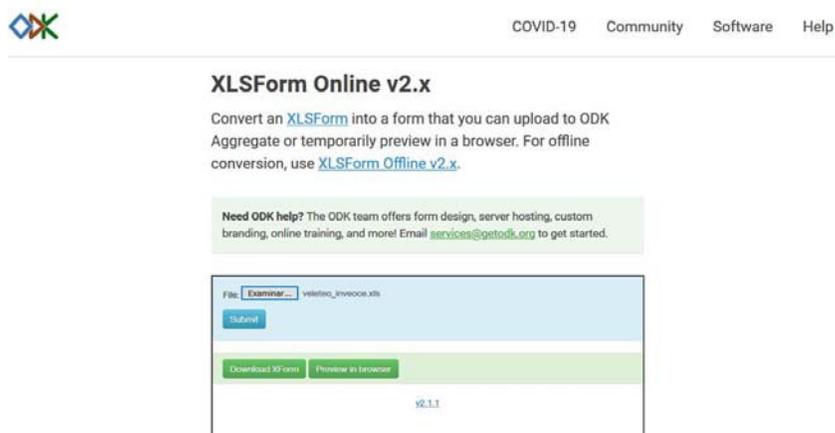


Figura 4.28: Convertidor XLSForm online

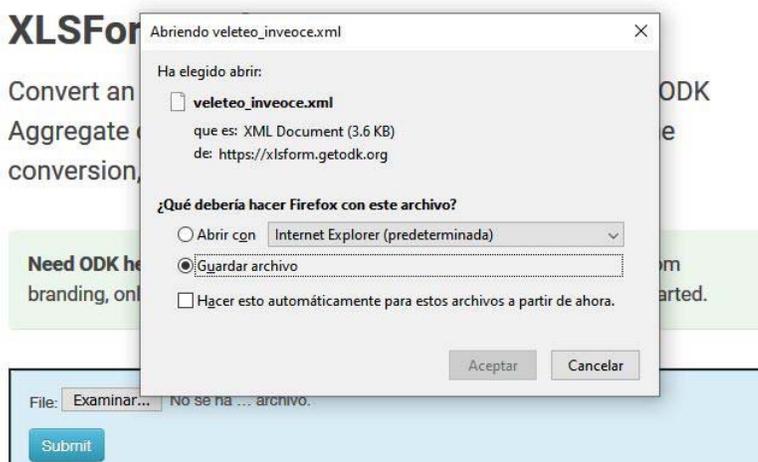


Figura 4.29: Descarga del formulario en formato XML



veleteo\_inveoce

▼ Datos principales

\* Siembra:  
*Elegir numero de siembra*

ninguno seleccionado ▼

\* Tipo (Siembra/Punto/Recoleccion):

ninguno seleccionado ▼

\* Color de Veleta:

ninguno seleccionado ▼

Figura 4.30: Previsualización del formulario en Enketo

### 4.3. Modelado de base de datos

En el proceso de instalación del servidor ODK Aggregate, se muestra un mensaje que indica que la plataforma Open Data Kit ha sido desarrollada y probada en PostgreSQL, presentando una opción para descarga directa de la última versión del motor de base de datos, y una vez elegido, en un proceso de segundo plano se procede con esta acción, solicitando además ajustes con el servidor, puerto y nombre del host, autenticación del ODK Aggregate, nombre de la base de datos y esquema a usar.

Se instaló en el computador utilizado para la documentación de este proyecto, pgAdmin versión 4.17, el cual es una plataforma de administración y desarrollo implementada en código abierto, empleada para PostgreSQL y sus sistemas de bases de datos relacionados, permitiendo desde consultas básicas de SQL, pasando por el monitoreo y configuración avanzada de bases de datos.

Como se indicó anteriormente, ODK Aggregate actúa como una caja negra en relación con la base de datos, ya que automáticamente crea las tablas propias para administración y gestión de la plataforma ODK, así como las tablas empleadas por los formularios donde se almacenarán los datos recolectados, añadiendo en su estructura los campos con similar nombre y tipo de dato necesario, tal cual fue definido en el archivo estándar XLSForm empleado en el proceso de diseño, como lo muestra la Figura 4.31, en donde se despliega la estructura de la tabla desde pgAdmin, y como se explica en la documentación oficial de la estructura de base de datos de ODK Aggregate [34], las tablas de los formularios (XForm) se identifican con el sufijo *\_CORE*, y los datos de cada XForm se mantienen en su propio conjunto independiente de tablas.

ODK Aggregate crea tablas para el almacenamiento de la definición de objetos de los archivos multimedia que estén cargados en un formulario

especifico, y este contenido binario se representa en 3 tablas encadenadas hacia abajo, de arriba abajo `_BIN` -> `_REF` -> `_BLB`; para nombre de archivo, contenido binario que hace referencia a un blob para un adjunto binario determinado, y un segmento de los datos binarios (valor) respectivamente, como puede observarse en la Figura 4.32.

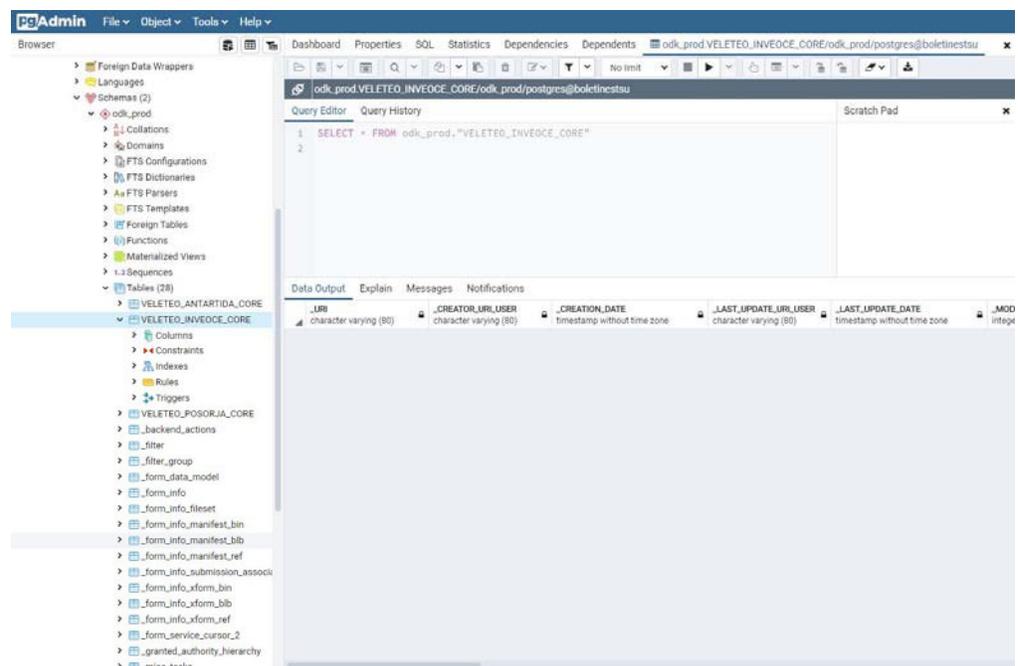


Figura 4.31: Estructura de la tabla de formulario

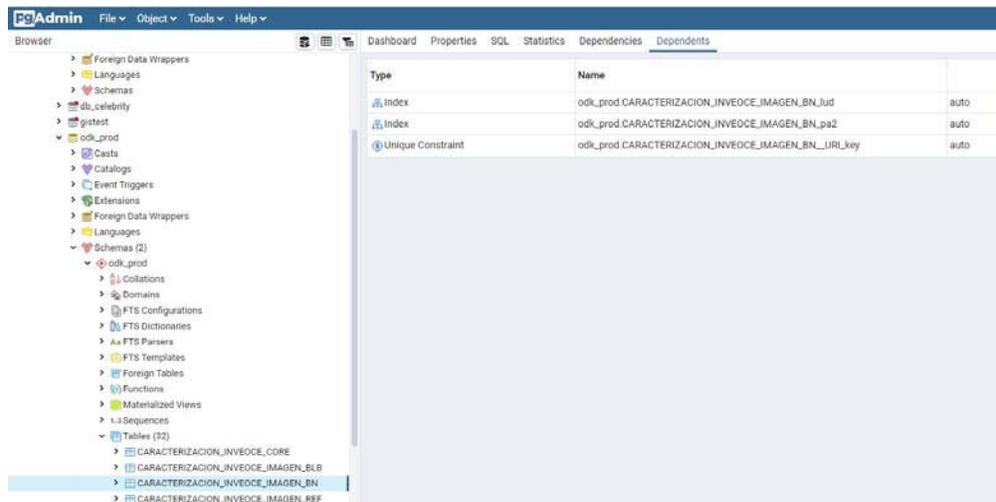


Figura 4.32: Estructura de la tabla de archivos

#### 4.4. Desarrollo de formularios

Como se explicó anteriormente, los formularios son diseñados empleando el Documento Estándar XLSForm, luego de ser convertidos a su definición en formato XM y no requieren otra interacción por parte del usuario, más allá de cargar esta definición en el servidor ODK Aggregate, por lo cual no existe un proceso de desarrollo como tal, ya que luego de esta carga los formularios quedan disponibles para su acceso, descarga al dispositivo móvil y uso en la recolección de datos.

#### **4.5. Desarrollo de sitio web de visualización**

Se desarrolló un sitio web de visualización con un diseño sencillo, el cual contiene un componente de visor geográfico para la presentación de los datos recolectados desde el dispositivo móvil; este sitio fue implementado basándose en el principio de software libre y empleando código abierto.

La plataforma ODK tiene la capacidad de acoplarse con un servidor de mapas GeoServer, lo cual permitirá la publicación de los datos recolectados empleando diversos servicios OWS (OGC Web Services), que facilitan la integración con alguna Infraestructura de Datos Espaciales (IDE).

##### **GeoServer**

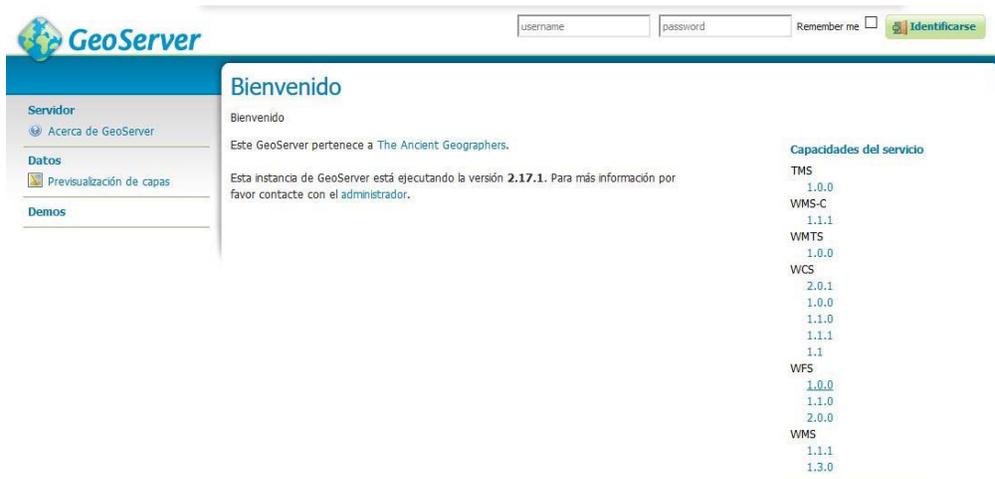
GeoServer es un servidor de código abierto de mapas e información geográfica para compartir datos geoespaciales en diferentes formatos, que permite la publicación de datos de cualquier fuente de datos espaciales, alcanzando la interoperabilidad de información geográfica con clientes web y clientes de escritorio, empleando estándares abiertos.

Implementa servicios OWS como Web Feature Service (WFS), Web Map Service (WMS), Web Coverage Service (WCS), que ofrecen entre otras

opciones, la visualización y descarga de datos desde un geoportal o sitio web, permitiendo la definición de estilos para la representación de los mismos, de acuerdo a los valores de los atributos que contienen; así también, GeoServer emplea Apache Tomcat, el mismo servidor de aplicaciones que ODK Aggregate, lo cual permite una mejor integración en la arquitectura de la solución propuesta.

En el presente trabajo, se utiliza el servicio WFS que permitirá la edición de información y consulta de los datos espaciales en línea, pudiendo ser de geometrías, así como de los atributos.

La instalación de este servidor es sencilla, ya que solo requiere la descarga del paquete WAR para el despliegue en el servidor Tomcat, proveyendo una interfaz de usuario intuitiva y amigable, así como un buen rendimiento. Se accede al GeoServer desde la dirección <http://10.130.22.110:8080/geoserver/web/>, como se muestra en la Figura 4.33.



**Figura 4.33: Pantalla principal del GeoServer**

Una vez instalado el servidor, se necesita acceder desde el GeoServer a los datos cargados desde ODK Aggregate para su publicación, para lo cual se requiere establecer una conexión con la base de datos de la instancia ODK creada con anterioridad, y dentro del mismo (GeoServer) crear un almacén seleccionando el origen de los datos que se quiere publicar, siendo el caso de este proyecto PostGIS, debiéndose proporcionar todos los parámetros de conexión con la base de datos, como lo muestran la Figura 4.34 y Figura 4.35.

Identificado como admin. [Cerrar sesión](#)

## Nuevo origen de datos

Seleccione el tipo de origen de datos que desea configurar

### Origenes de datos vectoriales

- Directory of spatial files (shapefiles) - Takes a directory of shapefiles and exposes it as a data store
- GeoPackage - GeoPackage
- PostGIS - PostGIS Database
- PostGIS (JNDI) - PostGIS Database (JNDI)
- Properties - Allows access to Java Property files containing Feature information
- Shapefile - ESRI(tm) Shapefiles (\*.shp)
- Web Feature Server (NG) - Provides access to the Features published a Web Feature Service, and the ability to perform transactions on the server (when supported / allowed).

### Origenes de datos raster

- ArcGrid - ARC/INFO ASCII GRID Coverage Format
- GeoPackage (mosaic) - GeoPackage mosaic plugin
- Geo TIFF - Tagged Image File Format with Geographic information
- ImageMosaic - Image mosaicking plugin
- WorldImage - A raster file accompanied by a spatial data file

### Otros orígenes de datos

- WMS - Configura un Web Map Service en cascada
- WMTS - Cascades a remote Web Map Tile Service

**Servidor**

- Estado del servidor
- Logs de GeoServer
- Información de contacto
- Acercar de GeoServer

**Datos**

- Previsualización de capas
- Espacios de trabajo
- Almacenes de datos
- Capas
- Grupos de capas
- Estilos

**Servicios**

- WCS
- WMTS
- WFS
- WMS

**Settings**

- Global
- JAI
- Coverage Access

**Cacheado de Teselas**

- Capas en caché
- Valores por defecto de cacheado
- Conjuntos de malla

Figura 4.34: Origen de datos en almacén de GeoServer

Identificado como admin. [Cerrar sesión](#)

## Nuevo origen de datos vectoriales

Agregar un nuevo origen de datos vectoriales

PostGIS  
PostGIS Database

### Información básica del almacén

Espacio de trabajo \*

ide

Nombre del origen de datos \*

Descripción

Habilitado

### Parámetros de conexión

host \*

localhost

port \*

5432

database

schema

public

user \*

password

.....

**Servidor**

- Estado del servidor
- Logs de GeoServer
- Información de contacto
- Acercar de GeoServer

**Datos**

- Previsualización de capas
- Espacios de trabajo
- Almacenes de datos
- Capas
- Grupos de capas
- Estilos

**Servicios**

- WCS
- WMTS
- WFS
- WMS

**Settings**

- Global
- JAI
- Coverage Access

**Cacheado de Teselas**

- Capas en caché
- Valores por defecto de cacheado
- Conjuntos de malla

Figura 4.35: Origen de datos vectoriales PostGIS

Es necesario crear una nueva capa, la cual contendrá los datos obtenidos de la base, y los publicará como servicios WMS y WFS, para lo cual se requiere en la capa configurar una vista SQL, previamente creada de una consulta del tipo “SELECT \* FROM...”, la cual fue implementada en el servidor de base de datos Postgre empleando la herramienta gráfica de administración pgAdmin. Esto se muestra en la Figura 4.36 y Figura 4.37.

De esto se obtiene que los atributos se carguen automáticamente y la identificación del campo geometría también se realizará de manera automática; así también, al ser el origen de datos una base Postgre, no se requiere seleccionar un campo como identificador.

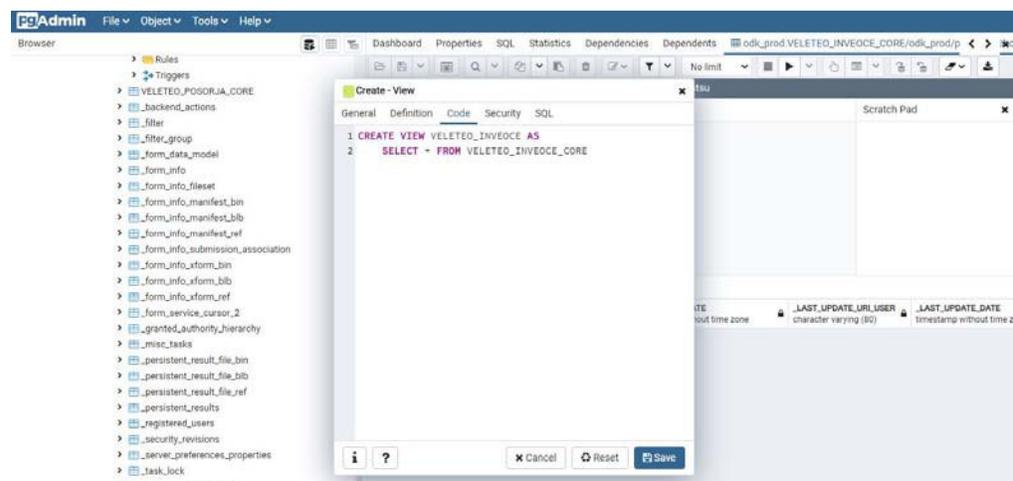


Figura 4.36: Creación de vista en la base de datos

Identificado como admin. [Cerrar sesión](#)

### Crear una nueva vista SQL

Definir una nueva vista SQL y configurar sus columnas identificadas y la geometría

Nombre de la vista

Sentencia SQL

Parámetros de la vista SQL  
 Averiguar parámetros a partir del SQL    [Agregar parámetro](#)    [Eliminar seleccionados](#)

<input type="checkbox"/>	Nombre	Valor por defecto	Validar la expresión regular
<input checked="" type="checkbox"/>	Escapar caracteres especiales de SQL		

Atributos  
 Refrescar     Averiguar tipo de geometría e identificador de CRS

**Figura 4.37: Creación de capa en GeoServer**

Para comprobar que la capa se encuentra correctamente creada, tiene conexión y obtiene datos desde la base, se emplea la opción Previsualización de capas, y en formato se selecciona WFS -> GeoJSON, el cual servirá como fuente de datos para el visualizador geográfico web, como se explicará más adelante. La comprobación y formato de salida se muestran en la Figura 4.38 y Figura 4.39, respectivamente.



Figura 4.38: Previsualización de capas en GeoServer

JSON	Datos sin procesar	Cabeceras
Guardar	Copiar	Contraer todo
		Expandir todo
Filtrar JSON		
type:		"FeatureCollection"
totalFeatures:		36
features:		
0:		
type:		"Feature"
id:		"veleteo_web.fid-1d68646b_1718012bae2_-6bf0"
geometry:		
type:		"Point"
coordinates:		
0:		-80.05580641
1:		-3.25516024
geometry_name:		"geom"
properties:		
_URI:		"uuid:118c7cd2-9b00-46d9-bb92-91874acfaf2d"
_CREATOR_URI_USER:		"anonymousUser"

Figura 4.39: Respuesta de servicio en formato GeoJSON

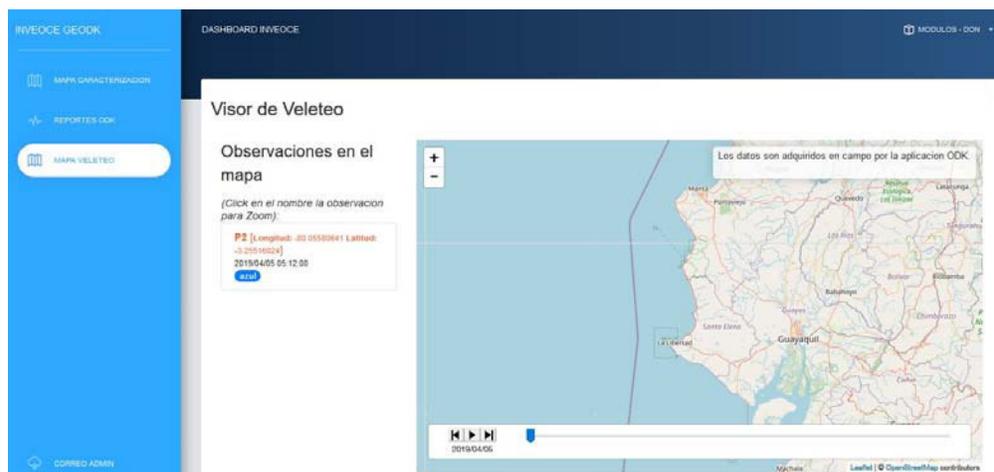
Estos pasos se deben realizar para cada uno de los formularios que se implementen en la aplicación móvil, y una vez que tengan creadas las capas, se tienen listos los datos para ser servidos en los formatos

anteriormente nombrados, los cual serán empleados para su presentación en un geoportal, visualizador geográfico web o GIS de escritorio.

### **Visualizador geográfico web**

Para la presentación de los datos recolectados con la aplicación móvil y formularios basados en la plataforma ODK, se desarrolló un micrositio web de diseño sencillo, y se implementaron páginas en las que se agregaron componentes como visualizador geográfico, herramientas de búsqueda de datos, así como la generación de reportes.

Para la implementación de las páginas web, se empleó una plantilla gratuita descargada del Internet, la cual está basada en HTML5, con soporte para Bootstrap 3.X (Framework CSS y Javascript diseñado para la elaboración de interfaces limpias y con un diseño responsivo), y jQuery (librería de código abierto, que simplifica el trabajo de programar en JavaScript, permitiendo añadir interactividad a un sitio web), como lo muestra la Figura 4.40.



**Figura 4.40: Vista general de plantilla web**

Para la representación visual de los datos en línea, se embebió un visualizador geográfico basado en Leaflet, librería Javascript de código abierto que data del año 2008, que se emplea ampliamente para el despliegue de mapas en la web. Es una herramienta sencilla y ligera, sin que esto suponga una limitación para sus capacidades técnicas, ya que es comparable con cualquier otro competidor dentro del ámbito del Web Mapping (término para referirse al diseño, aplicación, generación, visualización de datos geospaciales en la web) como, por ejemplo, OpenLayers.

La información presentada se obtiene desde la base de datos empleando la vista generada anteriormente, en forma de geometría de tipo punto, mediante el servicio WFS que se encuentra activo en el servidor

GeoServer, el cual devuelve como respuesta un archivo GeoJSON, el mismo que permite la visualización espacial de la visualización de los puntos, así como la información tabular y multimedia (en caso de requerirse) de la observación efectuada.

Posterior a embeber el mapa base con Leaflet, se procedió con la instalación del plug-in Leaflet.timeline, el mismo que permite mostrar cualquier dato geoespacial cambiante a lo largo del tiempo, desde puntos hasta polígonos, mediante la implementación de una línea de tiempo, que contiene controles de ejecución, pausa, retroceso y adelanto, y que basado en el campo fecha de los datos almacenados, se ajusta perfectamente para la representación en pantalla del dato recolectado, permitiendo el movimiento de manera sencilla entre los diferentes rangos de fechas que se cargan desde la consulta, como lo muestra la Figura 4.41.

## Visor de Veteo



**Figura 4.41: Control de línea de tiempo en visualizador geográfico**

Por último, para la presentación tabular de los datos se descargó e instaló en el servidor DataTables, un plug-in para la librería jQuery, además de la extensión Botones de exportación HTML5 (HTML5 export buttons), que permiten realizar copias del contenido de la tabla y exportación de datos a formatos Excel, CSV y PDF. Así también, en esta tabla se podrá realizar la búsqueda instantánea de datos, y ordenamiento entre columnas, como se explicó previamente al inicio de este capítulo. La implementación de la tabla y botones de exportación se muestran en la Figura 4.42.

TODOS LOS DATOS

Copy Excel CSV PDF - VELETEO

Search: amarillo

HORA MUESTREO	TIPO	SIEMBRA	CARACTERISTICA	COLOR	FECHAHORA_LANZADA
2020/03/05 08:13:00	Recogida	Siembra #5	superficial	amarillo	2020-03-05T18:13:00Z
2020/03/05 10:50:00	Recogida	Siembra #2	superficial	amarillo	2020-03-05T20:50:00Z
2020/03/05 09:55:00	Punto #5	Siembra #2	superficial	amarillo	2020-03-05T19:55:00Z
2020/03/05 10:22:00	Punto #4	Siembra #2	superficial	amarillo	2020-03-05T20:22:00Z
2020/03/05 05:23:00	Punto #2	Siembra #1	superficial	amarillo	2020-03-05T15:23:00Z
2020/03/05 06:20:00	Punto #2	Siembra #2	superficial	amarillo	2020-03-05T16:20:00Z
2020/03/05 07:06:00	Punto #2	Siembra #3	superficial	amarillo	2020-03-05T17:06:00Z
2020/03/05 07:44:00	Punto #2	Siembra #4	superficial	amarillo	2020-03-05T17:44:00Z

Showing 1 to 8 of 8 entries (filtered from 36 total entries)

Previous 1 Next

**Figura 4.42: Tabla de datos y botones de exportación**

#### 4.6. Plan de pruebas e implementación de formularios

Para comprobar el correcto funcionamiento de la aplicación móvil, se planificaron y ejecutaron las siguientes pruebas:

- Se descargaron e instalaron en el equipo móvil de un funcionario de la institución los dos formularios desarrollados, y se solicitó autorización a la Dirección del instituto para el uso de la aplicación a propósito que el personal iba a cumplir comisión de servicio en la provincia de El Oro

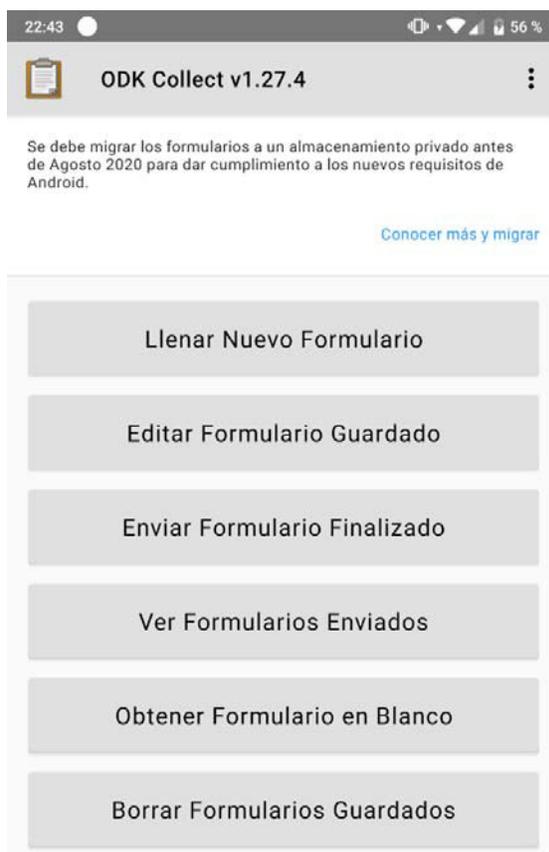
en el mes de marzo de 2020, y que el objetivo de la misma, así como las actividades de campo correspondientes tenían otra finalidad. Los formularios fueron empleados para la recolección de datos de veleteo y caracterización costera, durante un determinado lapso de tiempo, funcionando de manera exitosa permitiendo el almacenamiento de los datos en la base.

- Además, los formularios fueron descargados e instalados en el equipo móvil del suscrito, para la ejecución de los mismos simulando actividades de campo, dentro de las instalaciones del instituto en la ciudad de Guayaquil.

Por último, desde el dispositivo móvil, y configurando la opción de Servidor - Usuario, de la dirección URL del servidor ODK Aggregate, así como el nombre de usuario y contraseña, se tienen acceso a los formularios diseñados, los cuales pueden ser descargados para empezar con la recolección de datos.

Desde el menú principal de la aplicación ODK Collect, se selecciona la opción Obtener Formulario en Blanco, como lo muestra la Figura 4.43, mostrándose luego el listado y formularios almacenados en el servidor, tal como se observa en la Figura 4.44.

De este listado se marca el o los formularios requeridos, y se presiona el botón Obtener los Seleccionados, con lo cual se descarga el formulario al dispositivo móvil, y se vuelve al menú principal para seleccionar la opción Llenar nuevo formulario, donde se presentará con cual formulario se desea trabajar como lo muestran la Figura 4.45 con el formulario para datos de veleteo y la Figura 4.46 con el formulario para datos de caracterización costera.



**Figura 4.43: ODK Collect menú principal**

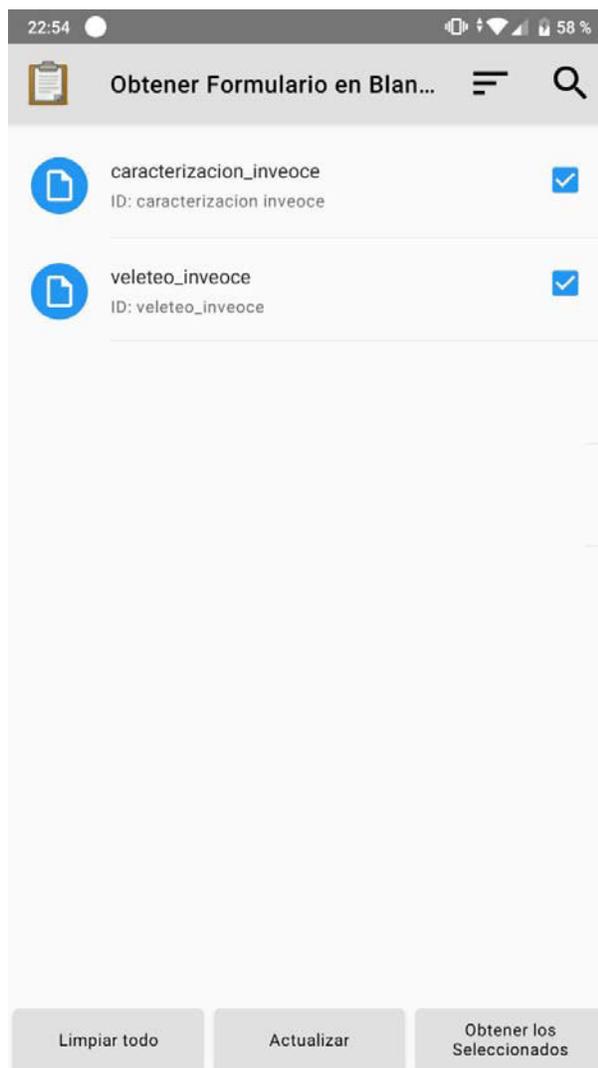


Figura 4.44: Listado de formularios disponibles

23:13 61 %

veleteo\_inveoce

Datos principales

\* **Siembra:**  
Elegir numero de siembra

Seleccione una respuesta ▾

---

Figura 4.45: Formulario para datos de veleteo

The image shows a mobile application interface for data entry. At the top, the status bar shows the time 23:15 and battery level at 61%. The app title is 'caracterizacion\_inveoce'. Below the title, there is a section labeled 'Datos principales'. The form contains five required fields, each marked with a red asterisk:

- \* Hora de recoleccion:** A dropdown menu with the text 'Seleccione una respuesta'.
- \* Elegir investigador:** A dropdown menu with the text 'Seleccione una respuesta'.
- \* Elegir localidad:** A dropdown menu with the text 'Seleccione una respuesta'.
- \* Fecha de recoleccion:** A button labeled 'Seleccione la fecha' and a text field below it containing 'No se ha seleccionado una fecha'.
- \* Elegir estacion:** A dropdown menu with the text 'Seleccione una respuesta'.

Figura 4.46: Formulario para datos de caracterización costera

## **CAPÍTULO 5**

### **ANÁLISIS DE RESULTADOS**

#### **5.1. Evaluación de la solución implementada**

Una vez implementada la aplicación para recolección móvil de datos de mediciones de corrientes marinas y caracterización costera empleando la plataforma Open Data Kit (ODK), se realiza la evaluación de la misma considerando los problemas que ha resuelto de manera efectiva.

Como evaluación general, al seleccionar la plataforma ODK como herramienta de diseño y desarrollo de la aplicación, se consideró que se basa en código abierto, no conlleva gastos de licenciamiento, no depende de un entorno de desarrollo integrado o lenguaje de programación de tipado, el hardware requerido puede ser de mediana prestación sin que

se requiera una gran inversión, funciona en los dos sistemas operativos móviles más empleados a nivel mundial (Android e iOS, aunque para el presente trabajo solo se desarrolló para el primero), no requiere experiencia previa por parte del equipo técnico, no supone una gran curva de aprendizaje, es intuitiva en su uso y el producto final son formularios livianos totalmente portables, que soportan una gran cantidad de funcionalidades.

Los problemas que ha resuelto la implementación de esta aplicación para recolección móvil de datos son:

- Daños en los formularios: al no requerirse tener en las actividades de campo el llenado de datos en formularios físicos de papel, ya no existe el problema del deterioro de los mismos, ya sea por manipulación, o factores ambientales como el sol o lluvia; así también no existirá el problema del traslape o pérdida de estos documentos al retorno a las oficinas del instituto. Así también, elimina el uso de recursos papel e impresión, y se pueden realizar cambios en los formatos de forma más rápida.
- Integridad de los datos: la aplicación garantiza que los datos no serán modificados luego de su recolección, siendo además válidos y consistentes.

- Sistematización de formularios: los formularios se diseñan dinámicamente, permitiendo cambios de manera rápida y sencilla, permitiendo ejecutar además la validación de campos obligatorios y tipos de datos, eliminando la posibilidad de error humano.
- Disminución de implementos: el técnico o investigador requerirá menos cantidad de implementos en las actividades de campo, ya que no deberá llevar consigo, por ejemplo: tablero, formularios en papel, lápiz o esferográfico, equipo GPS, cámara fotográfica, y equipo de comunicación, en vista que, para la recolección de los datos, necesitará únicamente el dispositivo móvil, con los formularios descargados previamente desde el servidor, sin que sea tampoco necesaria una conexión a internet, como anteriormente se indicó.
- Disminución de errores: la cantidad de errores en la recolección de los datos disminuirá, ya que el técnico está más enfocado en el trabajo que realiza y no en controlar que el formulario se encuentre en buen estado, así como de los implementos que requiera para la recolección de datos, lo cual significa mejoras en la calidad y consistencia de los mismos. Por otro lado, se obviarán las interpretaciones de escritura que se producen al llenar manualmente los formularios en campo, así como inconsistencias grafológicas que provoquen errores en la transcripción de los datos en hojas de cálculo.

- La aplicación móvil ha permitido la optimización en los tiempos de recolección de datos, así como la sistematización de los mismos, suponiendo entre otras ventajas, el que permite contar con información en formato digital y ordenada, pudiéndose realizar mediante el uso de un único dispositivo, el registro alfanumérico, de localización geográfica y fotográfico de observaciones realizadas en trabajos de campo, validar los campos solicitados, y por último proporcionar de manera inmediata y completamente automatizada los datos recolectados, para la posterior visualización y descarga en la institución.

Es importante mencionar que la parte técnica también tuvo sus retos, como el procedimiento de visualización de los resultados, los cuales la plataforma solo permite almacenar, pero la consulta se implementa mediante programación.

Al trabajar directamente con la base de datos de la plataforma, permite la rapidez en el acceso de la información y creación de herramientas como visualizadores y gráficos estadísticos. El uso del servidor que conecte directamente el ODK a un servicio web, hizo más sencillo el trabajo de integración, y dejó listo el desarrollo de la parte visual o front-end.

En la única comisión de servicios que se logró incluir a la aplicación en las actividades de campo, esta se usó de manera rápida en las salidas, las mismas que al principio se hacían con papel por la incertidumbre del uso de la aplicación, y considerando el poco tiempo que se la tuvo en uso para mediciones en campo, se capacitó brevemente al personal, porque debía hacerse el trabajo en diversas estaciones o lugares geográficos.

Se hizo sencillo el registro de los datos, sin embargo, al ser 2 registros a la vez (veleteo y caracterización) era considerado doble trabajo por el técnico. Es así, que como experiencia se anota que ese día en el bote se perdieron formularios de registro en formato papel, y los mismos contenían datos de 2 estaciones seguidas, pero se mencionó que estos datos también habían sido registrados en la aplicación, lo que provocó que se recuperara el tiempo de trabajo de forma digital y de fácil acceso.

Como evaluación general, una de las ventajas más grandes que se logró con la aplicación, es el uso del dispositivo móvil en sitios sin conexión a internet, puesto que la sincronización es automática al momento de llegar a un lugar con acceso a internet.

Así también se indica que la escalabilidad de la plataforma permite crear diferentes tipos de encuesta o formularios de levantamiento, no solo con

la capacidad de ingresar texto u opciones predefinidas, sino con objetos de tipo multimedia como audios, videos e imágenes, lo que permite un mejor monitoreo visual del entorno, y mejorar la experiencia de las actividades en campo.

## **5.2. Resultados de la solución implementada**

Para obtener resultados de la solución implementada, se efectuó un cuestionario a 12 técnicos de la Dirección de Oceanografía (ya que no todos los funcionarios de esta dirección son oceanógrafos), los mismos que, aun cuando no participan constantemente en los procesos de recolección de datos de mediciones de corrientes marinas y caracterización costera, tienen conocimiento por su formación académica de cómo ejecutarlos, y lo han efectuado por lo menos una vez en el último año, y que emplearon la aplicación sea en prueba de campo o dentro de la institución.

Los resultados obtenidos a partir de este cuestionario se encuentran tabulados en la Tabla 3, y su representación gráfica en las Figuras 5.1, 5.2, 5.3, 5.4, 5.5 y 5.6.

En el Anexo se muestra el cuestionario empleado en el presente trabajo de titulación.

Tabla 3: Resultados tabulados de cuestionario a técnicos

Pregunta	Siempre	A veces	Casi nunca
1. ¿Ha tenido problemas para la manipulación y conservación de formularios de papel para la recolección de datos de mediciones de corrientes marinas y caracterización costera?	6	4	2

Pregunta	Siempre	A veces	Casi nunca
2. ¿Ha encontrado problemas en los formularios de papel con su estado, integridad de datos, anotaciones inentendibles o campos vacíos?	2	6	4

Pregunta	Muy compleja	Compleja	Sencilla
3. ¿Usted cómo calificaría la interfaz de usuario de la aplicación?	1	2	9

Pregunta	Muy bueno	Bueno	Regular	Malo
4. ¿Usted cómo calificaría el tiempo de respuesta que toma el ingreso de datos en la aplicación móvil?	7	4	1	0

Pregunta	Si	No
5. ¿Considera que es más sencillo para la recolección de datos usar una aplicación móvil, que formularios de papel e instrumentos adicionales?	11	1

Pregunta	Si	No
6. ¿Cree Usted que la institución debería desarrollar e implementar diversos tipos de formularios para la recolección de datos empleando la plataforma ODK?	8	4

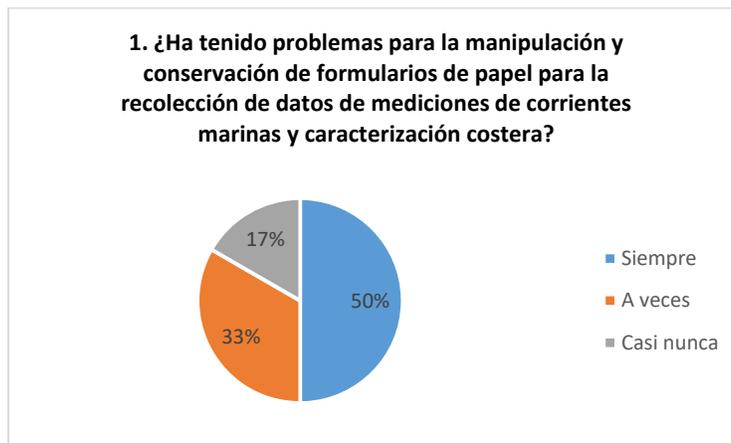


Figura 5.1: Gráfico pregunta 1

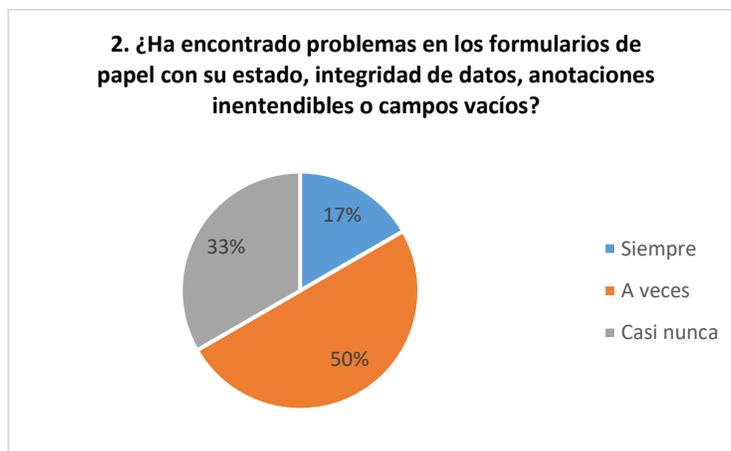
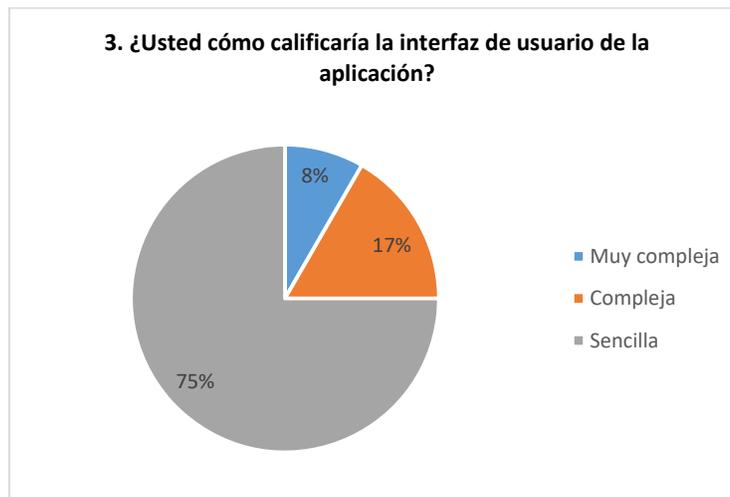
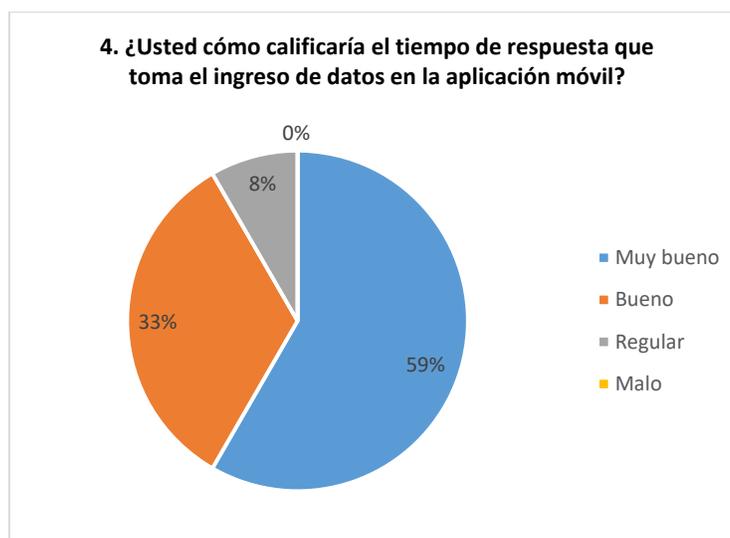


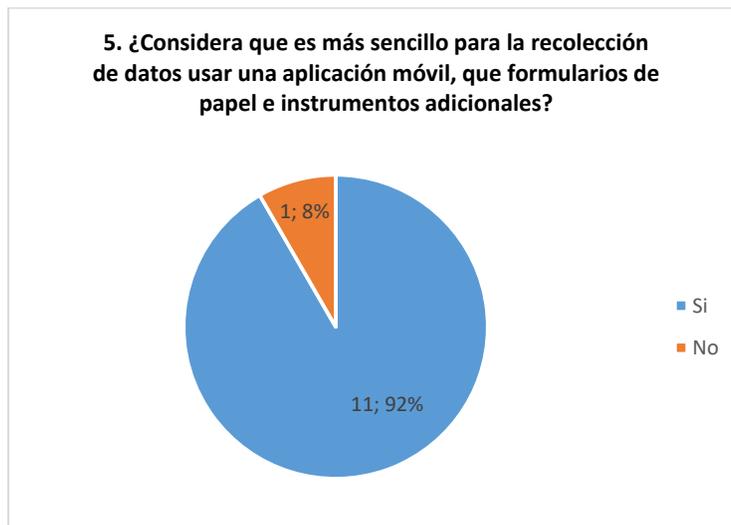
Figura 5.2: Gráfico pregunta 2



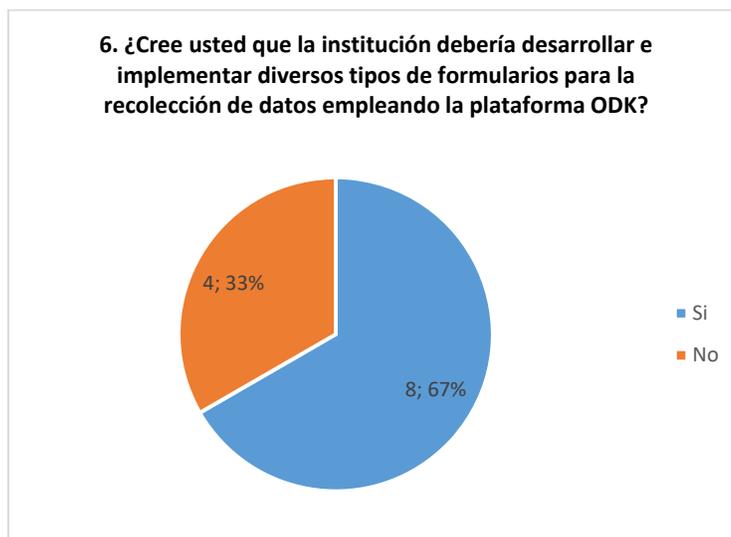
**Figura 5.3: Gráfico pregunta 3**



**Figura 5.4: Gráfico pregunta 4**



**Figura 5.5: Gráfico pregunta 5**



**Figura 5.6: Gráfico pregunta 6**

## **CONCLUSIONES Y RECOMENDACIONES**

### **Conclusiones**

1. El análisis de los requerimientos de los técnicos e investigadores, llevó al desarrollo de formularios digitales que permitan el ingreso de datos de una manera sencilla y ágil, así como reducir los problemas de pérdida de datos en campo y automatizar el envío de información.
2. Se diseñaron formularios que permiten la recolección móvil de datos, empleando una herramienta gratuita de código abierto, de fácil aprendizaje e implementación, de bajo costo económico, que supondrá mejoras al levantamiento de información en las actividades de campo.
3. La implementación del sitio web para visualización de datos y generación de reportes, permite contar con una herramienta de consulta y presentación rápida de los datos.

4. La puesta en marcha de la aplicación, de acuerdo a los resultados obtenidos posterior a su implementación, indican que es bien recibida por los usuarios, los mismos que ven en ella una herramienta que facilitará su trabajo en actividades de campo, minimizará los errores, preservará los datos y permitirá realizar consultas para futuras actividades.

### **Recomendaciones**

1. Utilizar el presente trabajo como producto base para el desarrollo de una aplicación para recolección móvil de datos que funcione de manera permanente en la institución.
2. Establecer como proceso de mejora continua institucional, la migración de formularios físicos de papel al empleo de una aplicación para recolección móvil de datos, cuando la situación lo amerite.
3. Capacitar al personal de sistemas de la Dirección de Tecnología de la institución en el uso de la plataforma ODK para desarrollo futuro de nuevos formularios.
4. Mejorar el proceso de investigación y actividades de campo, empleando tecnología gratuita para la recolección de la información.

## BIBLIOGRAFÍA

- [1] L. Fuentes, J. M. Troya, and A. Vallecillo, “Lección 1 Desarrollo de Software Basado en Componentes”.
- [2] R. Pressman, *Ingeniería del Software. Un enfoque práctico*. 2002.
- [3] M. A. Rojas, J. Carlos, and M. García, “Componentes de Software 1 INTRODUCCIÓN Y PRINCIPIOS BÁSICOS DEL DESARROLLO DE SOFTWARE BASADO EN COMPONENTES,” 2004.
- [4] “PROGRAMACIÓN ORIENTADA A OBJETOS – La Inter |.” [Online]. Available: <https://www.lainter.edu.mx/blog/2018/03/18/programacion-orientada-a-objetos/>. [Accessed: 03-Dec-2019].
- [5] “MWC 2018: El número de líneas móviles supera por primera vez a la población mundial | Tecnología | EL PAÍS.” [Online]. Available: [https://elpais.com/tecnologia/2018/02/27/actualidad/1519725291\\_071783.html](https://elpais.com/tecnologia/2018/02/27/actualidad/1519725291_071783.html). [Accessed: 14-Jan-2020].
- [6] M. C. G. Mantilla, L. L. C. Ariza, and B. M. Delgado, “Metodología para el desarrollo de aplicaciones móviles,” *Tecnura*, vol. 18, no. 40, pp. 20–35, 2014.
- [7] M. F. Goodchild, “Citizens as sensors: the world of volunteered geography,” *GeoJournal*, vol. 69, no. 4, pp. 211–221, 2007.
- [8] M. Minghini, “Multi-dimensional GeoWeb platforms for citizen science

and civic engagement applications,” Italy, 2014.

- [9] “46,4% de usuarios del Servicio Móvil Avanzado poseen un smartphone – Agencia de Regulación y Control de las Telecomunicaciones.” [Online]. Available:  
  
<https://www.arcotel.gob.ec/464-de-usuarios-del-servicio-movil-avanzado-poseen-un-smartphone/>. [Accessed: 10-Feb-2020].
- [10] G. Mu’ammar, “Embedding data quality in " GPS-tagged " field data,” no. November, 2007.
- [11] D. Browning, “Mobile Solutions for GIS Data Collection and Display,” p. 55, 2011.
- [12] C. Hartung, Y. Anokwa, W. Brunette, A. Lerer, C. Tseng, and G. Borriello, “Open data kit: Tools to build information services for developing regions,” *ACM Int. Conf. Proceeding Ser.*, 2010.
- [13] Friederike, “Benchmarking of Mobile Data Collection Solutions WHAT ASPECTS TO CONSIDER WHEN CHOOSING A TOOL/PLATFORM,” no. January, 2017.
- [14] “Welcome to ODK’s Docs! — Open Data Kit Docs.” [Online]. Available: <https://docs.opendatakit.org/>. [Accessed: 20-Feb-2020].
- [15] “ODK Build — Open Data Kit Docs.” [Online]. Available: <https://docs.opendatakit.org/build-intro/>. [Accessed: 20-Feb-2020].

- [16] “XForms 2.0 - W3C XForms Group Wiki (Public).” [Online]. Available: [https://www.w3.org/MarkUp/Forms/wiki/XForms\\_2.0](https://www.w3.org/MarkUp/Forms/wiki/XForms_2.0). [Accessed: 20-Feb-2020].
- [17] “XLSForm — Open Data Kit Docs.” [Online]. Available: <https://docs.opendatakit.org/xlsform/>. [Accessed: 20-Feb-2020].
- [18] “ODK Collect — Open Data Kit Docs.” [Online]. Available: <https://docs.opendatakit.org/collect-intro/>. [Accessed: 20-Feb-2020].
- [19] “ODK Aggregate — Open Data Kit Docs.” [Online]. Available: <https://docs.opendatakit.org/aggregate-intro/>. [Accessed: 21-Feb-2020].
- [20] D. F. Pacheco Prado, “Infraestructuras de datos espaciales en dispositivos móviles inteligentes,” Universidad del Azuay, 2013.
- [21] “Estándares del Open GeoSPatial Consortium — OSGeo-Live 10.5 Documentation.” [Online]. Available: <https://live.osgeo.org/archive/10.5/es/standards/standards.html>. [Accessed: 26-Feb-2020].
- [22] “Web Map Service (WMS) — OSGeo-Live 10.5 Documentation.” [Online]. Available: [https://live.osgeo.org/archive/10.5/es/standards/wms\\_overview.html](https://live.osgeo.org/archive/10.5/es/standards/wms_overview.html). [Accessed: 26-Feb-2020].
- [23] “Web Feature Service (WFS) — OSGeo-Live 10.5 Documentation.”

[Online]. Available:

[https://live.osgeo.org/archive/10.5/es/standards/wfs\\_overview.html](https://live.osgeo.org/archive/10.5/es/standards/wfs_overview.html).

[Accessed: 26-Feb-2020].

- [24] “Web Coverage Service (WCS) — OSGeo-Live 10.5 Documentation.”

[Online]. Available:

[https://live.osgeo.org/archive/10.5/es/standards/wcs\\_overview.html](https://live.osgeo.org/archive/10.5/es/standards/wcs_overview.html).

[Accessed: 27-Feb-2020].

- [25] “GeoServer — OSGeoLive 13.0 Documentation.” [Online]. Available:

[https://live.osgeo.org/es/overview/geoserver\\_overview.html](https://live.osgeo.org/es/overview/geoserver_overview.html). [Accessed:

27-Feb-2020].

- [26] “¿Qué es una geodatabase?—ArcGIS Help | ArcGIS Desktop.” [Online].

Available:

[https://desktop.arcgis.com/es/arcmap/latest/manage-](https://desktop.arcgis.com/es/arcmap/latest/manage-data/geodatabases/what-is-a-geodatabase.htm)

[data/geodatabases/what-is-a-geodatabase.htm](https://desktop.arcgis.com/es/arcmap/latest/manage-data/geodatabases/what-is-a-geodatabase.htm). [Accessed: 27-Feb-

2020].

- [27] “PostgreSQL: About.” [Online]. Available:

<https://www.postgresql.org/about/>. [Accessed: 27-Feb-2020].

- [28] “PostGIS — OSGeoLive 13.0 Documentation.” [Online]. Available:

[https://live.osgeo.org/es/overview/postgis\\_overview.html](https://live.osgeo.org/es/overview/postgis_overview.html). [Accessed:

27-Feb-2020].

- [29] “NOAA’s National Ocean Service Education: Currents: How Are Currents Measured?” [Online]. Available:  
  
<https://oceanservice.noaa.gov/education/kits/currents/07measure1.html>  
. [Accessed: 28-May-2020].
- [30] “Form Building Tools — ODK Docs.” [Online]. Available:  
<https://docs.getodk.org/form-tools/>. [Accessed: 23-Jul-2020].
- [31] “XLSForm Docs.” [Online]. Available: <http://xlsform.org/en/>. [Accessed: 23-Jul-2020].
- [32] “ODK - XLSForm Online v2.x.” [Online]. Available:  
<https://getodk.org/xlsform/>. [Accessed: 11-Aug-2020].
- [33] “Aggregate Database Structure · getodk/getodk Wiki · GitHub.” [Online]. Available: <https://github.com/getodk/getodk/wiki/Aggregate-Database-Structure>. [Accessed: 17-Aug-2020].

## ANEXO

El presente cuestionario forma parte del trabajo de titulación, por lo que solicito su cooperación contestando cada una de las preguntas de forma objetiva y veraz. La información aquí ingresada es confidencial y reservada, ya que solo será utilizada sólo para fines de investigación. Quedo agradecido de la atención brindada.

Instrucciones: Lea cuidadosamente las preguntas y elija la opción que mejor se adapte a su preferencia.

**1. ¿Ha tenido problemas para la manipulación y conservación de formularios de papel para recolección de datos de mediciones de corrientes marinas y caracterización costera?**

Siempre       A veces       Casi nunca

**2. ¿Ha encontrado problemas en los formularios de papel con su estado, integridad de datos, anotaciones inentendibles o campos vacíos?**

Siempre       A veces       Casi nunca

**3. ¿Usted cómo calificaría la interfaz de usuario de la aplicación?**

Muy compleja       Compleja       Sencilla

**4. ¿Usted cómo calificaría el tiempo de respuesta que toma el ingreso de datos en la aplicación móvil?**

Muy bueno     Bueno     Regular     Malo

**5. ¿Considera que es más sencillo para la recolección de datos usar una aplicación móvil, que formularios de papel e instrumentos adicionales?**

Si     No

**6. ¿Cree usted que la institución debería desarrollar e implementar diversos tipos de formularios para la recolección de datos empleando la plataforma ODK?**

Si     No