



FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA PROFESIONAL DE INGENIERÍA DE COMPUTACIÓN Y SISTEMAS

**AGENTE CIUDADANO – APLICACIÓN MÓVIL PARA REPORTAR
LA UBICACIÓN DE VEHÍCULOS ROBADOS**

PRESENTADA POR
SOFÍA GIULIANA ACUÑA NIÑO
SILVIA ELENA BARBA QUEZADA

TESIS
PARA OPTAR EL TÍTULO PROFESIONAL DE
INGENIERA DE COMPUTACIÓN Y SISTEMAS

LIMA – PERÚ

2014



Reconocimiento - No comercial - Sin obra derivada
CC BY-NC-ND

El autor sólo permite que se pueda descargar esta obra y compartirla con otras personas, siempre que se reconozca su autoría, pero no se puede cambiar de ninguna manera ni se puede utilizar comercialmente.

<http://creativecommons.org/licenses/by-nc-nd/4.0/>



USMP
UNIVERSIDAD DE
SAN MARTÍN DE PORRES

**FACULTAD DE
INGENIERÍA Y ARQUITECTURA**

**ESCUELA PROFESIONAL DE INGENIERÍA DE COMPUTACIÓN Y
SISTEMAS**

**AGENTE CIUDADANO – APLICACIÓN MÓVIL PARA
REPORTAR LA UBICACIÓN DE VEHÍCULOS ROBADOS**

TESIS

**PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO DE
COMPUTACIÓN Y SISTEMAS**

PRESENTADA POR

ACUÑA NIÑO, SOFÍA GIULIANA

BARBA QUEZADA, SILVIA ELENA

LIMA – PERÚ

2014



Dedico esta tesis a mis padres, quienes me dieron vida, educación, apoyo y consejo. A mis compañeros de estudios, a mis profesores y amigos. A mi novio Jonathan quien me apoyo y alentó para continuar y no desistir.
Para todos ellos hago esta dedicatoria.



Dedicatoria SILVIA

RESUMEN

El presente documento está basado en el desarrollo de una aplicación móvil cuyo objetivo es proveer a la ciudadanía una herramienta que permita validar si un vehículo sospecho es robado; es así como nace “Agente Ciudadano”, aplicación que permitirá involucrar a la ciudadanía en la recuperación de un vehículo robado. El “Agente Ciudadano” permitirá consultar el estado de un vehículo ingresando la placa, informar la ubicación específica de manera discreta de un vehículo robado, comunicar de manera extraoficial el robo de un vehículo, visualizar reportes de los datos ingresados a través de la página web; información que podría encontrarse a disposición de la Policía Nacional del Perú (PNP).

La metodología usada en el presente proyecto es una metodología basada en SCRUM, la cual permite realizar entregas parciales en cortos periodos de tiempo. Los requerimientos identificados serán plasmados en el Product Backlog y divididos en 3 Sprints. El primer Sprint tendrá como entregable el desarrollo del registro de usuarios y comunicar robos; en el segundo Sprint entregará el desarrollo de la consulta de placas y los datos obtenidos de la ubicación a través del GPS y en el tercer Sprint se desarrollará una página web que mostrará reportes de los datos ingresados por los agentes.

Palabras Claves: Aplicación Móvil, Vehículo, Robo, Agente Ciudadano, Ciudadanía

ABSTRACT

This document is based on the development of a mobile application to provide citizens a tool to validate the suspicion of a vehicle possibly stolen; that's how the application "Agente Ciudadano" was created in order to involve citizens in the recovery of a stolen vehicles. The "Agente Ciudadano" allows requesting information of status of a vehicle entering the plate, discreetly record relevant data of a stolen vehicle or suspect, this information will be available to the Policía Nacional del Perú (PNP), register specific characteristics of the vehicle and receive reports periodically.

The methodology used in this project is based on a methodology SCRUM, which allows partial deliveries in short periods of time. The requirements identified will be reflected in the Product Backlog and Sprints divided into 3. The first Sprint will release the development of User registration and notification in a web page, in the second Sprint will be giving the query development boards and the data obtained from the location through GPS, in the third Sprint will present the development reports from the data entered by the agents.

Keywords: Mobile Application, Vehicle, Stolen Car, Agente Ciudadano, Citizenship.

ÍNDICE GENERAL

ÍNDICE DE TABLAS	x
ÍNDICE DE FIGURAS	xii
INTRODUCCIÓN	xiv
Situación Problemática	xv
Objetivos de la Investigación	xv
Objetivo General	xv
Objetivos Específicos	xvi
Justificación	xvi
Justificación Práctica	xvii
Justificación Social	xvii
Justificación Financiera	xvii
CAPÍTULO I. MARCO TEÓRICO	
1.1 Antecedentes	1
1.1.1 Delincuencia en Lima	
1.1.2 Robo de Vehículos	2
1.1.3 Aplicaciones Móviles para la Seguridad Ciudadana	4
1.1.4 Casos de éxito - Aplicaciones para Seguridad Ciudadana en Lima	6
1.1.4.1 Alerta Surco	
1.1.4.2 Bellavista Emergencia	8
1.1.4.3 JM Emergencias	10
1.1.4.4 Lince Apps	11
1.1.4.5 Miraflores en tu corazón	13
1.1.4.6 SOS Alto Al crimen	15
	vi

1.1.4.7	Vecino Vigilante	17
1.1.5	Cuadro Comparativo Aplicaciones brindadas por las Municipalidades de Lima y Callao	20
1.2	Bases Teóricas	21
1.2.1	Android Studio	
1.2.2	Administrador de Base de Datos MySQL	
1.2.3	Apache Tomcat	23
1.2.4	Global Positioning System – GPS	24
1.2.5	Metodologías	26
1.2.5.1	RUP (Rational Unified Procces)	
1.2.5.2	SCRUM	29
1.2.5.3	XP - Extreme Programing	34
1.2.5.3.1	HERRAMIENTAS DE LA XP	36
1.2.6	Netbeans	38
1.2.7	Sistemas Operativos para Dispositivos Móviles	39
1.2.7.1	Android	
1.2.7.2	Blackberry Os	45
1.2.7.3	IOS	47
1.2.8	SQLite	50
1.2.9	Telefonía Móvil	52
1.3	Definición términos básicos	53
 CAPÍTULO II.		
 METODOLOGÍA		
2.1	Materiales	58

2.2	Métodos	59
2.2.1	Requerimientos	
2.3	Desarrollo del Proyecto	74
2.3.1	SPRINT 1: Registro de Usuario y Comunicar Robo	
2.3.2	SPRINT 2: Placa y GPS	76
2.3.3	SPRINT 3: Aplicación web	78
2.3.4	Diseño de Tablas	80
2.3.5	Diccionario de tablas	81
2.3.6	Diccionario de datos	82
2.3.7	Arquitectura del sistema	87
CAPÍTULO III.		
PRUEBAS Y RESULTADOS		88
3.1	Casos de Pruebas	
3.2	Resultado de pruebas	89
CAPÍTULO IV.		
DISCUSIÓN Y APLICACIONES		91
4.1	Diferenciación de la aplicación Agente Ciudadano	91
4.2	Elección del sistema operativo	93
CONCLUSIONES		94
RECOMENDACIONES		95
ANEXOS		96
FUENTES DE INFORMACIÓN		112

ÍNDICE DE TABLAS

Tabla N° 1 – Cálculo de ingreso por concepto de publicidad – Móvil	xviii
Tabla N° 2 - Cálculo de ingreso por concepto de publicidad – Página Web	
Tabla N° 3 - Flujo de Caja Económico	
Tabla N° 4 – Evaluación del proyecto	xix
Tabla N° 5 - Cuadro Comparativo de Funcionalidades Generales de las aplicaciones de Seguridad en las Municipalidades de Lima y Callao.....	20
Tabla N° 6 - Actividades SCRUM	34
Tabla N° 7 - Cuadro Comparativo Metodologías	37
Tabla N° 8 - Características de Android.....	41
Tabla N° 9 - Versiones de ANDROID	44
Tabla N° 10 – Capa IOS	49
Tabla N° 11 - Product Backlog.....	61
Tabla N° 12 - Requerimientos Funcionales	64
Tabla N° 13 - SCRUM –Etapas	66
Tabla N° 14 - Roles y Responsabilidades.....	70
Tabla N° 15 - SPRINT 1: Requerimientos Funcionales	76
Tabla N° 16 - SPRINT 2: Requerimientos Funcionales	78
Tabla N° 17 - SPRINT 3: Requerimientos Funcionales	79
Tabla N° 18 - Tablas Agente Ciudadano	81
Tabla N° 19 - Tabla Agente	83
Tabla N° 20 - Tabla Color	
Tabla N° 21 - Tabla Distrito.....	
Tabla N° 22 - Tabla Marca	84
Tabla N° 23 - Tabla Modelo	
Tabla N° 24 - Tabla RoboReportado.....	85
Tabla N° 25 - Tabla TipoVehiculo	
Tabla N° 26 - Tabla Vehículo	86

Tabla N° 27 - Tabla VehiculoRobado	87
Tabla N° 28 – Listado de Casos de Prueba	89
Tabla N° 29 – Comparaciones y semejanzas entre otras aplicaciones y el Agente Ciudadano	92
Tabla N° 30 – Detalle de instalación de aplicaciones móviles	97
Tabla N° 31 – Cuadro de Gastos y Pérdidas	98
Tabla N° 32 – Prototipos SPRINT 1	99
Tabla N° 33 – Prototipos SPRINT 2	100
Tabla N° 34 – Prototipos SPRINT 3	102
Tabla N° 35 – CP1: Aceptar Términos y Condiciones	
Tabla N° 36 – CP2: Registrar Agente	103
Tabla N° 37 – CP3: Comunicar Robo	
Tabla N° 38 – CP4: Actualizar Datos	104
Tabla N° 39 – CP5: Consultar Placa	
Tabla N° 40 – CP6: Reportar Vehículo	105
Tabla N° 41 – CP7: Ingresar a la Página Web	
Tabla N° 42 – CP8: Consultar Reporte	106
Tabla N° 43 – CP9: Exportar Reporte	

ÍNDICE DE FIGURAS

Figura N° 1 - Sistemas Operativos Móviles Más Usados en Perú	xvi
Figura N° 2 - Cuadro % de Responsabilidad Ante la Inseguridad Ciudadana	2
Figura N° 3 - Estadística de Seguridad Ciudadana	
Figura N° 4 - Cuadro Comparativo 2013	4
Figura N° 5 – Aplicación Alerta Surco	7
Figura N° 6 - Aplicación Bellavista Emergencias	9
Figura N° 7 - Aplicación JM Emergencias	10
Figura N° 8 - Aplicación Lince Apps	12
Figura N° 9 - Aplicación Miraflores En Tu Corazón	14
Figura N° 10 - Aplicación S.O.S Alto al Crimen	16
Figura N° 11 - Aplicación Vecino Vigilante	18
Figura N° 12 - Arquitectura Apache Tomcat	24
Figura N° 13 - Como Funciona GPS	26
Figura N° 14 - Fases de Ciclo de Vida RUP	28
Figura N° 15 - Proceso Scrum	31
Figura N° 16 - Arquitectura Android	42
Figura N° 17 - Versiones de ANDROID	44
Figura N° 18 - Arquitectura BlackBerry OS	45
Figura N° 19 - Arquitectura de iOS	48
Figura N° 20 - Planificación de la Iteración	67
Figura N° 21 - Modelo de Datos	80
Figura N° 22 – Arquitectura del Sistema	87
Figura N° 23 –Pruebas: Muestra 1	90
Figura N° 24 –Pruebas: Muestra 2	90
Figura N° 25 –Promedio de Instalaciones de aplicaciones móviles realizadas a noviembre 2014	97
Figura N° 26 – Resultado de Pruebas: Portada, Términos y Condiciones	107
Figura N° 27 – Resultado de Pruebas: Registrar Agente y Bienvenida	

Figura N° 28 –Resultado de Pruebas: Menú Principal y Comunicar Robo	108
Figura N° 29 – Resultado de Pruebas: Placa y GPS	
Figura N° 30 – Resultado de Pruebas: Reportar Ubicación y Mensajes	109
Figura N° 31 –Resultado de Pruebas: Página Web y Reportes	111



INTRODUCCIÓN

En los últimos años, el parque automotriz de Lima Metropolitana ha venido incrementándose de manera exponencial y con ello también el número de robos de vehículos, cuya problemática se ve reflejada casi a diario en los noticieros y periódicos nacionales (DIROVE, s/f; Arroyo, 2013). La Dirección de Investigación de Robo de Vehículos – DIROVE (s/f) publicó un informe con cifras oficiales de vehículos robados y recuperados, desde el año 2008 al año 2013, el cual nos muestra que el porcentaje de robo de vehículos viene incrementándose, mientras que el porcentaje de recuperación de los vehículos continua disminuyendo. Dicho informe nos permite concluir que la Policía Nacional del Perú – DIROVE, requiere de mayor apoyo para poder hacerle frente a la delincuencia. Arroyo (2013) comentó que el robo de vehículos ha aumentado en un 5%; y además brindó datos estadísticos relevantes sobre la cantidad de robo de vehículos en el 2012, solo en la capital; así como la cantidad de vehículos que fueron recuperados por la División de Robo de Vehículos (DIROVE) de la Policía Nacional del Perú (segundo y tercer párrafo del artículo).

Actualmente, la Policía Nacional del Perú viene realizando una ardua labor para luchar contra la ola delincencial en nuestra ciudad, pero esta no se da abasto, por lo que consideramos que sería de gran aporte contar con la participación activa de la ciudadanía; ya que es la ciudadanía la directamente afectada por este problema social. Sin embargo, la inseguridad y el miedo que siente la ciudadanía impiden que se involucre en apoyar a la policía para mitigar este problema. Uno de los principales motivos es el hecho de verse involucrados en una situación de peligro, lo cual deja en su mayoría deja un sentimiento de impotencia para aquellos ciudadanos que si quisieran colaborar con la causa (DIROVE, s/f; Arroyo, 2013).

Debido a ello, se brinda como solución la aplicación móvil “Agente Ciudadano”, que permitirá crear una comunidad de usuarios que informará a

los ciudadano sobre los últimos robos y a su vez contribuya con información actualizada sobre la ubicación de vehículos robados, los cuales estarían a disposición de la Policía Nacional del Perú (PNP), con la finalidad de que esta pueda contar con mayor información para recuperar dichos vehículos (DIROVE, s/f; Arroyo, 2013).

Situación Problemática

En base a lo indicado por Espinoza, Fernández y El Comercio (2014), Martínez, Ramírez y Rodríguez (2013), existen pocas aplicaciones móviles accesibles y seguras que permitan a la ciudadanía brindar información, en caso cuenten con información de algún vehículo robado o sospechoso. Actualmente, el ciudadano puede avisar de estos acontecimientos de manera personal, vía telefónica o por aplicaciones móviles a la Policía Nacional del Perú. De optar por esta última opción, son directamente dirigidos a una central telefónica, lo que podría conseguir que el ciudadano desista en comunicarse, por el miedo de verse involucrados.

Por lo expuesto, se ha identificado que el problema es que la ciudadanía cuenta con pocos medios donde puedan visualizar de manera rápida si un vehículo sospechoso es robado, e informar la ubicación del vehículo sin comprometer su integridad personal.

Objetivos de la Investigación

Objetivo General

Desarrollar una aplicación móvil que permita a los ciudadanos de Lima Metropolitana, consultar si un vehículo es robado, informar su ubicación e informar el robo de sus vehículos.

Objetivos Específicos

- Brindar una aplicación móvil intuitiva, que no comprometa la integridad personal del usuario.
- Desarrollar una aplicación móvil donde la ciudadanía pueda registrar ubicaciones de vehículos robados y robos de sus vehículos; los cuales estarían a disposición de la Policía Nacional del Perú.
- Desarrollar una página web que permita centralizar e informar sobre el robo de vehículos por distrito.

Justificación

En esta sección se describirá las justificaciones prácticas, sociales y financieras para el desarrollo de la tesis. En base a lo obtenido de Espinoza, Fernandez. Lima Como Vamos (2014) y teniendo en cuenta las siguientes estadísticas:

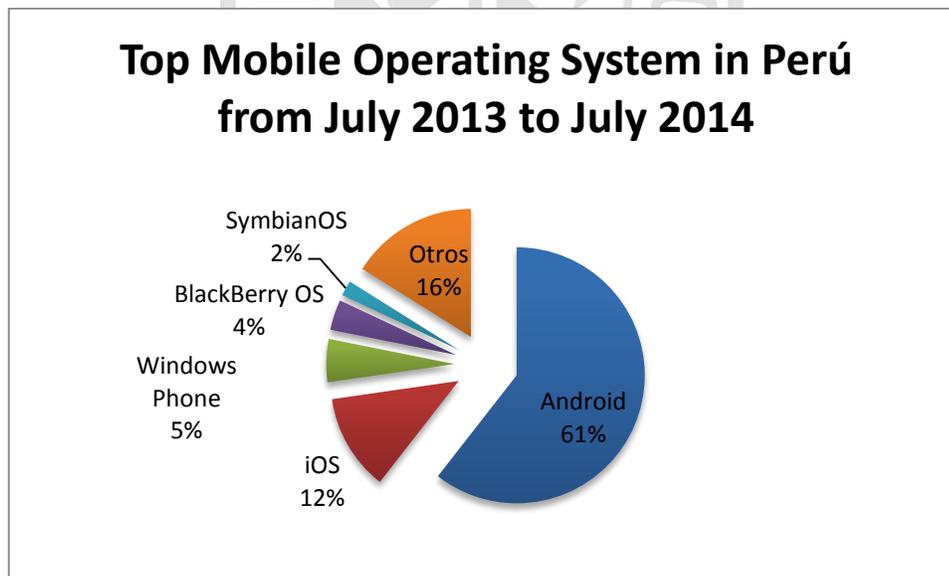


Figura N° 1 - Sistemas Operativos Móviles Más Usados en Perú
Fuente: StatCounter (s/f)

Presentamos las siguientes justificaciones:

Justificación Práctica

En base a lo indicado por Espinoza, Fernández y El Comercio (2014), Martínez, Ramírez y Rodríguez (2013). Desde el punto de vista práctico se propone implementar una aplicación móvil que permita a los ciudadanos brindar información de manera segura y sencilla la ubicación de vehículos robados; para con ello involucrar a la ciudadanía y así contar con más agentes dispuestos a contribuir contra la lucha delincriminal.

Justificación Social

En base a lo indicado por Espinoza, Fernández y El Comercio (2014), Martínez, Ramírez y Rodríguez (2013). Desde el punto de vista social, esta herramienta es inclusiva, permitiendo a la ciudadanía involucrarse y sentirse parte de la lucha contra la delincuencia. Se desea que en un futuro cercano existan más medios como el “Agente Ciudadano” que permitan afianzar la relación entre los ciudadanos y la Policía Nacional del Perú.

Justificación Financiera

Actualmente, el costo mensual por concepto de publicidad para aplicaciones y páginas web que tienen una audiencia potencial de 5,000 a 10,000 personas es de 300 soles.

Ver Anexo 1 – Promedio de Instalaciones de aplicaciones móviles realizadas a noviembre 2014, donde muestra la cantidad promedio de descargas de aplicaciones móviles ya existentes, para la cual se ha basado la estimación de la audiencia del proyecto Agente Ciudadano en su primera fase.

Se ha identificado que la aplicación móvil Agente Ciudadano cuenta con 3 espacios habilitados para publicidad y la página web Agente Ciudadano cuenta con 5 espacios habilitados.

Publicidad Móvil		
Costo Mensual:	300	S/.
Costo Anual	3,600	S/.
Espacio Publicitario Movil (3):	10,800	S/.

Tabla N° 1 – Cálculo de ingreso por concepto de publicidad – Móvil
Fuente: Elaboración propia

Publicidad Página Web		
Costo Mensual:	300	S/.
Costo Anual	3,600	S/.
Espacio Publicitario Web (5):	18,000	S/.

Tabla N° 2 - Cálculo de ingreso por concepto de publicidad – Página Web
Fuente: Elaboración propia

Tomando en cuenta los ingresos por publicidad y los egresos para el desarrollo de la aplicación móvil y la página web del Agente ciudadano el flujo de caja económica anual sería el siguiente:

AGENTE CIUDADANO
Flujo de Caja
Expresado en S/.

	Año 0	Año 1	Año 2	Año 3
INGRESOS				
Publicidad Movil		10,800	12,960	16,848
Publicidad Web		18,000	21,600	25,920
EGRESOS				
Inversión				
- Libros,Licencias	-2,000	-500	-500	-500
- Equipos de Computo	-4,000			
Costos de Operación				
- Gastos Operativos		-4,320	-5,184	-6,415
- Gastos Administrativos		-1,440	-1,728	-2,138
Impuesto a la Renta		-6,943	-8,464	-10,631
FC ECONÓMICO	-6,000	15,597	18,684	23,084
Tasa de Crecimiento de Ingresos		20%		
Tasa de gastos operativos		15%		
Tasa de gastos administrativos		5%		
Depreciación		25%		
Impuesto a la Renta		30%		
Tasa de Descuento		3%		

Tabla N° 3 - Flujo de Caja Económico
Fuente: Elaboración propia

Ver Anexo 2 –Estado de Ganancias y pérdidas

Los resultados de la evaluación del proyecto indica:

TIR	272%
VAN	48,435.70
Ratio de Costo/Beneficio	9.07

Tabla N° 4 – Evaluación del proyecto

Fuente: Elaboración propia

Luego de haber realizado el análisis financiero del proyecto para la implementación del Agente Ciudadano, se puede determinar que el proyecto es rentable. Los datos nos indican que la tasa de inversión de retorno (TIR) es de 272% por lo cual el proyecto es aceptado. Además, se obtiene que el Valor Actual Neto (VAN) es de valor positivo, por lo cual el proyecto también es aceptado y que el periodo de recuperación de la inversión sería menor de un año. La principal fuente de ingresos es por los costos de la publicidad, los cuales irán incrementándose en un 20% mensual debido a que se espera que la aplicación móvil Agente Ciudadano alcance mayor acogida en la ciudadanía.

CAPÍTULO I. MARCO TEÓRICO

1.1 Antecedentes

1.1.1 Delincuencia en Lima

El incremento de la delincuencia en los últimos años, ha generado la sensación de inseguridad en la ciudadanía. A diario existen más víctimas de la delincuencia, testigos de robos o se informa de las víctimas en los medios de comunicación. (Rodríguez, 2013)

Rodríguez (2013) comentó que “en el Perú el 84.9% de limeños cree que será víctima de un delito, Sin embargo, la victimización real alcanza solo al 40,1%” (parr.1). En el informe de la organización Lima Como Vamos del Observatorio Ciudadano Independiente (2013), se concluye una percepción alta de inseguridad en la capital, ubicándola en la segunda ciudad de Sudamérica con el mayor índice de percepción de inseguridad ciudadana y la segunda en Latinoamérica (superada por México D.F.). Además, brinda datos sobre la responsabilidad de las diferentes instituciones ante la lucha contra la inseguridad ciudadana, atribuida por los ciudadanos. Dichos datos se muestran a mayor detalle en el siguiente cuadro:



Figura N° 2 - Cuadro % de Responsabilidad Ante la Inseguridad Ciudadana
Fuente: Lima Cómo Vamos (2014).

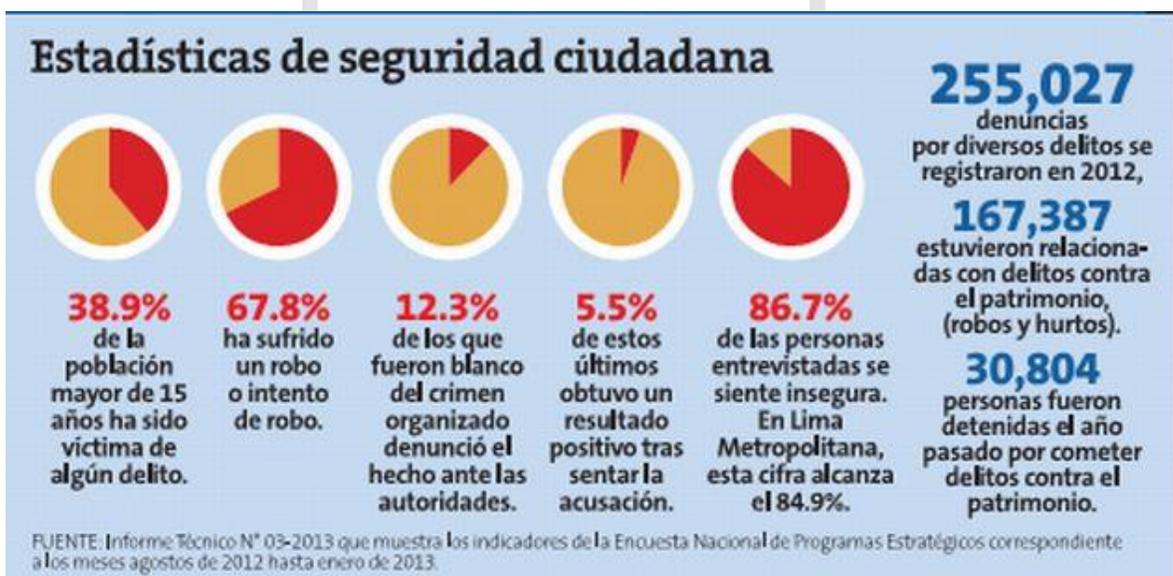


Figura N° 3 - Estadística de Seguridad Ciudadana
Fuente: Martínez (2013).

1.1.2 Robo de Vehículos

El robo de vehículos es una de los mayores problemas que vive la comunidad en los últimos años, éste problema sigue incrementando de tal forma que las estadísticas se pueden medir en horas.

León (2014) comentó:

Cada tres horas los delincuentes roban dos autos, de acuerdo a información de la División de Investigación de Robo de Vehículos DIROVE. Esta preocupante cifra se desprende de los 1.734 vehículos que han sido reportados como robados en lo que va del año y al día se pierden unos 16. Solo se han logrado recuperar hasta la fecha unos 1.051 automóviles, informó esa unidad policial. Las marcas preferidas por los 'amigos de lo ajeno' son: Hyundai, Nissan, Kia, Toyota y Volkswagen. Al día, entre 14 y 20 personas son víctimas de robo de sus vehículos en las calles de la capital. La recuperación puede darse entre 15 y 20 días o más, dependiendo de cada caso. (párr. 1).

En la página oficial de la DIROVE (s/f) se publicó un cuadro comparativo donde se puede observar la cantidad de vehículos robados desde el año 2008 al 2013, así como los trabajos realizados para la recuperación de vehículos.

■ COMPARATIVO 2013

CUADRO COMPARATIVO DE VEHICULOS ROBADOS Y RECUPERADOS CON PORCENTAJE DE RECUPERACION DE RECUPERACION DEL MES DE ENERO A JUNIO DE LOS AÑOS 2008, 2009, 2010, 2011, 2012, 2013

	2008	2009	2010	2011	2012	2013
VEHICULOS ROBADOS	3640	3818	3266	2988	3454	3674
VEHICULOS RECUPERADOS	1566	2085	2303	2495	2500	2070
% DE RECUPERACION	43%	55%	71%	84%	72%	56%



Figura N° 4 - Cuadro Comparativo 2013
Fuente: DIROVE (s/f).

1.1.3 Aplicaciones Móviles para la Seguridad Ciudadana

Algunas municipalidades han desplegado aplicaciones para la seguridad ciudadana con el fin de que la comunidad cuente con medios más accesibles para reportar incidencias, problemas y/o denuncias. Entre los medios considerables más accesibles a la comunidad se encuentran los Smartphone y las redes sociales. Al respecto, Espinoza (2014) publicó:

Según los datos de la empresa de logística inteligente Celisticis, la importación de teléfonos de alta gama pasó de 20% en el 2012 a 35% en el 2013. Es decir, ahora uno de cada tres teléfonos móviles es un celular de alta gama. Además, se estima que el porcentaje subirá hasta 40% o 45% este año. En América Latina, las ventas de teléfonos inteligentes aumentaron 31,18%. Aunque los feature phones (celulares de gama baja) aún dominan el mercado

nacional, poco a poco la balanza se inclina hacia el otro lado, ya que el consumidor móvil peruano “busca conectividad, usar las redes sociales para compartir con sus contactos lo que está haciendo, jugar en línea, etc.”, afirma Luis Jara, Country Manager de Celistics en el Perú (párr. 2 y 3).

Dado que el uso de teléfonos viene incrementado, algunos municipios han creado 'Apps' (aplicaciones) para que sus vecinos puedan informar robos, asaltos o accidentes y sean auxiliados cuanto antes por serenos, bomberos o policías. Desde el 2013, comunas como Miraflores, Pueblo Libre, Jesús María, San Borja, Surco, entre otros, han implementados estas aplicaciones móviles gratuitas, que pueden ser descargadas desde sus páginas web o las tiendas virtuales del proveedor de cada celular. Estas denuncias llegan directamente por las centrales de seguridad de cada distrito desde donde se movilizan los agentes de serenazgo. Calvo (2014) (parr.1 y 2).

Fernández (2014) comentó que:

Las aplicaciones brindadas por las municipalidades de Lima, dirigen llamadas directamente a una central que se encarga de enviar un agente, para así atender una incidencia reportada, que puede ser una sospecha de robo, un robo efectuado, semáforos malogrados, calles sucias, vehículos mal estacionados, entre otros. Al respecto, El Comercio (2014) publicó que las municipalidades de Surco, Jesús María, Miraflores y San Borja han desarrollado aplicaciones móviles como nueva estrategia para enfrentar la seguridad ciudadana y otros problemas distritales. Como informa El Comercio, cerca de 3 mil alertas llegan cada mes a través de las aplicaciones móviles instaladas en los celulares inteligentes o en las

tablets de los usuarios, a las centrales de seguridad de los citados distritos. En todos los casos, están disponibles en App Store y Google Play. Todas las descargas son gratuitas (parr.3).

La municipalidad de Miraflores (2013) publicó:

La aplicación “Miraflores en tu corazón”, en los primeros 20 días de creada, tuvo más de 2,500 descargas gratuitas. Esta innovación consiguió que, en menos de un mes, la municipalidad haya resuelto 369 reportes de incidentes relacionados principalmente con autos mal estacionados, ruidos molestos, arreglo de pistas y veredas, limpieza pública o semáforos que no funcionaban correctamente, entre otras. Los ciudadanos pueden enviar fotografías, audios y la ubicación exacta de donde detectaron el problema o incidente para que la municipalidad brinde una rápida solución. Luego, ellos recibirán inmediatamente un número de atención para monitorear su solicitud. La aplicación además ofrece alertas sobre los bloqueos temporales en las vías del distrito y muestra las actualizaciones del Twitter de la municipalidad de Miraflores y de la Central Alerta Miraflores. (párr.7).

1.1.4 Casos de éxito - Aplicaciones para Seguridad Ciudadana en Lima

1.1.4.1 Alerta Surco

La municipalidad de Surco (2014) publicó:

La aplicación móvil “Alerta Surco”, se ha convertido en una importante herramienta para atender las denuncias ante un hecho delictivo o solicitar apoyo ante una emergencia (parr.1), reforzando de esta

manera las herramientas de Seguridad Ciudadana en el distrito (parr.2). Permite informar al Centro de Control de Operaciones (CCO) sobre una emergencia, que inmediatamente será respondido a través de una llamada del personal de la municipalidad.



Figura N° 5 – Aplicación Alerta Surco
Fuente: Delgado (2013).

Delgado (2013) publicó las siguientes características:

a. Incidencias disponibles a reportar:

Libre, permite reportar cualquier tipo de incidencia originadas en el Distrito.

b. Funcionalidades disponibles:

- GPS: Al ingresar a la aplicación por primera vez solicita envía la ubicación exacta del usuario.
- DEBES REGISTRARTE: Luego de instalar la aplicación, solicita registrarse e ingresar datos

(Número de Teléfono Celular, Número de Documento, Nombres y Apellidos, Email).

- FACEBOOK: Deriva al FACEBOOK oficial de la municipalidad de Surco.
- TWITER: Deriva al TWITER oficial de la municipalidad de Surco.
- CONTACTOS: Muestra el número de la central de atención de la municipalidad de Surco.
- SERENAZGO: Permite enviar alerta a la central de atención de la municipalidad de Surco (un agente devuelve la llamada).
- AMBULANCIA: Permite enviar alerta a la central de atención de la municipalidad de Surco (un agente devuelve la llamada).
- BOMBEROS: Permite enviar alerta a la central de atención de la municipalidad de Surco (un agente devuelve la llamada).
- CONFIGURACIÓN: Permite editar los datos o reiniciar los datos (elimina la cuenta y solicita ingresar de nuevo los datos).

1.1.4.2 Bellavista Emergencia

El Diario 16 (2014) publicó:

La aplicación “Bellavista Emergencias” es la aplicación de seguridad, lanzada por la municipalidad del distrito, que puede alertar en tiempo real a la policía y serenazgo sobre actos delictivos, incendios, accidentes de tránsito y otros”. “La aplicación puede ser instalada en Smartphones o tablets que usen sistema Android y es de

descarga gratuita. Esta habilita a los vecinos para acceder a cuatro opciones: accidente de tránsito, incendio, asalto y SOS (para otro tipo de delitos).” (párr. 1 y 2)



Figura N° 6 - Aplicación Bellavista Emergencias
Fuente: Municipalidad de Bellavista (2014).

Municipalidad de Bellavista (2014) publicó las siguientes características:

a. Incidencias disponibles a reportar:

Accidentes, Incendio, Asalto y Ayuda. Cualquiera de las opciones mencionadas deriva al usuario a la Central Telefónica de la municipalidad de Bellavista.

b. Funcionalidades disponibles.

- MUNICIPALIDAD DE BELLAVISTA: Luego de instalar la aplicación, solicita registrarse e ingresar datos (DNI, Nombres y Apellidos, Celular, Correo).

- ENVÍA UNA REFERENCIA O INFORMACIÓN ADICIONAL: Envía la referencia a la central de atención de la municipalidad de Bellavista.

1.1.4.3 JM Emergencias

La municipalidad de Jesús María (2014) publicó:

JM Emergencias es una aplicación móvil desarrollada para alertar incidentes, asaltos, accidentes mediante un celular Smartphone. Esta aplicación se encuentra interconectada en tiempo real con la Central de Seguridad Ciudadana de la municipalidad de Jesús María. Al seleccionar cualquiera de las opciones envía una alerta a la Central de Seguridad Ciudadana de la municipalidad de Jesús María, donde un encargado devuelve la llamada al ciudadano (párr. 1 y 2).



Figura N° 7 - Aplicación JM Emergencias
Fuente: Rospigliosi (2014).

Rospigliosi (2014) publicó las siguientes características:

a. Incidencias disponibles a reportar:

Accidentes, Incendio, SOS y Asalto. Cualquiera de las opciones mencionadas deriva al usuario a la Central Telefónica de la municipalidad de Jesús María.

b. Funcionalidades disponibles:

- REFERENCIA O INFORMACIÓN ADICIONAL: Permite registrar información adicional que es enviada directamente a la central cuando se selecciona un tipo de incidencia.
- 1380: Al elegir esta opción realiza una llamada a línea rápida de emergencias de la municipalidad de Jesús María.
- 63-40100: Al elegir esta opción realiza una llamada a la Central de Seguridad Ciudadana de la municipalidad de Jesús María.
- GPS (NO VISIBLE): Al elegir un tipo de incidencia envía la ubicación exacta del usuario.

1.1.4.4 Lince Apps

Rospigliosi (2014) publicó:

LINCE APPS, es una aplicación desarrollada como un medio de alerta de emergencias para ambulancias, asaltos, bomberos para todos los celulares Smartphone que cuenten con sistema Android. Esta aplicación será interconectada con la central del monitoreo de la municipalidad de Lince el cual se podrá visualizar la emergencia como un punto georreferenciado en un mapa digital por el

GPS del celular utilizado y que permitirá la llegada inmediata al lugar de los hechos de policías y serenos (párr. 1 y 2).



Figura N° 8 - Aplicación Lince Apps
Fuente: Municipalidad de Lince (2014).

Municipalidad de Lince (2014) publicó las siguientes características:

a. Incidencias disponibles a reportar:

Libre, permite reportar cualquier tipo de incidencia originadas en el Distrito.

b. Funcionalidades disponibles:

- DEBE REGISTRAR SUS DATOS ANTES DE UTILIZAR LA APLICACIÓN: Luego de instalar la aplicación, solicita registrarse e ingresar datos (DNI, Nombres y Apellidos, Celular, Correo).

- BOMBEROS: Envía una alerta con la ubicación de GPS a la Policía Nacional del Perú y/o Serenazgo de la municipalidad de Lince.
- AMBULANCIAS: Envía una alerta con la ubicación de GPS a la Policía Nacional del Perú y/o Serenazgo de la municipalidad de Lince.
- ASALTOS: Envía una alerta con la ubicación de GPS a la Policía Nacional del Perú y/o Serenazgo de la municipalidad de Lince.
- AYUDA: Envía una alerta con la ubicación de GPS a la Policía Nacional del Perú y/o Serenazgo de la municipalidad de Lince.
- CENTRAL DE EMERGENCIAS: Muestra el número de la central de emergencias de la municipalidad de Lince.
- LÍNEA GRATUITA: Muestra el número de la Línea Gratuita de la municipalidad de Lince.
- GPS (NO VISIBLE): Obtiene la ubicación exacta del usuario.

1.1.4.5 Miraflores en tu corazón

La municipalidad de Miraflores (s/f) publicó:

La aplicación Miraflores en tu corazón tuvo más de 2,500 descargas gratuitas en solo 20 días. Esta innovación ya ha conseguido que, en menos de un mes, la municipalidad haya resuelto 369 reportes de incidentes relacionados principalmente con autos mal estacionados, ruidos molestos, arreglo de pistas y veredas, limpieza pública o semáforos que no funcionaban correctamente, entre otras. Los

usuarios pueden enviar – inclusive en forma anónima - fotografías, audios y la ubicación exacta de donde detectaron el problema o incidente para que la municipalidad brinde una rápida solución. Luego, ellos recibirán inmediatamente un número de atención para monitorear su solicitud. La aplicación además ofrece alertas sobre los bloqueos temporales en las vías del distrito y muestra las actualizaciones del Twitter de la municipalidad de Miraflores y de la Central Alerta Miraflores” (párr. 3).



Figura N° 9 - Aplicación Miraflores En Tu Corazón
Fuente: Municipalidad de Miraflores (2014).

Municipalidad de Miraflores (2014) publicó las siguientes características:

- a. Incidencias disponibles a reportar:

Limpieza Pública, Semáforo Inoperativo, Parques Y Jardines, Vereda/Pista Mal Estado, Ruidos Molestos, Iluminación Pública, Auto Mal Estacionado y Otras Consultas. Cualquiera de las opciones mencionadas muestra la pantalla con las funcionalidades de foto, audio y ubicación.

b. Funcionalidades disponibles:

- FOTO: Permite subir fotos de la incidencia previamente seleccionada.
- AUDIO: Permite subir un audio de la incidencia previamente seleccionada.
- UBICACIÓN: Permite registrar la ubicación exacta de la incidencia previamente seleccionada.
- TWITTER: Accede al TWITTER de la municipalidad de Miraflores.
- ANUNCIOS: Muestra los anuncios presentados por la municipalidad de Miraflores.
- MI CUENTA: Muestra la bandeja de alertas reportadas por el usuario y su estado de atención.
- TU REVISTA: Link que muestra la revista de la municipalidad de Miraflores.

1.1.4.6 SOS Alto Al crimen

La municipalidad de La Molina (2014) publicó:

Es una aplicación celular gratuita, ofrecida como contribución a la comunidad que brinda las herramientas apropiadas para elevar la participación ciudadana a través del envío de mensajes S.O.S y denuncias de hechos delictivos de forma sencilla, rápida y efectiva. Con tan solo presionar el botón de S.O.S tus contactos elegidos y el Serenazgo

recibirán tu pedido de auxilio, con tu ubicación geográfica a través de la información enviada por GPS (párr.2).



Figura N° 10 - Aplicación S.O.S Alto al Crimen
Fuente: Pele System y Lola-Tech (2014).

Pele System y Lola-Tech (2014) publicó las siguientes características:

a. Incidencias disponibles a reportar:

Venta Ilegal de Pirotécnicos, Riesgo en Defensa Civil en Locales Comerciales, Venta de Juguetes Tóxicos, Narcotráfico, Terrorismo, Secuestro, Homicidio, Pandillaje, Violencia Familiar, Estafa, Robo de Vehículos, Corrupción, Violación, Robo de Casas, Marcas - Robo a Mano Armada, Ordenamiento Vial, Envía Placa y Otros.

b. Funcionalidades disponibles:

- REGISTRARSE: Luego de instalar la aplicación, solicita registrarse e ingresar datos (Número de

teléfono, Email (solicita activar la cuenta), Fecha de Nacimiento, Sexo, Términos y condiciones - solicita aceptar los términos de la aplicación).

- GPS: Al ingresar a la aplicación por primera vez solicita envía la ubicación exacta del usuario.
- DESCRIPCIÓN: Permite registrar el detalle de la incidencia seleccionada.
- UBICACIÓN: Muestra la ubicación detectada por el GPS.
- FECHA Y HORA: Muestra la fecha y hora de registro.
- FOTOS: Permite adjuntar fotos.
- CONTACTOS: Permite agregar contactos en la aplicación.
- MENSAJES: Mensajes registrados por los creadores de la aplicación.
- ENVIAR SOS: Envía alerta al Serenazgo y a tus contactos.

1.1.4.7 Vecino Vigilante

La municipalidad de Jesús María (s/f) publicó:

Es una aplicación móvil desarrollada con la finalidad de que los vecinos puedan enviar información de los diversos incidentes que puedan ocurrir en el distrito. Así también podrán incluir fotos con ubicación mediante el GPS, sobre lo siguiente: Tránsito, Medio Ambiente, Semáforos y autos mal estacionados, Ruidos molestos, Pistas y veredas, Alumbrado Público, Parque y Jardines, Robos ocurridos en el distrito que no ha sido reportados (párr.1)



Figura N° 11 - Aplicación Vecino Vigilante
Fuente: Municipalidad de Jesús María (2014).

Municipalidad de Jesús María (2014) publicó las siguientes características:

a. Incidencias disponibles a reportar:

Pista y Veredas, Ruido Molesto, Parques y Jardines, Alumbrado, Tránsito y Semáforos, Limpieza Pública, Otros y Robos. Cualquiera de las opciones permite enviar una foto y/o registrar datos adicionales.

b. Funcionalidades disponibles:

- Directorio: Lista los teléfonos y/o ubicación de los servicios brindados por la municipalidad (Casa del Adulto Mayor, Centro Cultural Servulo Gutierrez, Centro Municipal de Justicia, Local Cuba, Palacio Municipal, Casa del Vecino, Casa Jesús María y José, Casa del Vecino, Concha Acústica del campo de Marte, Policlínico Municipal, Casa de la Juventud, Gerencia de Seguridad Ciudadana, Gerencia de

Medio Emergencia, Consultoría Médico Veterinario, Vivero Municipal, Polideportivo Micky Rospigliosi, Estadio Municipal).

- MENSAJES: Mensajes enviados por la municipalidad de Jesús María.
- MIS DATOS: Luego de instalar la aplicación, solicita registrarse e ingresar datos (DNI, Nombre, Apellido, Correo, Dirección, Distrito, Tel. Móvil, Tel. Fijo).
- CENTRAL SERENAZNO 63-40100: Realiza una llamada directa a la central de Serenazgo de la municipalidad de Jesús María.
- 63-40100: Al elegir esta opción realiza una llamada a la Central de Seguridad Ciudadana de la municipalidad de Jesús María.
- GPS (NO VISIBLE): Al elegir un tipo de incidencia envía la ubicación exacta del usuario.
- FOTO: Permite subir enviar una foto del reporte.
- UBICACIÓN Y DATOS ADICIONALES DE SU REPORTE: Permite registrar datos adicionales del reporte.

1.1.5 Cuadro Comparativo Aplicaciones brindadas por las Municipalidades de Lima y Callao

El siguiente cuadro muestra un resumen de las funcionalidades generales de las aplicaciones previamente analizadas:

APLICACIONES	FUNCIONALIDADES GENERALES				DISTRITO
	Central Telefónica	Foto	Audio	GPS	
ALERTA SURCO	SI	NO	NO	SI	Surco
BELLAVISTA EMERGENCIAS	SI	NO	NO	NO	Bellavista
JM EMERGENCIAS	SI	NO	NO	NO	Jesús María
LINCE APP	NO	NO	NO	SI	Lince
MIRAFLORES EN TU CORAZÓN	NO	SI	SI	SI	Miraflores
SOS ALTO AL CRIMEN	NO	SI	NO	SI	La Molina
VECINO VIGILANTE	SI	SI	NO	SI	Jesús María

Tabla N° 5 - Cuadro Comparativo de Funcionalidades Generales de las aplicaciones de Seguridad en las Municipalidades de Lima y Callao
Fuente: Municipalidad de Bellavista, Municipalidad de Lince, Municipalidad de Jesús María, Municipalidad de Miraflores, Municipalidad de Santiago de Surco y Municipalidad de La Molina (2014).

El cuadro comparativo muestra que la principal funcionalidad en la mayoría de las aplicaciones existentes desarrolladas por las municipalidades es el comunicar al ciudadano directamente con una Central Telefónica para brindar asistencia al usuario o brindar la opción de enviar fotos, audios y ubicación a la Central Telefónica para reportar un incidente. En algunos otros casos, las aplicaciones están orientadas a la gestión municipal, como son el arreglo de calles, veredas, limpieza y mantenimiento de zonas verdes.

1.2 Bases Teóricas

1.2.1 Android Studio

Android Developers (s.f.) publicó que Android Studio es un nuevo entorno de desarrollo integrado para el sistema operativo Android lanzado por Google, diseñado para ofrecer nuevas herramientas para el desarrollo de aplicaciones y alternativa al entorno Eclipse.

Android Studio permite ver los cambios visuales que realizamos en nuestra aplicación en tiempo real, pudiendo además ver cómo se visualiza en diferentes dispositivos Android con distintas configuraciones y resoluciones de forma simultánea.

Otra de las características de Android Studio son sus nuevas herramientas de empaquetado y etiquetado de código para organizarnos al implementar grandes cantidades de código, sirviéndose además de un sistema drag & drop para mover los componentes a través de la interfaz de usuario.

Además, este nuevo entorno cuenta con Google Cloud Messaging, una funcionalidad con la que podremos enviar datos desde el servidor a terminales Android a través de la nube, siendo esta una forma de enviar notificaciones Push a nuestras apps.

Android Studio ofrece:

- Un entorno de desarrollo claro y robusto.
- Facilidad para testear el funcionamiento en otros tipos de dispositivos.
- Asistentes y plantillas para los elementos comunes de programación en Android.
- Un completo editor con muchas herramientas extra para agilizar el desarrollo de nuestras aplicaciones.

1.2.2 Administrador de Base de Datos MySQL

Cobol, Gómez, Pérez y Rocha (2005) (p.339) indicaron:

MySQL es un sistema de administración de bases de datos relacionales rápido, sólido y flexible. Es ideal para crear bases de datos con acceso desde página web dinámicas, para la creación de sistemas de transacciones *on-line* o para cualquier otra solución profesional que implique almacenar datos, teniendo la posibilidad de realizar múltiples y rápidas consultas.

MySQL ofrece varias ventajas respecto a otros sistemas gestores de bases de datos:

- Tiene licencia pública, permitiendo no solo la utilización del programa sino también la consulta y modificación de su código fuente. Resulta por tanto fácil de personalizar y adaptar a las necesidades concretas.
- El programa está desarrollado en C y C++, lo que facilita su integración en otras aplicaciones desarrolladas igualmente en esos lenguajes.
- MySQL utiliza el lenguaje SQL (Lenguaje de Consulta Estructurado) que es el lenguaje de consulta más usado y estandarizado para acceder a base de datos relacionales.
- Es un sistema cliente/servidor, permitiendo trabajar como servidor multiusuario y de subprocesamiento múltiple, es decir, cada vez que se establece una conexión con el servidor, el programa servidor crea un subproceso para manejar la solicitud del cliente, controlando el acceso simultáneo de un gran número de usuarios a los datos y asegurando el acceso solo a usuarios autorizados.
- Es portable, es decir, puede ser llevado a cualquier plataforma informática. MySQL está disponible en más de veinte

plataformas diferentes incluyendo las distribuciones más usadas de Linux, Mac X, Unix y Windows.

1.2.3 Apache Tomcat

Apache (s/f) indicó:

Apache Tomcat es una implementación de software de código abierto de las tecnologías Java Servlets y JavaServer Pages. Las especificaciones Java Servlet y JavaServer Pages son desarrollados bajo la Java Community Process. Apache Tomcat se desarrolla en un entorno abierto y participativo y liberado bajo la licencia Apache versión 2 de Apache Tomcat pretender ser una colaboración de los mejores desarrolladores de su clase en todo el mundo.

a. Arquitectura

Pavón (2012) indica:

- Servidor HTTP
- Contenedor de Servlets
 - Ejecuta Servlets
 - Convierte páginas JPS y JSF en servlets
- Arquitectura jerárquica y modular

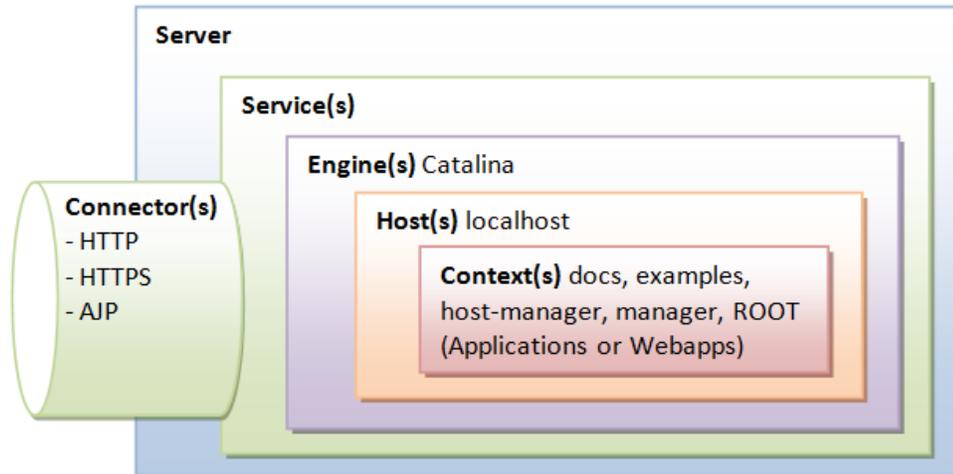


Figura N° 12 - Arquitectura Apache Tomcat
Fuente: Pavón (2012).

b. Características:

La Universidad de Ciencias y Administración - UCAD (s/f) indicó:

- Apache es uno de los mejores servidores de Webs utilizados en la red internet desde hace mucho tiempo.
- Es un servidor de web flexible, rápido y eficiente, continuamente actualizado y adaptado a los nuevos protocolos (HTTP 1.1). Implementa los últimos protocolos, aunque se base en el HTTP / 1.1.
- Puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo y con la API de programación de módulos.
- Incentiva la realimentación de los usuarios, obteniendo nuevas ideas, informes de fallos y parches para solicitudión de los mismos.

1.2.4 Global Positioning System – GPS

Global Positioning System (GPS, s/f) explicó:

Es un sistema de radionavegación de los Estados Unidos de América, basado en el espacio, que proporciona servicios fiables de posicionamiento, navegación, y cronometría gratuita e ininterrumpidamente a usuarios civiles en todo el mundo. A todo el que cuente con un receptor del GPS, el sistema le proporcionará su localización y la hora exacta en cualesquiera condiciones atmosféricas, de día o de noche, en cualquier lugar del mundo y sin límite al número de usuarios simultáneos. El GPS se compone de tres elementos: los satélites en órbita alrededor de la Tierra, las estaciones terrestres de seguimiento y control, y los receptores del GPS propiedad de los usuarios. Desde el espacio, los satélites del GPS transmiten señales que reciben e identifican los receptores del GPS; ellos, a su vez, proporcionan por separado sus coordenadas tridimensionales de latitud, longitud y altitud, así como la hora local precisa (párr.2).

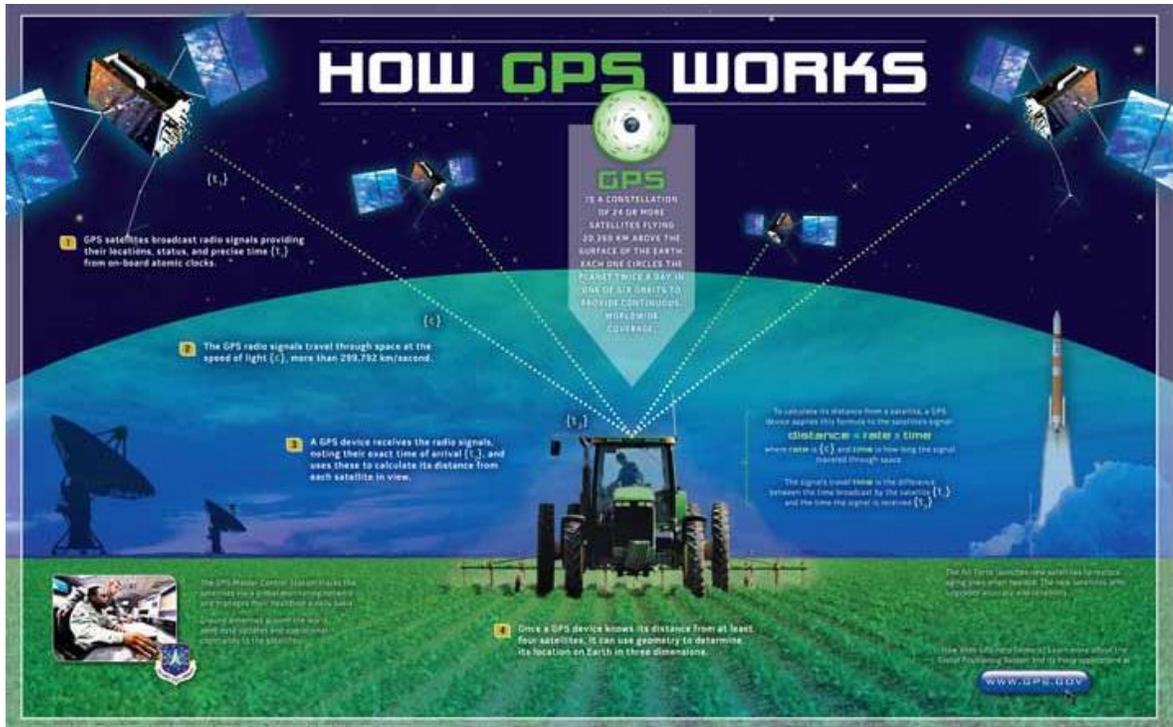


Figura N° 13 - Como Funciona GPS
Fuente: Gobierno de los Estados Unidos (s/f).

1.2.5 Metodologías

1.2.5.1 RUP (Rational Unified Procces)

IBM (s/f) explicó que:

Es una plataforma de proceso de desarrollo de software configurable que entrega mejores prácticas comprobadas y una arquitectura configurable. Permite seleccionar y desplegar solamente los componentes de proceso que se necesita para cada etapa del proyecto.

a. Características:

- Dirigido por Casos de Uso: –Los casos de uso son los artefactos primarios para establecer el comportamiento deseado del sistema

- Centrado en la Arquitectura: –La arquitectura es utilizada para conceptualizar, construir, administrar y evolucionar el sistema en desarrollo
- Iterativo e Incremental:
 - Maneja una serie de entregas ejecutables.
 - Integra continuamente la arquitectura para producir nuevas versiones mejoradas.
- Conceptualmente amplio y diverso.
- Enfoque orientado a objetos.
- En evolución continua.
- Adaptable.
- Repetible.
- Permite mediciones: Estimación de costos y tiempo, nivel de avance, etc.

La plataforma RUP incluye:

- Herramientas para configurar RUP para las necesidades específicas de su proyecto.
- Herramientas para transformar sus propios conocimientos internos en componentes del proceso.
- Herramientas potentes y personalizables de despliegue basadas en Web.
- Una comunidad online para intercambio de mejores prácticas con pares y líderes del mercado.

b. Fases del Ciclo de vida:

- Inicio: Se define la visión, alcance del sistema, se identifica los principales, autores, riesgos y costos.
- Elaboración: Identificación de riesgos, dominio del problema, análisis y arquitectura.

- Construcción: Construcción del software, puede ser dividido en iteraciones.
- Transición: Transición del desarrollo a producción.

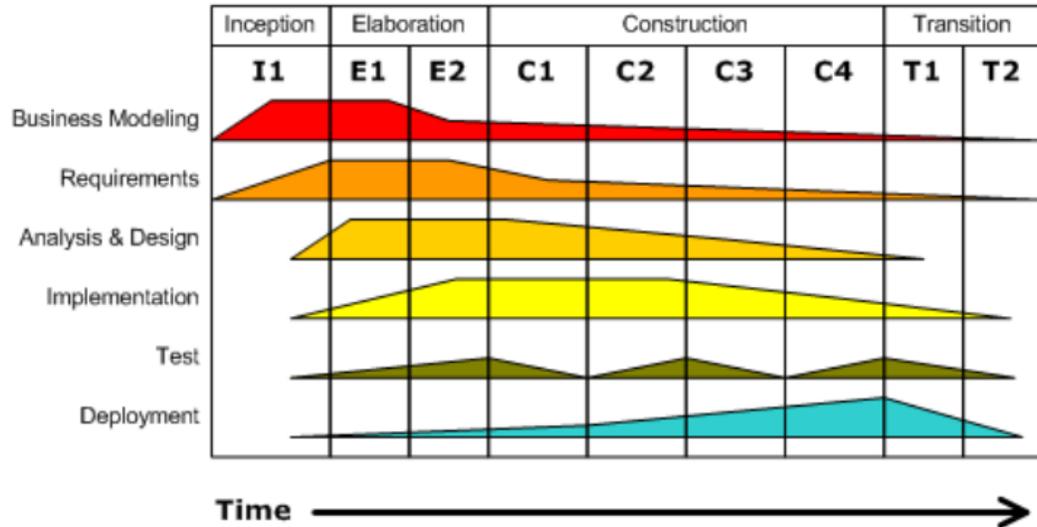


Figura N° 14 - Fases de Ciclo de Vida RUP
Fuente: IBM (s/f).

c. Principios de Desarrollo

El RUP está basado en 5 principios clave que son:

- Adaptar el proceso: El proceso deberá adaptarse a las características propias del proyecto u organización. El tamaño del mismo, así como su tipo o las regulaciones que lo condicionen, influirán en su diseño específico. También se deberá tener en cuenta el alcance del proyecto.
- Balancear prioridades: Los requerimientos de los diversos inversores pueden ser diferentes, contradictorios o disputarse recursos limitados. Debe encontrarse un balance que satisfaga los deseos de todos.

- Demostrar valor iterativamente: Los proyectos se entregan, aunque sea de un modo interno, en etapas iteradas. En cada iteración se analiza la opinión de los inversores, la estabilidad y calidad del producto, y se refina la dirección del proyecto así como también los riesgos involucrados.
- Elevar el nivel de abstracción: Este principio dominante motiva el uso de conceptos reutilizables tales como patrón del software, lenguajes 4GL o esquemas (frameworks) por nombrar algunos. Esto previene a los ingenieros de software ir directamente de los requisitos a la codificación de software a la medida del cliente. Un nivel alto de abstracción también permite discusiones sobre diversos niveles arquitectónicos. Éstos se pueden acompañar por las representaciones visuales de la arquitectura, por ejemplo con UML.
- Enfocarse en la calidad: El control de calidad no debe realizarse al final de cada iteración, sino en todos los aspectos de la producción. El aseguramiento de la calidad forma parte del proceso de desarrollo y no de un grupo independiente.

1.2.5.2 SCRUM

SCRUM (s/f) explicó que:

Es un marco de trabajo para la colaboración eficaz en equipo o proyectos complejos. Proporciona un conjunto de buenas prácticas para trabajar

colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos.

En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales.

Scrum también se utiliza para resolver situaciones en que no se está entregando al cliente lo que necesita, cuando las entregas se alargan demasiado, los costes se disparan o la calidad no es aceptable, cuando se necesita capacidad de reacción ante la competencia, cuando la moral de los equipos es baja y la rotación alta, cuando es necesario identificar y solucionar ineficiencias sistemáticamente o cuando se quiere trabajar utilizando un proceso especializado en el desarrollo de producto.

En Scrum un proyecto se ejecuta en bloques temporales cortos y fijos (iteraciones de un mes natural y hasta de dos semanas). Cada iteración tiene que proporcionar un resultado completo, un incremento de producto final que sea susceptible de

ser entregado con el mínimo esfuerzo al cliente cuando lo solicite.

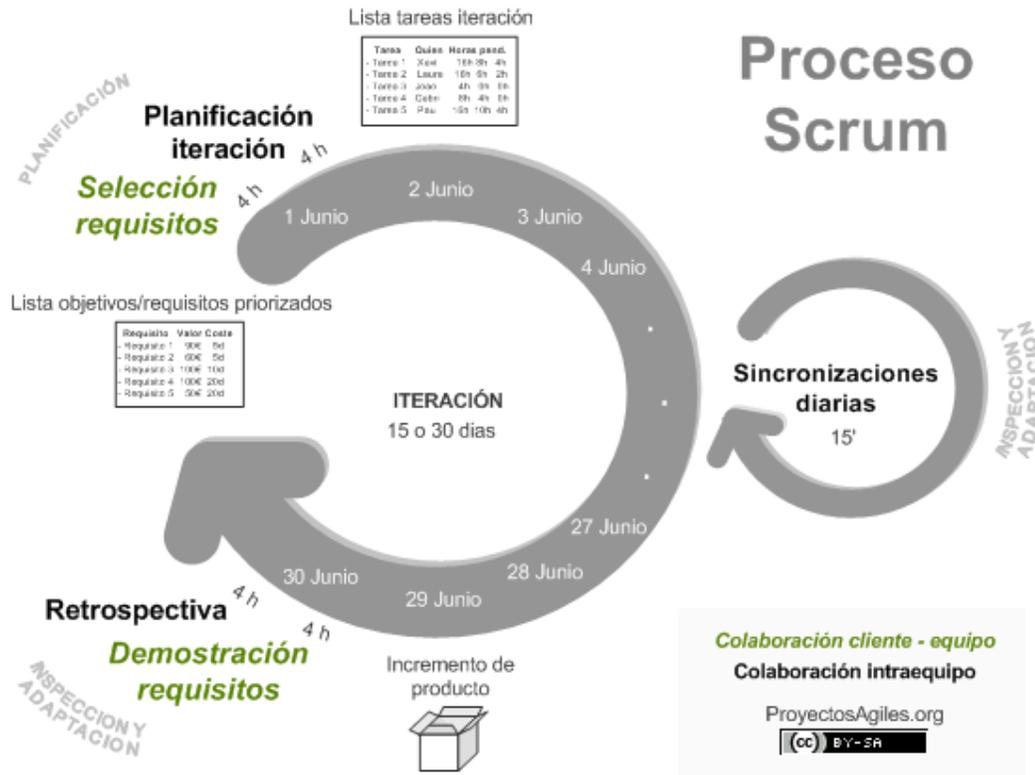


Figura N° 15 - Proceso Scrum

Fuente: Scrum.org (s/f).

El proceso parte de la lista de objetivos/requisitos priorizada del producto, que actúa como plan del proyecto. En esta lista el cliente prioriza los objetivos balanceando el valor que le aportan respecto a su coste y quedan repartidos en iteraciones y entregas. De manera regular el cliente puede maximizar la utilidad de lo que se desarrolla y el retorno de inversión mediante la replanificación de objetivos del producto, que realiza durante la iteración con vista a las siguientes iteraciones.

Las actividades que se llevan a cabo en SCRUM son las siguientes:

Planificación de la iteración	<p>El primer día de la iteración se realiza la reunión de planificación de la iteración. Tiene dos partes:</p> <ol style="list-style-type: none">1. Selección de requisitos (4 horas máximo). El cliente presenta al equipo la lista de requisitos priorizada del producto o proyecto. El equipo pregunta al cliente las dudas que surgen y selecciona los requisitos más prioritarios que se compromete a completar en la iteración, de manera que puedan ser entregados si el cliente lo solicita.2. Planificación de la iteración (4 horas máximo). El equipo elabora la lista de tareas de la iteración necesarias para desarrollar los requisitos a que se ha comprometido. La estimación de esfuerzo se hace de manera conjunta y los miembros del equipo se auto asignan las tareas.
Ejecución de la iteración	<p>Cada día el equipo realiza una reunión de sincronización (15 minutos máximo). Cada miembro del equipo inspecciona el trabajo que el resto está realizando (dependencias entre tareas, progreso hacia el objetivo de la iteración, obstáculos que pueden impedir este objetivo) para poder hacer las adaptaciones necesarias que permitan cumplir con el compromiso adquirido. En la reunión cada miembro del equipo responde a tres preguntas:</p> <ul style="list-style-type: none">• ¿Qué he hecho desde la última reunión de sincronización?• ¿Qué voy a hacer a partir de este momento?

	<ul style="list-style-type: none"> • ¿Qué impedimentos tengo o voy a tener? <p>Durante la iteración el Facilitador (Scrum Master) se encarga de que el equipo pueda cumplir con su compromiso y de que no se merme su productividad.</p> <ul style="list-style-type: none"> • Elimina los obstáculos que el equipo no puede resolver por sí mismo. • Protege al equipo de interrupciones externas que puedan afectar su compromiso o su productividad.
<p>Inspección y adaptación</p>	<p>El último día de la iteración se realiza la reunión de revisión de la iteración. Tiene dos partes:</p> <ol style="list-style-type: none"> 1. Demostración (4 horas máximo). El equipo presenta al cliente los requisitos completados en la iteración, en forma de incremento de producto preparado para ser entregado con el mínimo esfuerzo. En función de los resultados mostrados y de los cambios que haya habido en el contexto del proyecto, el cliente realiza las adaptaciones necesarias de manera objetiva, ya desde la primera iteración, re planificando el proyecto. 2. Retrospectiva (4 horas máximo). El equipo analiza cómo ha sido su manera de trabajar y cuáles son los problemas que podrían impedirle progresar adecuadamente, mejorando de manera continua su productividad. El Facilitador se encargará de ir eliminando los obstáculos identificados. <p>Actividades</p> <ul style="list-style-type: none"> • Planificación de la iteración (Sprint Planning) • Ejecución de la iteración (Sprint) • Reunión diaria de sincronización del equipo (Scrum Daily Meeting)

	<ul style="list-style-type: none"> • Demostración de los requisitos completados (Sprint Review) • Retrospectiva (Sprint Retrospective) • Replanificación del proyecto <p>Responsabilidades</p> <ul style="list-style-type: none"> • Cliente (Product Owner) • Facilitador (Scrum Master) • Equipo (Team) <p><i>Herramientas</i></p> <ul style="list-style-type: none"> • Lista de requisitos priorizada (Product Backlog) • Lista de tareas de la iteración (Sprint Backlog) • Gráficos de trabajo pendiente (Burndown Chart)
<p>Herramientas que se suelen utilizar en Scrum</p>	<ul style="list-style-type: none"> • Definición de completado (DoD) • Historias de usuario • Tablero de tareas • Planning Poker • Velocidad del equipo • Principios de Lean Software Development

Tabla N° 6 - Actividades SCRUM
Fuente: Scrum.org (s/f).

1.2.5.3 XP - Extreme Programing

Beck, A. (2004) explicó que:

La programación extrema, o Extreme Programming (XP), es una metodología de desarrollo ágil, una de las más exitosas en tiempo reciente. Su autor principal es Kent Beck, quien eligió algunas características de otras metodologías y las relacionó de forma que cada una complementara a la otra.

Así, la XP se puede definir como un conjunto de pasos de diversas metodologías, acopladas de manera que sean pasos flexibles a seguir utilizadas con el uso común, para realizar un desarrollo más agradable y sencillo.

Esta metodología tiene como base la simplicidad y como objetivo principal la satisfacción del cliente; para lograrlo se deben tomar en cuenta cuatro valores fundamentales:



a. Comunicación

Es muy importante que haya una comunicación constante con el cliente y dentro de todo el equipo de trabajo, de esto dependerá que el desarrollo se lleve a cabo de una manera sencilla, entendible y que se entregue al cliente lo que necesita.

b. Simplicidad

En la XP se refiere que ante todo y sin importar qué funcionalidad requiera el usuario en su sistema, éste debe ser fácil. El diseño debe ser sencillo y amigable al usuario, el código debe ser simple y entendible, programando solo lo necesario y lo que se utilizará.

c. Retroalimentación

Es la comunicación constante entre el desarrollador y el usuario.

d. Coraje

Se refiere a la valentía que se debe tener al modificar o eliminar el código que se realizó con tanto esfuerzo; el desarrollador debe

saber cuándo el código que desarrolló no es útil en el sistema y, por lo mismo, debe ser eliminado.

Dentro de la programación extrema se tiene 12 principios que llevan o guían el desarrollo con esta metodología:

- El principio de pruebas
- Proceso de planificación
- El cliente en el lugar
- Programación en parejas
- Integración continua
- Refactorización
- Entregas pequeñas
- Diseño simple
- Metáfora
- Propiedad colectiva del código
- Estándar de codificación
- La semana de 40 horas

1.2.5.3.1 HERRAMIENTAS DE LA XP

a. Historias de usuarios

Son tarjetas físicas en las cuales se anota una descripción de una funcionalidad del sistema, en una oración, se le da un número y un título para ser identificada.

b. Casos de prueba de aceptación

Son tarjetas que se elaboran para realizar las pruebas de cada historia de usuario.

c. Tarea de ingeniería

Son tarjetas que se elaboran para ayudar y simplificar la programación de una historia de usuario.

d. Tarjetas CRC

Describen las clases utilizadas en la programación de una historia.

e. Ventajas y desventajas

Una de las ventajas de la programación extrema es que se adapta al desarrollo de sistemas pequeños y grandes; optimiza el tiempo de desarrollo; permite realizar el desarrollo del sistema en parejas para complementar los conocimientos; el código es sencillo y entendible, además de la poca documentación a elaborar para el desarrollo del sistema.

Las desventajas son que no se tiene la definición del costo y el tiempo de desarrollo; el sistema va creciendo después de cada entrega al cliente y nadie puede decir que el cliente no querrá una función más; se necesita de la presencia constante del usuario. Otra desventaja es la programación en parejas, algunos desarrolladores son celosos del código que escriben y no les es grato que alguien más modifique las funciones que realizó o que su código sea desechado por no cubrir el estándar.

CARACTERÍSTICAS	RUP	XP	SCRUM
Heredan modelos	X	-	-
Documentación estricta	X	-	-
Más enfocado en los procesos	X	-	-
Más enfocado en las personas	-	X	X
Resultados rápidos	-	X	X
Cliente Activo	-	X	X
Manejo del tiempo	X	X	X
Refactorización del código	-	X	-
Iterativo	X	X	X
Respuesta a los cambios	-	X	X

Tabla N° 7 - Cuadro Comparativo Metodologías

Fuente: Pérez (2011).

1.2.6 Netbeans

Beck, A. (2004) explicó que:

NetBeans es un entorno de desarrollo integrado para el desarrollo principalmente con Java, pero también con otros idiomas, en particular, PHP, C/C, y HTML5. El IDE NetBeans está escrito en Java y puede ejecutarse en Windows, OS X, Linux, Solaris y otras plataformas.

La plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes modulares de software llamados módulos. Las aplicaciones basadas en la plataforma NetBeans pueden ser extendidos por los desarrolladores de terceros.

a. Características principales

- Asistentes para la creación y configuración de distintos proyectos, incluida la elección de algunos frameworks.
- Buen editor de código, multilenguaje, con el habitual coloreado y sugerencias de código, control de versiones, localización de ubicación de la clase actual, comprobaciones sintácticas y semánticas, plantillas de código.
- Simplifica la gestión de grandes proyectos con el uso de diferentes vistas, asistentes de ayuda, y estructurando la visualización de manera ordenada.
- Acceso a base de datos: desde el propio Netbeans podemos conectarnos a distintos sistemas gestores de bases de datos, como pueden ser Oracle, MySql y demás, y ver las tablas, realizar consultas y modificaciones, y todo ello integrado en el propio IDE.

1.2.7 Sistemas Operativos para Dispositivos Móviles

Figueredo (2006) indicó:

El sistema operativo (S.O.) se define como la capa compleja entre el hardware y el usuario, concebible también como una máquina virtual, que facilita al usuario o al programador las herramientas e interfaces adecuadas para realizar sus tareas informáticas. Un Sistema Operativo para dispositivos móviles es considerado el programa principal y éste es capaz de administrar todos sus recursos para ser utilizados de manera eficiente, cómoda y sin interrupciones, de tal manera que el usuario pueda mantener una comunicación sin problemas haciendo uso de los recursos que el hardware suministra. El uso de uno u otro S.O. determinará las capacidades multimedia de los dispositivos y la forma de interactuar con el usuario. Existen multitud de opciones, las más extendidas son Symbian, BlackBerry, Windows Mobile, iOS y Android, sin dejar de mencionar aquellos dispositivos con sistema operativo Linux. (párr. 2)

1.2.7.1 Android

Android Inc., (s/f) explicó:

ANDROID es un conjunto de software de código abierto creado para una amplia gama de dispositivos con diferentes factores de forma. El principal propósito de Android es la creación de una plataforma de software abierto disponible para fabricantes de equipos originales y desarrolladores

para hacer que sus ideas innovadoras se conviertan en realidad y para introducir un producto de éxito en el mundo real que mejora la experiencia móvil de los usuarios. El resultado es un producto de consumo total, con calidad de producción con código fuente abierto para la personalización y portabilidad.

a. Características de Android

Diseño De Dispositivo	La plataforma es adaptable a pantallas más grandes.
Almacenamiento	SQLite, una base de datos liviana, la cual es usada para propósitos de almacenamiento de datos
Conectividad	Android soporta las siguientes tecnologías de conectividad: GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE y WiMAX
Mensajería	SMS y MMS son formas de mensajería.
Navegador Web	El navegador web incluido en Android está basado en el motor de renderizado de código abierto WebKit, emparejado con el motor JavaScript V8 de Google Chrome.
Soporte De Java	Aunque la mayoría de las aplicaciones están escritas en Java, no hay una máquina virtual Java en la plataforma.
Soporte Para Hardware Adicional	Android soporta cámaras de fotos, de vídeo, pantallas táctiles, GPS, acelerómetros, giroscopios, magnetómetros, sensores de proximidad y de presión, sensores de luz, termómetro, aceleración por GPU 2D y 3D.
Entorno De Desarrollo	Incluye un emulador de dispositivos, herramientas para depuración de memoria y análisis del rendimiento del software.

Google Play	Google Play es un catálogo de aplicaciones gratuitas o de pago en el que pueden ser descargadas e instaladas en dispositivos Android sin la necesidad de un PC.
Multi-Táctil	Android tiene soporte nativo para pantallas capacitivas con soporte multi-táctil que inicialmente hicieron su aparición en dispositivos como el HTC Hero.
Video llamada	Android soporta video llamada a través de Google Talk desde su versión HoneyComb.
Multitarea	Multitarea real de aplicaciones está disponible, es decir, las aplicaciones que no estén ejecutándose en primer plano reciben ciclos de reloj.

Tabla N° 8 - Características de Android
Fuente: Android (s/f).

b. Arquitectura Android

Android está construido sobre kernel de Linux de código abierto y permite ser ampliado para incorporar nuevas tecnologías. Los componentes del sistema operativo Android se muestran en la siguiente imagen:



Figura N° 16 - Arquitectura Android
Fuente: Android Developers (s.f.).

- **Aplicaciones:** Incluyen un cliente de correo electrónico, programa de SMS, calendario, mapas, navegador, contactos y otros. Todas las aplicaciones escritas en Java.
- **Marco de trabajo de aplicaciones:** Los desarrolladores tienen acceso completo a los mismos APIs del framework usados por las aplicaciones base. La arquitectura está diseñada para simplificar la reutilización de componentes; cualquier aplicación puede publicar sus

capacidades y cualquier otra aplicación puede luego hacer uso de esas capacidades.

- Bibliotecas: Incluye un conjunto de bibliotecas de C/C++ usadas por varios componentes del sistema.
- Runtime de Android: Incluye un set de bibliotecas base que proporcionan la mayor parte de las funciones disponibles en las bibliotecas base del lenguaje Java. Cada aplicación Android corre su propio proceso, con su propia instancia de la máquina virtual Dalvik.
- Núcleo Linux: Android depende de Linux para los servicios base del sistema como seguridad, gestión de memoria, gestión de procesos, pila de red y modelo de controladores. También actúa como capa de abstracción entre el hardware y el resto de la pila de software.

c. **Versiones de Android**

- Basterra, Berteza, Borello, Castillo, y Venturi (2012) publicaron que “Android ha tenido numerosas actualizaciones desde su liberación inicial. Estas actualizaciones al sistema operativo base arreglan bugs y agregan nuevas funciones. Generalmente cada actualización del sistema es desarrollada bajo un nombre en código de un elemento relacionado con postres.

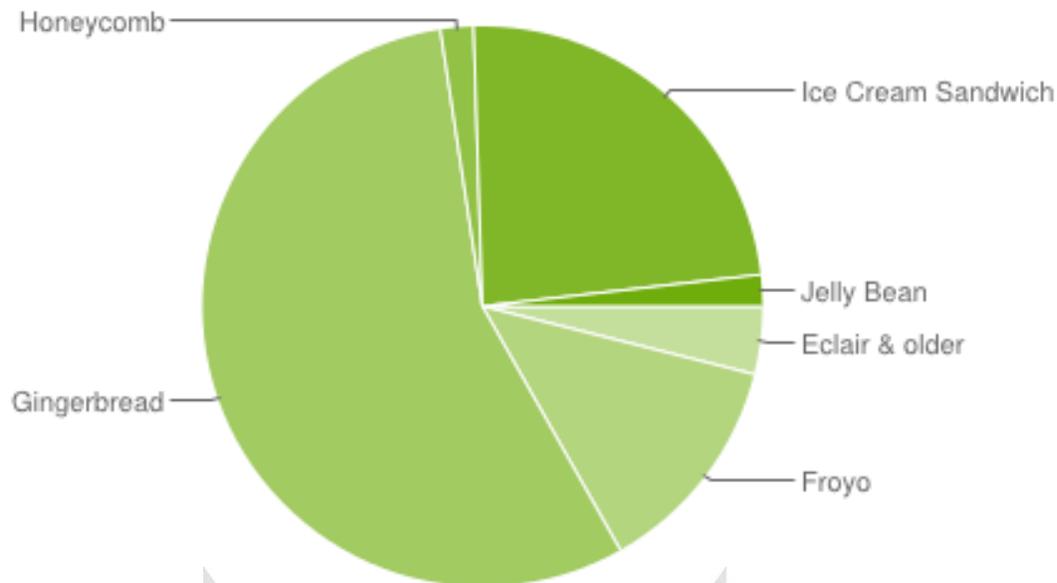


Figura N° 17 - Versiones de ANDROID
Fuente: Android Developers (s.f.).

NOMBRE	VERSIÓN / AÑO
Cupcake	Versión 1.5 / Abril - 2009
Donut	Versión 1.6 / Setiembre – 2009
Éclair	Versión 2.0 / Octubre – 2009
Froyo	Versión 2.2 / Mayo 2010
Gingerbread	Versión 2.3 / Diciembre 2010
Honeycomb	Versión 3.0 / Febrero 2011
Ice Cream Sandwich	Versión 4.0 / Octubre 2011
Jelly Bean	Versión 4.1 / Julio 2012
Jelly Bean	Versión 4.2 / Noviembre 2012
Jelly Bean	Versión 4.3 / 2013
KitKat	Versión 4.4 / Octubre 2013.
Lollipop	Versión 5.0 / Noviembre 2014

Tabla N° 9 - Versiones de ANDROID
Fuente: Android Developers (s.f.).

1.2.7.2 Blackberry Os

BlackBerry (2014) explicó:

BlackBerry OS es un sistema operativo móvil desarrollado por RIM para los dispositivos BlackBerry. El sistema permite multitarea y tiene soporte para diferentes métodos de entrada adoptados por RIM para su uso en computadoras de mano, particularmente la trackwheel, trackball, touchpad y pantallas táctiles. El SO BlackBerry está claramente orientado a su uso profesional como gestor de correo electrónico y agenda. Desde la cuarta versión se puede sincronizar el dispositivo con el correo electrónico, el calendario, tareas, notas y contactos de Microsoft Exchange Server además es compatible también con Lotus Notes y Novell GroupWise. BlackBerry Enterprise Server (BES) proporciona el acceso y organización del email a grandes compañías identificando a cada usuario con un único BlackBerry PIN.

a. Arquitectura BlackBerry OS

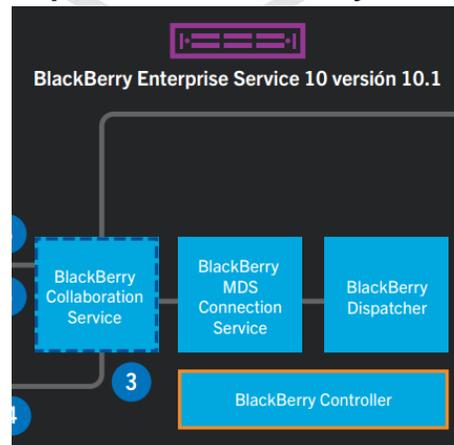


Figura N° 18 - Arquitectura BlackBerry OS
Fuente: Blackberry (2014).

➤ BlackBerry Device Service console:

Puede administrar cuentas de usuario y grupos, funciones administrativas, configuraciones de software, perfiles de correo electrónico y políticas de TI a las cuentas de usuario. La consola de BlackBerry Device Service se conecta a BlackBerry Configuration Database y Microsoft Active Directorio. La información de usuario actualizada en Microsoft Active Directory se puede sincronizar manualmente con BlackBerry Device Service console.

➤ BlackBerry Controller:

Supervisa BlackBerry Dispatcher, BlackBerry MDS Connection Service y BlackBerry Collaboration Service, y también los reinicia si dejan de responder.

➤ BlackBerry Dispatcher:

Mantiene una conexión SRP con BlackBerry Infrastructure a través de Internet. BlackBerry Dispatcher también comprime y cifra, y descifra y descomprime los datos que se transfieren a través de Internet desde y hasta los dispositivos.

➤ BlackBerry MDS Connection Service:

Proporciona una conexión segura entre el dispositivo con BlackBerry 10 y la red de la empresa, incluidas las conexiones a la intranet de la empresa y a BlackBerry Collaboration Service. La conexión se utiliza cuando el dispositivo no está conectado a la red Wi-Fi de la empresa. BlackBerry MDS Connection Service también se encarga de proporcionar la funcionalidad de inserción de la empresa.

b. Últimas Versiones de BlackBerry OS

BlackBerry 10 OS (s/f) publicó sobre el BlackBerry 10 OS:

➤ Lanzado 30 de enero de 2013.

➤ Cuenta con BlackBerry Priority Hub, que permite acceder a los mensajes, alertas, calendario, eventos todo en un lugar.

- Teclado predictivo, aprende la manera en la que escribes y sugiere palabras.
- Mejor consumo de la energía, hasta 25 horas de duración de la batería.
- Navegador de Internet mucho más dinámico.
- Mejora en las funciones de contados, fotos, editor de imágenes y calendario Inteligente.

1.2.7.3 IOS

Ramírez (2013) explicó:

La iOS es el sistema operativo móvil de la empresa Apple Inc. Originalmente desarrollado para el iPhone (iPhone OS), siendo después usado en dispositivos como el iPod Touch, iPad y el Apple TV, la instalación de este sistema operativo no es permitida en hardware de terceros.

La interfaz de usuario de iOS está basada en el concepto de manipulación directa, usando gestos multitáctiles. Los elementos de control consisten de deslizadores, interruptores y botones. La respuesta a las órdenes del usuario es inmediata y provee de una interfaz fluida. La interacción con el sistema operativo incluye gestos como deslices, toques, pellizcos, los cuales tienen definiciones diferentes dependiendo del contexto de la interfaz. Se utilizan acelerómetros internos para hacer que algunas aplicaciones respondan a sacudir el dispositivo (por ejemplo, para el comando deshacer) o rotarlo en tres dimensiones (un resultado común es cambiar de modo vertical al apaisado u horizontal).

iOS se deriva del sistema operativo de equipos de escritorio Mac OS X, que a su vez está basado en Darwin BSD, y por lo tanto es un sistema operativo Unix.” (p. 1)

a. Arquitectura iOS

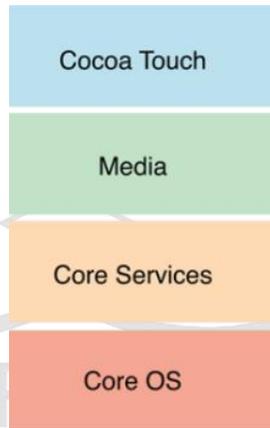


Figura N° 19 - Arquitectura de iOS
Fuente: Apple Inc.(2013).

CAPA	DESCRIPCIÓN
Capa Cocoa Touch	La capa de Cocoa Touch contiene marcos fundamentales para la construcción de aplicaciones de iOS. Estos marcos definen el aspecto de su aplicación. También proporcionan la aplicación básica de infraestructura y soporte para tecnologías claves como la multitarea, de entrada basada en el contacto, notificaciones push y muchos servicios del sistema de alto nivel.
Capa Media	La capa Media contiene los gráficos, audio y tecnologías de vídeo que se utilizan para implementar las multimedia en sus aplicaciones. Las tecnologías en esta capa hacen que sea fácil para construir aplicaciones que se ven y suenan muy bien.
Capa Core Services	La capa Core Services contiene los servicios fundamentales del sistema para las aplicaciones. La clave

CAPA	DESCRIPCIÓN
	entre el Servicio y el Core Foundation y Foundation Frameworks, definen los tipos básicos que todas las aplicaciones utilizan. Esta capa también contiene las tecnologías individuales para admitir características como la ubicación, iCloud, medios de comunicación social, y la creación de redes.
Capa Core OS	La capa Core OS contiene las características de bajo nivel que la mayoría de las otras tecnologías se basan. Incluso si no se utiliza estas tecnologías directamente en las aplicaciones, lo más probable es que estén siendo utilizados por otros marcos de trabajo. Y en situaciones en las que se necesite para hacer frente de forma explícita con la seguridad o la comunicación con un accesorio de hardware externo, lo hace utilizando el marco que en esta capa.

Tabla N° 10 – Capa IOS
Fuente: Apple Inc.(2013).

b. Últimas Versiones de iOS

Apple Inc. (2013) publicó que “la IOS 7

- Lanzado el 18 de Setiembre del 2013
- Rediseño de la interfaz y nuevas funciones.
- Implementación del Centro de Control, con rápidas opciones de configuración.
- El Centro de Notificaciones ofrece mucho más información que antes como fecha, mails, alertas, notificaciones de aplicaciones, etc.

- Nueva aplicación de cámara, coloca los formatos de disparos centrados y accesibles.
- Nuevos filtros en fotos y la opción de agregarlos en tiempo real.
- Ampliación de las opciones de conectividad entre sus dispositivos mediante AirDrop, esta tecnología buscar entre los usuarios más cercanos con quienes puedes compartir contenido.” (p. 1)

1.2.8 SQLite

SQLite (s/f) publicó que:

SQLite es una herramienta de software libre, que permite almacenar información en dispositivos empujados de una forma sencilla, eficaz, potente, rápida y en equipos con pocas capacidades de hardware, como puede ser una PDA o un teléfono celular. SQLite implementa el estándar SQL92 y también agrega extensiones que facilitan su uso en cualquier ambiente de desarrollo. Esto permite que SQLite soporte desde las consultas más básicas hasta las más complejas del lenguaje SQL, y lo más importante es que se puede usar tanto en dispositivos móviles como en sistemas de escritorio, sin necesidad de realizar procesos complejos de importación y exportación de datos, ya que existe compatibilidad al 100% entre las diversas plataformas disponibles, haciendo que la portabilidad entre dispositivos y plataformas sea transparente.

a. Características

Estas son algunas de las características principales de SQLite:

- La base de datos completa se encuentra en un solo archivo.
- Puede funcionar enteramente en memoria, lo que la hace muy rápida.
- Tiene un footprint menor a 230KB.
- Es totalmente autocontenida (sin dependencias externas).
- Cuenta con librerías de acceso para muchos lenguajes de programación.
- Soporta texto en formato UTF-8 y UTF-16, así como datos numéricos de 64 bits.
- Soporta funciones SQL definidas por el usuario (UDF).
- El código fuente es de dominio público y se encuentra muy bien documentado.

b. Plataformas de SQLite

SQLite está construida en C, lo cual facilita la migración a diversas plataformas de sistemas operativos y de dispositivos. Dado que una base de datos de SQLite se almacena por completo en un solo archivo, está puede ser exportada a cualquier otra plataforma y tener interoperabilidad al 100% sin ningún requerimiento de programación adicional o cambios de configuración.

Las plataformas principales dónde SQLite se encuentra funcionando son:

- Windows 95, 98, ME, 2000, XP y Vista
- Windows CE & Pocket PC

- Mac OSX
- Linux
- OpenEmbedded
- PalmOS
- Symbian

1.2.9 Telefonía Móvil

Huidobro y Conesa (2006) publicaron:

Las redes para comunicaciones móviles se iniciaron ya hace varias décadas, pero su uso se limitaba a ciertos servicios de carácter público tales como el servicio de policía, bomberos, ambulancias, etc. Estas redes, de uso privado, no tenían conexión a la red telefónica básica, por lo que cada entidad tenía que montar su propia infraestructura; posteriormente su uso se va extendiendo a otros servicios con lo que empieza a ser interesante y rentable disponer de una red amplia, con una infraestructura común, que pueda dar servicio a flotas o a todo el que requiera, conforme un estándar. Un red o servicio de este tipo, cuyos usuarios son individuales, es lo que se denominó inicialmente Telefonía Móvil Automática o TMA, utilizándose ahora, simplemente el término Telefonía Móvil (TM) o Celular. (p. 158)

Asimismo comentaron que, este servicio puede ser considerado en cierta forma como una extensión del servicio básico telefónico, ya que su finalidad es proporcionar idénticos servicios al usuario que se encuentra desplazándose, mediante un terminal telefónico portátil. Por consiguiente, éste puede efectuar y recibir

llamadas, a/ desde cualquier abonado fijo o móvil, nacional o internacional dentro de la zona de cobertura del sistema. La consecución de una amplia cobertura y capacidad de tráfico, utilizando un número limitado de frecuencias, es posible solo gracias al empleo de una estructura celular. (p. 159)

1.3 Definición términos básicos

a. ANDROID SDK (Software Development Kit)

Android Developers. (s.f.) indicó:

Es un kit de desarrollo de software ANDROID provee la librería de interfaz de programación de aplicación (Application Programming Interface - API) y las herramientas necesarias para construir, probar y proveer la librería API y herramientas de desarrollo necesarias para construir, probar las aplicaciones para Android.

b. Aplicación móvil

UNAD. (s/f) publicó que “las aplicaciones móviles son los conjuntos de instrucciones lógicas, procedimientos, reglas, documentación, datos e información asociada a estas que funcionan específicamente en dispositivos móviles, como por ejemplo teléfonos inteligentes, televisores inteligentes, tabletas, reloj, entre otros” (párr. 4).

c. Dirección de Investigación de Robo de Vehículos - DIROVE

DIROVE (s/f) publicó que “la DIROVE –PNP es una unidad especializada en la lucha contra el delito especial de hurto, robo y receptación de vehículos motorizados terrestres sus autopartes y accesorios” (p. 1)

d. Smartphone

Speckmann (2008) indicó:

Un Smartphone es un dispositivo electrónico que permite a su usuario llevar consigo un objeto de dimensiones pequeñas, que cumplan funciones similares a las de un computador personal y cuya evolución no se limita a las transferencias de voz sino también a cierta cantidad de funciones que han atraído la atención de los usuarios como por ejemplo: Navegar por internet, reproducir datos multimedia, realizar transacciones bancarias, orientarnos vía GPS, descargar juegos, conectarnos a las redes sociales, ver televisión vía streaming, etc.

Una de las funciones más importantes que caracteriza a un dispositivo móvil es la conectividad e interacción con las redes de datos, tal es el caso de internet el cual se ha convertido en un elemento fundamental para las comunicaciones.

Sin embargo, lo que hace atractivo a un Smartphone es el sistema operativo, que junto con el hardware lograr desarrollar operaciones que permiten correr cierta cantidad de procesos que son traducidos en importantes y llamativas aplicaciones, convirtiéndolos en dispositivos más competitivos que otros.

e. Software Libre

Hispa Linux (s/f) publicó:

Software Libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. De modo más preciso, se refiere a cuatro libertades de los usuarios del software:

- La libertad de usar el programa, con cualquier propósito (libertad 0).
- La libertad de estudiar cómo funciona el programa, y adaptarlo a tus necesidades (libertad 1). El acceso al código fuente es una condición previa para esto.
- La libertad de distribuir copias, con lo que puedes ayudar a tu vecino (libertad 2).
- La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie. (libertad 3). El acceso al código fuente es un requisito previo para esto.

Un programa es software libre si los usuarios tienen todas estas libertades. Así pues, deberías tener la libertad de distribuir copias, sea con o sin modificaciones, sea gratis o cobrando una cantidad por la distribución, a cualquiera y a cualquier lugar. El ser libre de hacer esto significa (entre otras cosas) que no tienes que pedir o pagar permisos.

También deberías tener la libertad de hacer modificaciones y utilizarlas de manera privada en tu trabajo u ocio, sin ni siquiera tener que anunciar que dichas modificaciones existen. Si publicas tus cambios, no tienes por qué avisar a nadie en particular, ni de ninguna manera en particular.

La libertad para usar un programa significa la libertad para cualquier persona u organización de usarlo en cualquier tipo de sistema informático, para cualquier clase de trabajo, y sin tener obligación de comunicárselo al desarrollador o a alguna otra entidad específica.

La libertad de distribuir copias debe incluir tanto las formas binarias o ejecutables del programa como su código fuente, sean

versiones modificadas o sin modificar (distribuir programas de modo ejecutable es necesario para que los sistemas operativos libres sean fáciles de instalar). Está bien si no hay manera de producir un binario o ejecutable de un programa concreto (ya que algunos lenguajes no tienen esta capacidad), pero debes tener la libertad de distribuir estos formatos si encontraras o desarrollaras la manera de crearlos.

Para que las libertades de hacer modificaciones y de publicar versiones mejoradas tengan sentido, debes tener acceso al código fuente del programa. Por lo tanto, la posibilidad de acceder al código fuente es una condición necesaria para el software libre.

Para que estas libertades sean reales, deben ser irrevocables mientras no hagas nada incorrecto; si el desarrollador del software tiene el poder de revocar la licencia aunque no le hayas dado motivos, el software no es libre.

Son aceptables, sin embargo, ciertos tipos de reglas sobre la manera de distribuir software libre, mientras no entren en conflicto con las libertades centrales. Por ejemplo, copyleft es la regla que implica que, cuando se redistribuya el programa, no se pueden agregar restricciones para denegar a otras personas las libertades centrales. Esta regla no entra en conflicto con las libertades centrales, sino que más bien las protege.

“Software libre” no significa “no comercial”. Un programa libre debe estar disponible para uso comercial, desarrollo comercial y distribución comercial. El desarrollo comercial del software libre ha dejado de ser inusual; el software comercial libre es muy importante.

Pero el software libre sin “copyleft” también existe. Creemos que hay razones importantes por las que es mejor usar “copyleft”,

pero si tus programas son software libre sin ser “copyleft”, los podemos utilizar de todos modos.

Cuando se habla de software libre, es mejor evitar términos como: “regalar” o “gratis”, porque esos términos implican que lo importante es el precio, y no la libertad.

f. StatCounter

StatCounter (s/f) indicó:

Stats se basa en datos agregados recopilados por StatCounter en una muestra superior a 15 mil millones de páginas vistas al mes recogidos de toda la red StatCounter de más de 3 millones de sitios web. Las estadísticas se actualizan y se ponen a disposición cada 4 horas, sin embargo están sujetos a pruebas de control de calidad y revisión durante 14 días a partir de su publicación.

CAPÍTULO II. METODOLOGÍA

2.1 Materiales

Los materiales elegidos para el presente proyecto han sido seleccionados en su mayoría, por ser gratuitos o de bajo costo. A continuación se lista los materiales identificados:

a. Base de Datos

- Sistema de Administración de Base de Datos MySQL, el cual ha sido elegido porque soporta diferentes sistemas operativos, trabaja con una gran variedad de compiladores, se amolda bien para aplicaciones ligeras de respuesta rápida, procesa y accede a gran volumen de información.
- SQLite ha sido elegido ya que se encuentra integrado en todos los dispositivos Android. Utilizando una base de datos SQLite en Android no requiere un procedimiento de configuración o administración de la base de datos.

b. Hardware

- Laptops (2 unidades), equipadas con sistema operativo Windows 7, DD 500 GB, 4GB de Memoria RAM, procesador Intel i5.
- Teléfono móvil inteligente – Smartphone (1 Unidad), de marca Samsung modelo Galaxy S3, con sistema operativo Android.

c. Software

- Android Studio, ha sido elegido por que permite ver los cambios visuales que se realizan en la aplicación en tiempo real, además se visualizan en diferentes dispositivos Android con distintas configuraciones y resoluciones en forma simultánea. Crea diferentes versiones de la aplicación.
- NetBeans, ha sido elegido por ser un entorno de desarrollo integrado, multilenguaje completo y modular, es Open Source y permite desarrollar aplicaciones de escritorio como web integrado con lenguaje HP.
- Servidor Apache + Tomcat, servidor de código abierto, disponible para una gran variedad de Sistemas Operativos, por ser de código fuente abierto, su software es transparente, cuenta con una amplia comunidad de desarrollo, con buen soporte y asistencia técnica.

2.2 Métodos

La metodología aplicada en el presente proyecto ha sido desarrollada en base a las buenas prácticas sugeridas por SCRUM.

2.2.1 Requerimientos

Esta sección se detalla los requerimientos funcionales y no funcionales identificados con el usuario.

a. Requerimientos Funcionales

En la siguiente tabla se especifican los requerimientos manifestados por el usuario (Historia de usuario - HU), para el desarrollo de la aplicación móvil Agente Ciudadano y los reportes en la página web.

PRODUCT BACKLOG

Código HU	Prioridad	Descripción
HU1	1	Crear una aplicación móvil gratuita.
HU2	2	Crear una aplicación móvil que permita registrar los datos básicos del cliente: Nombre completo, apellidos, DNI, correo electrónico, distrito.
HU3	2	Registrar datos de vehículo robado: Tipo, color, placa, característica específica.
HU4	3	Crear una página web que permita visualizar los vehículos registrados como robados.
HU5	3	Debe emitir mensajes de agradecimiento al usuario cuando alguno de los vehículos reportados ha sido recuperado.
HU6	1	La aplicación móvil debe poder visualizarse en Smartphone.
HU7	1	Crear una aplicación móvil que permite guardar la ubicación de autos robados.
HU8	1	Permitir consultar el estado del vehículo (robado, requisitoriado).
HU9	2	Cuando se realice la consulta de la placa debe mostrarse el estado de "Robado".
HU10	3	Debe permitir visualizar un historial de los vehículos registrados como robados por el usuario.
HU11	3	Debe permitir visualizar un historial de los vehículos reportados a través de la aplicación.
HU12	1	Crear una aplicación web que permita visualizar la

PRODUCT BACKLOG		
Código HU	Prioridad	Descripción
		información brindada por los ciudadanos y se encuentre a disposición de la Policía Nacional del Perú.
HU13	3	La aplicación web debe permitir exportar en Excel.
HU14	1	La aplicación web debe permitir visualizar: ubicación del vehículo, fecha de registro y hora.

Tabla N° 11 - Product Backlog
Fuente: Elaboración Propia

A continuación se detalla los requerimientos funcionales. Se puede observar la trazabilidad entre las historias de usuarios y requerimientos funcionales.

REQUERIMIENTOS FUNCIONALES		
Código	Descripción	Código HU
RF01	La primera vez que sea instalada la aplicación, deberán mostrarse las condiciones y términos de uso.	HU1, HU6
RF02	La aplicación móvil deberá solicitar la aceptación del usuario de las condiciones y términos de uso.	HU1, HU6
RF03	La aplicación móvil mostrará una pantalla inicial con el logo respectivo.	HU1, HU6
RF04	Se mostrarán opciones en la pantalla inicial.	HU1, HU6
RF05	La aplicación móvil deberá permitir rechazar los términos y condiciones.	HU1, HU6
RF06	Al finalizar la instalación, deberá mostrarse un mensaje dando la bienvenida al usuario. Luego mostrará el menú de opciones.	HU1, HU6
RF07	La aplicación móvil mostrará el menú de opciones	HU1, HU6

REQUERIMIENTOS FUNCIONALES

Código	Descripción	Código HU
	disponibles en la aplicación, en el siguiente orden: Consultar Placa, Actualizar Agente, Comunicar Robo.	
RF08	Los colores mostrados en el menú deberán tener relación con el logo de la aplicación.	HU1, HU6
RF09	Las opciones del menú deberán mostrarse en imágenes y textos.	HU1, HU6
RF10	Para poder hacer uso de las opciones del menú, el usuario deberá registrarse haciendo uso de la opción "Unirse".	HU1, HU6
RF11	La aplicación móvil deberá solicitar ingresar datos personales del usuario: Nombres y apellidos del usuario, DNI, correo electrónico.	HU2, HU4
RF12	Aplicación Móvil. Deberá validarse que todos los datos hayan sido ingresados.	HU2
RF13	Aplicación Móvil. Al finalizar el registro, deberá mostrarse un mensaje de bienvenida en pantalla.	HU2
RF14	Aplicación Móvil. Deberá mostrarse un solo campo de ingreso de texto, indicando que el texto a ingresar es la placa que desea consultarse.	HU9
RF15	Aplicación Móvil. Deberá existir un solo botón en esta pantalla, para consultar la placa del vehículo.	HU8, HU9
RF16	Aplicación Móvil. Al dar clic en el botón consultar, deberá mostrar si el	HU8, HU9

REQUERIMIENTOS FUNCIONALES

Código	Descripción	Código HU
	vehículo consultado es robado o no.	
RF17	<p>Aplicación Móvil.</p> <p>Si el vehículo tiene registrada una denuncia de robo, entonces deberá mostrarse un mensaje indicando que es un vehículo robado y habilitarse un botón para reportar la ubicación.</p>	HU7, HU9
RF18	<p>Aplicación Móvil.</p> <p>Si el vehículo no tiene registrada una denuncia de robo, entonces deberá mostrarse un mensaje indicando que el estado del vehículo es "No requisitoriado"</p>	HU8, HU9
RF19	Al elegir la opción "Reportar", deberá validarse que el GPS del Smartphone se encuentre activo, de no estarlo mostrará un mensaje al usuario para que proceda a activarlo.	HU7, HU11
RF20	Deberá mostrarse un mensaje de confirmación luego de haber comunicado la ubicación del vehículo.	HU7, HU11
RF21	La aplicación web deberá obtener y mostrar la ubicación del vehículo enviada desde el Smartphone.	HU7, HU11
RF22	<p>La aplicación móvil deberá obtener los siguientes datos:</p> <ul style="list-style-type: none"> - Placa: número de la placa - Marca: marca del vehículo (combo de selección) - Tipo: es el tipo de vehículo (combo de selección) - Color: color del vehículo (combo de selección) 	HU3

	<ul style="list-style-type: none"> - Foto: foto del vehículo (no obligatorio) - Características específicas: Campo que permita registrar características específicas del auto - Distrito: Distrito donde fue robado el auto (combo de selección) 	
RF23	<p>Aplicación Móvil.</p> <p>Solo debe mostrarse un botón que registre todos los datos.</p>	HU3, HU4
RF24	<p>Aplicación Móvil.</p> <p>Deberá validarse que todos los datos sean ingresados.</p>	HU3
RF25	<p>La aplicación web debe contar con una página de inicio que muestre el logo de la aplicación móvil y el login de ingreso.</p>	HU12, HU14
RF26	<p>La aplicación web debe estar a disponibilidad de la Policía Nacional del Perú.</p>	HU12, HU14
RF27	<p>La aplicación web debe mostrar opciones de búsqueda para consultar los reportes realizados por los Agentes ciudadanos.</p>	HU12, HU14
RF28	<p>La aplicación web debe mostrar la ubicación enviada desde el Smartphone, fecha y hora de registro.</p>	HU12, HU14
RF29	<p>La aplicación web deberá permitir exportar en Excel lo mostrado en pantalla.</p>	HU13
RF30	<p>El usuario deberá contar con una cuenta de correo que se encuentre asociada a Google Play.</p>	HU1, HU6
RF31	<p>Al recuperarse el vehículo reportado por el Agente Ciudadano, el sistema deberá mostrar un mensaje de agradecimiento al agente.</p>	HU5

Tabla N° 12 - Requerimientos Funcionales
Fuente: Elaboración Propia

b. Requerimientos No Funcionales

A continuación se listan los requerimientos no funcionales identificados con el usuario, para el desarrollo de la aplicación móvil Agente Ciudadano.

- La aplicación debe ser escalable para seguir creciendo con nuevas funcionalidades para adecuarse a las necesidades de los usuarios.
- La aplicación debe tener la capacidad de realizar las funciones para las que está diseñado las 24 horas del día, los 7 días de la semana.
- La aplicación debe ser segura y confiable, se debe asegurar la confidencialidad del usuario.
- La aplicación debe ser compatible con versiones de Android superiores a la 2.3 Gingerbread.
- La aplicación debe tener acceso a los sistemas de requisitorias y denuncias de la Policía Nacional del Perú.
- La aplicación debe ser accesible a través de las tiendas de aplicaciones para su descarga.
- El Smartphone debe tener como mínimo una memoria de 512 MB de memoria RAM para la ejecución de la aplicación.
- El Smartphone debe tener como mínimo de espacio de 50 MB para realizar la instalación de la aplicación.
- El Smartphone debe estar provisto de una antena GPS que brinde la ubicación del usuario.
- La aplicación web debe tener la capacidad de ejecutarse en diferentes navegadores como Internet Explorer.

- La página web debe estar disponible las 24 horas del día, los 7 días de la semana, debe tener una conectividad a internet y acceso de forma constante.
- La aplicación web debe ser intuitiva. Debe permitir al usuario la localización de la información de manera rápida y fácil.
- La aplicación web debe permitir loguearse de manera que solo puedan acceder las personas autorizadas.
- La aplicación web debe contar con un diseño amigable para el usuario.
- La aplicación web deberá permitir la actualización del software base y módulos, sin comprometer la funcionalidad de la aplicación.

c. Metodología Aplicada

El proyecto estará comprendido por 3 etapas, las cuales a su vez se dividen en las siguientes fases:

SCRUM	FASE	ENTREGABLE
Planificación de la Entrega	Definición	- Product Backlog
Sprint (*)	Análisis y Diseño	- Prototipos
	Construcción	- Código Fuente desarrollado
	Pruebas	- Pruebas y Resultado de pruebas
Despliegue	Despliegue	- Documento de Capacitación - Código fuente de la aplicación

Tabla N° 13 - SCRUM –Etapas

Fuente: Elaboración Propia

(*) Se realizarán 3 Sprint y tendrán una duración de 2 semanas c/u.

A continuación definiremos cada etapa y fase comprendida en el proyecto:

d. Planificación de la Iteración

1) Definición

En esta etapa se definirá la lista de requisitos priorizada, la cual en adelante llamaremos Product Backlog. El Product backlog deberá ser priorizado por el Product Owner, con apoyo del Scrum Master y el equipo.

- Se definirán a detalle los requisitos priorizados.
- Se definirá las funcionalidades a desarrollar.
- Se definirá la cantidad de requerimientos a ejecutar por cada sprint, delimitando así el alcance de cada sprint.

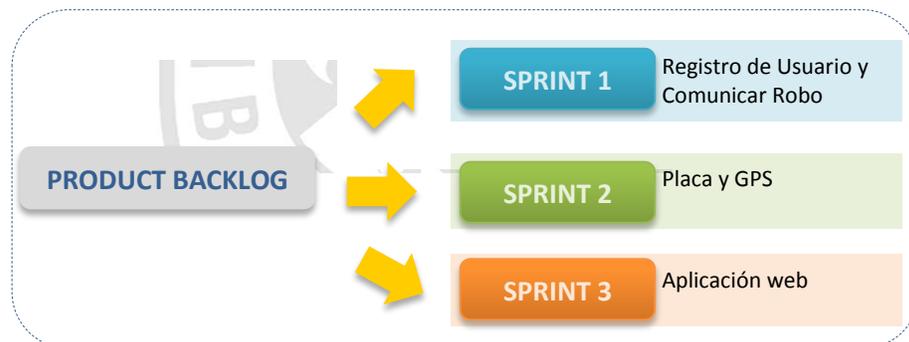


Figura N° 20 - Planificación de la Iteración
Fuente: Elaboración Propia

2) SPRINT

Es una iteración limitada en bloques de tiempo; ocurre uno tras otro. La duración máxima del Sprint será de 3 semanas. Se compone de:

a) FASE: Análisis Y Diseño

Se diseñará y detallará las especificaciones para la implementación del producto. El equipo del proyecto definirá la solución de acuerdo a los requerimientos que se atenderán por Sprint.

- Creación de prototipos; con el objetivo de describir el diseño y la presentación que deberá tener la solución final, con la finalidad de usarlo como esquema para la construcción de la solución.
- La creación de los primeros prototipos, será un diseño visual que será provisto al desarrollador para la programación del código.

b) FASE: Construcción

Desarrollo de la solución en base a los prototipos entregados en la etapa de diseño.

- Se realizarán pruebas unitarias de la versión desarrollada y pruebas de la funcionalidad en conjunto.
- El Código fuente a entregar representa todos los componentes, librerías, procedimientos almacenados o archivos modificados en la solución actual desarrollados como parte del alcance, los cuales serán propiedad del Cliente.
- Durante las reuniones diarias (Scrum daily meetings), el equipo hará uso del Scrum Taskboard para el seguimiento de las tareas realizadas (pendiente, en curso, terminado)

c) FASE: Pruebas

Se realizarán pruebas de la versión obtenida en el sprint y se brindará el soporte correctivo necesario a las observaciones que se puedan detectar en esta etapa.

d) FASE: Despliegue

Una vez aceptado el producto, El Product Owner coordinará con el equipo del proyecto la instalación del producto para ser usado en su primera versión por un grupo reducido de usuarios. Con la finalidad de iniciar el proceso de maduración del producto.

e) Roles y Responsabilidades

Durante los Sprints es necesario contar con 1 recurso del Cliente que apoye al equipo.

ROLES	RESPONSABILIDADES
Scrum Master	Como Gerente de proyectos está a cargo de orquestar al equipo del Proyecto. Asegura el cumplimiento de las reglas establecidas, el Product Backlog, realizar reuniones de SCRUM para verificar cumplimientos de objetivos y que el equipo cuente con las condiciones necesarias para que pueda cumplir con el objetivo.
Propietario del Producto (Product Owner)	Recurso del Cliente, representante de todas las personas interesadas en el proyecto, con autoridad de decisiones y el único con relación directa ante el equipo. Define los objetivos del producto, prioriza requisitos, aprueba calendarios de entrega, colabora con la planificación de objetivos de cada iteración.
Analista Funcional	Responsable del análisis de los requerimientos del diseño. Debe crear las historias de usuarios. Diseñar los prototipos.
Desarrollador	Responsable del diseño, modelamiento,

ROLES	RESPONSABILIDADES
	construcción y pruebas de los componentes.
Tester	Responsable de la elaboración de pruebas de la solución durante el proceso de pruebas de la misma.

Tabla N° 14 - Roles y Responsabilidades
Fuente: Elaboración Propia



3) Cronograma

	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
0	Proyecto de Tesis: Agente Ciudadano	59.5 días	lun 11/08/14	vie 21/11/14		
1	▢ FASE Inicial	4 días	lun 11/08/14	vie 15/08/14		Silvia Barba,Sofía Acuña
2	Levantamiento inicial de información	1 día	lun 11/08/14	mar 12/08/14		Silvia Barba,Sofía Acuña
3	▢ Identificación de usuarios claves	0.8 días	mar 12/08/14	mié 13/08/14		Silvia Barba,Sofía Acuña
4	Entrevista con usuarios	0.3 días	mar 12/08/14	mar 12/08/14	2	Silvia Barba,Sofía Acuña
5	Levantamiento general de información	0.2 días	mar 12/08/14	mar 12/08/14	4	Silvia Barba,Sofía Acuña
6	Presentación de idea general con usuarios	0.3 días	mar 12/08/14	mié 13/08/14	5	Silvia Barba,Sofía Acuña
7	Definición de problemática	0.2 días	mié 13/08/14	mié 13/08/14	6	Silvia Barba,Sofía Acuña
8	Identificación de posibles metodología a implementar	0.5 días	mié 13/08/14	jue 14/08/14	7	Silvia Barba,Sofía Acuña
9	▢ Definición de objetivo	0.5 días	mié 13/08/14	mié 13/08/14	3	Silvia Barba,Sofía Acuña
10	Definición de objetivo general	0.3 días	mié 13/08/14	mié 13/08/14		Silvia Barba,Sofía Acuña
11	Definición de Objetivos específicos	0.2 días	mié 13/08/14	mié 13/08/14	10	Silvia Barba,Sofía Acuña
12	▢ Definición de Justificación	1 día	mié 13/08/14	vie 15/08/14	9	Silvia Barba,Sofía Acuña
13	Definición de Justificación prácticas	0.3 días	mié 13/08/14	jue 14/08/14		Silvia Barba,Sofía Acuña
14	Definición de Justificación social	0.4 días	jue 14/08/14	jue 14/08/14	13	Silvia Barba,Sofía Acuña
15	Definición de Justificación económicas	0.3 días	jue 14/08/14	vie 15/08/14	14	Silvia Barba,Sofía Acuña
16	Documentación	4 días	lun 11/08/14	vie 15/08/14		Silvia Barba,Sofía Acuña
17	▢ Marco teórico	5 días	lun 18/08/14	mar 26/08/14	1	Silvia Barba,Sofía Acuña
18	▢ Definición de Antecedentes	1.7 días	lun 18/08/14	mié 20/08/14		Silvia Barba,Sofía Acuña
19	Identificación de los antecedentes del problema	0.5 días	lun 18/08/14	lun 18/08/14	16	Silvia Barba,Sofía Acuña
20	Identificación de herramientas	0.5 días	lun 18/08/14	mar 19/08/14	19	Silvia Barba,Sofía Acuña
21	Comparación y selección de herramientas para el desarrollo del proyecto	0.3 días	mar 19/08/14	mar 19/08/14	20	Silvia Barba,Sofía Acuña



	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
22	Análisis de casos de éxito	0.4 días	mar 19/08/14	mié 20/08/14	21	Silvia Barba,Sofia Acuña
23	Definición de Bases teóricas	1 día	mié 20/08/14	jue 21/08/14		Silvia Barba,Sofia Acuña
24	Identificación de herramientas para solución del problema	1 día	mié 20/08/14	jue 21/08/14	22	Silvia Barba,Sofia Acuña
25	Identificación de metodologías	1.3 días	jue 21/08/14	vie 22/08/14		Silvia Barba,Sofia Acuña
26	Definición de materiales	0.5 días	jue 21/08/14	jue 21/08/14	24	Silvia Barba,Sofia Acuña
27	Definición de metodologías	0.6 días	vie 22/08/14	vie 22/08/14	26	Silvia Barba,Sofia Acuña
28	Comparación y selección de metodología a implementar	0.2 días	vie 22/08/14	vie 22/08/14	27	Silvia Barba,Sofia Acuña
29	Definición de términos Básicos	0.6 días	lun 25/08/14	lun 25/08/14	28	Silvia Barba,Sofia Acuña
30	Documentación	5 días	lun 18/08/14	mar 26/08/14		Silvia Barba,Sofia Acuña
31	Elaboración de Metodología	49.5 días	mar 26/08/14	jue 20/11/14	17	Silvia Barba,Sofia Acuña
32	Definición de alcance	0.5 días	mar 26/08/14	mar 26/08/14	30	Silvia Barba,Sofia Acuña
33	Planificación de la Entrega	5 días	mar 26/08/14	jue 04/09/14		Silvia Barba,Sofia Acuña
34	Definición de tareas	1 día	mar 26/08/14	jue 28/08/14	32	Silvia Barba,Sofia Acuña
35	Elaboración de cronograma	0.5 días	jue 28/08/14	jue 28/08/14	34	Silvia Barba,Sofia Acuña
36	Requerimientos priorizados	3.5 días	jue 28/08/14	jue 04/09/14		Silvia Barba,Sofia Acuña
37	Identificación de prioridades	0.5 días	jue 28/08/14	vie 29/08/14	35	Silvia Barba,Sofia Acuña
38	Validación de prioridades con usuario	0.5 días	vie 29/08/14	vie 29/08/14	37	Silvia Barba,Sofia Acuña
39	Identificación de Sprints	1 día	lun 01/09/14	mar 02/09/14	38	Silvia Barba,Sofia Acuña
40	Generación de Backlog	1 día	mar 02/09/14	mié 03/09/14	39	Silvia Barba,Sofia Acuña
41	Entrega de Backlog	0.5 días	mié 03/09/14	jue 04/09/14	40	Silvia Barba,Sofia Acuña
42	Sprints	42 días	jue 04/09/14	lun 17/11/14	33	Silvia Barba,Sofia Acuña
43	Sprint 1 - Registro de usuarios y Comunicar Robo	15 días	jue 04/09/14	mar 30/09/14		Silvia Barba,Sofia Acuña
44	Análisis y Diseño	4 días	jue 04/09/14	jue 11/09/14		Silvia Barba,Sofia Acuña
45	Análisis de requerimientos	1 día	jue 04/09/14	vie 05/09/14	41	Silvia Barba,Sofia Acuña
46	Documentación	1 día	vie 05/09/14	lun 08/09/14	45	Silvia Barba,Sofia Acuña
	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
47	Diseño	2 días	lun 08/09/14	jue 11/09/14		Silvia Barba,Sofia Acuña
48	Diseño de prototipos	1 día	lun 08/09/14	mar 09/09/14	46	Silvia Barba,Sofia Acuña
49	Validación de prototipos con usuario	1 día	mar 09/09/14	jue 11/09/14	48	Silvia Barba,Sofia Acuña
50	Entrega de prototipos	0 días	jue 11/09/14	jue 11/09/14	49	Silvia Barba,Sofia Acuña
51	Construcción	7 días	jue 11/09/14	mar 23/09/14		Silvia Barba,Sofia Acuña
52	Desarrollo	5 días	jue 11/09/14	vie 19/09/14	50	Sofia Acuña,Silvia Barba
53	Pruebas unitarias	2 días	vie 19/09/14	mar 23/09/14	52	Sofia Acuña,Silvia Barba
54	Certificación	4 días	mar 23/09/14	mar 30/09/14		Silvia Barba,Sofia Acuña
55	Generación del Plan de Prueba	1 día	mar 23/09/14	jue 25/09/14	53	Silvia Barba,Sofia Acuña
56	Ejecución de pruebas	2 días	jue 25/09/14	lun 29/09/14	55	Silvia Barba,Sofia Acuña
57	Elaboración de Manual de Usuario	0.5 días	lun 29/09/14	mar 30/09/14	56	Silvia Barba,Sofia Acuña
58	Entrega de resultado de pruebas	0.5 días	mar 30/09/14	mar 30/09/14	57	Silvia Barba,Sofia Acuña
59	Reunión con Usuario	0.5 días	mar 30/09/14	mié 01/10/14	43	Silvia Barba,Sofia Acuña
60	Acta de aceptación de usuario	0.5 días	mié 01/10/14	jue 02/10/14	59	Silvia Barba,Sofia Acuña
61	Documentación de feedback	1 día	jue 02/10/14	vie 03/10/14	60	Silvia Barba,Sofia Acuña
62	Sprint 2 - placa y GPS	14.5 días	mar 30/09/14	vie 24/10/14	43	Silvia Barba,Sofia Acuña
63	Análisis y Diseño	4 días	mar 30/09/14	mar 07/10/14		Silvia Barba,Sofia Acuña
64	Análisis de requerimientos	1 día	vie 03/10/14	lun 06/10/14	61	Silvia Barba,Sofia Acuña
65	Documentación	1 día	lun 06/10/14	mar 07/10/14	64	Silvia Barba,Sofia Acuña
66	Diseño	2 días	mar 30/09/14	vie 03/10/14		Silvia Barba,Sofia Acuña
67	Diseño de prototipos	1 día	mar 30/09/14	jue 02/10/14		Silvia Barba,Sofia Acuña
68	Validación de prototipos con usuario	1 día	jue 02/10/14	vie 03/10/14	67	Silvia Barba,Sofia Acuña
69	Entrega de prototipos	0 días	vie 03/10/14	vie 03/10/14	68	Silvia Barba,Sofia Acuña
70	Construcción	7 días	mar 07/10/14	lun 20/10/14	63	Silvia Barba,Sofia Acuña
71	Desarrollo	5 días	mar 07/10/14	jue 16/10/14		Sofia Acuña,Silvia Barba
72	Pruebas unitarias	2 días	jue 16/10/14	lun 20/10/14	71	Sofia Acuña,Silvia Barba

	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
73	[-] Certificación	3.5 días	lun 20/10/14	vie 24/10/14		Silvia Barba,Sofia Acuña
74	Generación del Plan de Prueba	1 día	lun 20/10/14	mar 21/10/14	72	Silvia Barba,Sofia Acuña
75	Ejecución de pruebas	2 días	mar 21/10/14	vie 24/10/14	74	Silvia Barba,Sofia Acuña
76	Elaboración de Manual de Usuario	0.5 días	vie 24/10/14	vie 24/10/14	75	Silvia Barba,Sofia Acuña
77	Entrega de resultado de pruebas	0.5 días	vie 24/10/14	vie 24/10/14	75	Silvia Barba,Sofia Acuña
78	Reunión con usuario	0.5 días	lun 27/10/14	lun 27/10/14	77	Silvia Barba,Sofia Acuña
79	Acta de aceptación de usuario	0.5 días	lun 27/10/14	mar 28/10/14	78	Silvia Barba,Sofia Acuña
80	Documentación de feedback	1 día	mar 28/10/14	mié 29/10/14	79	Silvia Barba,Sofia Acuña
81	[-] Sprint 3 - Aplicación Web	12.5 días	lun 27/10/14	lun 17/11/14		Silvia Barba,Sofia Acuña
82	[-] Análisis y Diseño	3 días	lun 27/10/14	jue 30/10/14	62	Silvia Barba,Sofia Acuña
83	Análisis de requerimientos	1 día	lun 27/10/14	mar 28/10/14		Silvia Barba,Sofia Acuña
84	Documentación	1 día	mar 28/10/14	mié 29/10/14	83	Silvia Barba,Sofia Acuña
85	[-] Diseño	2 días	mar 28/10/14	jue 30/10/14		Silvia Barba,Sofia Acuña
86	Diseño de prototipos	1 día	mar 28/10/14	mié 29/10/14	83	Silvia Barba,Sofia Acuña
87	Validación de prototipos con usuario	1 día	mié 29/10/14	jue 30/10/14	86	Silvia Barba,Sofia Acuña
88	Entrega de prototipos	0 días	jue 30/10/14	jue 30/10/14	87	Silvia Barba,Sofia Acuña
89	[-] Construcción	7 días	jue 30/10/14	mié 12/11/14	82	Silvia Barba,Sofia Acuña
90	Desarrollo	5 días	jue 30/10/14	vie 07/11/14	88	Sofia Acuña,Silvia Barba
91	Pruebas unitarias	2 días	lun 10/11/14	mié 12/11/14	90	Sofia Acuña,Silvia Barba
92	[-] Certificación	2.5 días	mié 12/11/14	lun 17/11/14		Silvia Barba,Sofia Acuña
93	Generación del Plan de Prueba	1 día	mié 12/11/14	jue 13/11/14	89	Silvia Barba,Sofia Acuña
94	Ejecución de pruebas	2 días	mié 12/11/14	vie 14/11/14	91	Silvia Barba,Sofia Acuña
95	Elaboración de Manual de Usuario	0.5 días	lun 17/11/14	lun 17/11/14	94	Silvia Barba,Sofia Acuña
96	Entrega de resultado de pruebas	0 días	vie 14/11/14	vie 14/11/14	94	Silvia Barba,Sofia Acuña
97	Informe final de resultado de pruebas	1 día	jue 04/09/14	vie 05/09/14		Silvia Barba,Sofia Acuña
	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
98	Reunión con Usuario	0.5 días	vie 05/09/14	vie 05/09/14	97	Silvia Barba,Sofia Acuña
99	Acta de aceptación de usuario	0.5 días	lun 08/09/14	lun 08/09/14	98	Silvia Barba,Sofia Acuña
100	Despliegue	2 días	lun 17/11/14	jue 20/11/14	42	Silvia Barba,Sofia Acuña
101	Cierre del proyecto	1 día	jue 20/11/14	vie 21/11/14	100	Silvia Barba,Sofia Acuña

2.3 Desarrollo del Proyecto

En el presente capítulo se desarrollaran los entregables identificados en cada Sprint los cuales fueron indicados en el capítulo anterior, haciendo uso de la metodología elegida.

La implementación del proyecto se encuentra dividida de la siguiente manera:

2.3.1 SPRINT 1: Registro de Usuario y Comunicar Robo

En este Sprint se desarrolla las actividades necesarias para registrar un usuario y generaciones notificaciones.

RF= Requerimiento Funcional, HU= Historia de Usuario.

Código	Nombre	Descripción	Código HU
RF01	Instalación de la aplicación	La primera vez que sea instalada la aplicación, deberán mostrarse las condiciones y términos de uso.	HU1, HU6
RF02		La aplicación móvil deberá solicitar la aceptación del usuario de las condiciones y términos de uso.	HU1, HU6
RF03		La aplicación móvil mostrará una pantalla inicial con el logo respectivo.	HU1, HU6
RF04		Se mostrarán opciones en la pantalla inicial.	HU1, HU6
RF05		La aplicación móvil deberá permitir rechazar los términos y condiciones.	HU1, HU6
RF06		Al finalizar la instalación, deberá mostrarse un mensaje dando la bienvenida al usuario. Luego mostrará el menú de opciones.	HU1, HU6
RF07		La aplicación móvil mostrará el menú de	HU1,

Código	Nombre	Descripción	Código HU
		opciones disponibles en la aplicación, en el siguiente orden: Consultar Placa, Actualizar Agente, Comunicar Robo.	HU6
RF08	Menú de opciones	Los colores mostrados en el menú deberán tener relación con el logo de la aplicación.	HU1, HU6
RF09		Las opciones del menú deberán mostrarse en imágenes y textos.	HU1, HU6
RF10		Para poder hacer uso de las opciones del menú, el usuario deberá registrarse haciendo uso de la opción "Unirse".	HU1, HU6
RF11	Registrar Agente	La aplicación móvil deberá solicitar ingresar datos personales del usuario: Nombres y apellidos del usuario, DNI, correo electrónico.	HU2, HU4
RF12		Aplicación Móvil. Deberá validarse que todos los datos hayan sido ingresados.	HU2
RF13		Aplicación Móvil. Al finalizar el registro, deberá mostrarse un mensaje de bienvenida en pantalla.	HU2
RF22	Comunicar Robo	La aplicación móvil deberá obtener los siguientes datos: <ul style="list-style-type: none"> - Placa: número de la placa - Marca: marca del vehículo (combo de selección) - Tipo: es el tipo de vehículo (combo de selección) - Color: color del vehículo (combo de 	HU3

Código	Nombre	Descripción	Código HU
		selección) - Foto: foto del vehículo (no obligatorio) - Características específicas: Campo que permita registrar características específicas del auto - Distrito: Distrito donde fue robado el auto (combo de selección)	
RF23		Aplicación Móvil. Solo debe mostrarse un botón que registre todos los datos.	HU3, HU4
RF24		Aplicación Móvil. Deberá validarse que todos los datos sean ingresados.	HU3

Tabla N° 15 - SPRINT 1: Requerimientos Funcionales
 Fuente: Elaboración Propia

Ver Anexo 3 – Diseño de Prototipos - Registro de Usuario y Comunicar Robo

2.3.2 SPRINT 2: Placa y GPS

En este Sprint se desarrolla las actividades necesarias para registrar un usuario y generaciones notificaciones.

Dónde: RF= Requerimiento Funcional, HU= Historia de Usuario.

Código	Nombre	Descripción	Código HU
RF14	Consultar Placa	Aplicación Móvil. Deberá mostrarse un solo campo de ingreso	HU9

Código	Nombre	Descripción	Código HU
		de texto, indicando que el texto a ingresar es la placa que desea consultarse.	
RF15		Aplicación Móvil. Deberá existir un solo botón en esta pantalla, para consultar la placa del vehículo.	HU8, HU9
RF16		Aplicación Móvil. Al dar clic en el botón consultar, deberá mostrar si el vehículo consultado es robado o no.	HU8, HU9
RF17		Aplicación Móvil. Si el vehículo tiene registrada una denuncia de robo, entonces deberá mostrarse un mensaje indicando que es un vehículo robado y habilitarse un botón para reportar la ubicación.	HU7, HU9
RF18		Aplicación Móvil. Si el vehículo no tiene registrada una denuncia de robo, entonces deberá mostrarse un mensaje indicando que el estado del vehículo es "No requisitoriado"	HU8, HU9
RF19	Reportar Robo	Al elegir la opción "Reportar", deberá validarse que el GPS del Smartphone se encuentre activo, de no estarlo mostrará un mensaje al usuario para que proceda a activarlo.	HU7, HU11
RF20		Deberá mostrarse un mensaje de confirmación luego de haber comunicado la	HU7, HU11

Código	Nombre	Descripción	Código HU
		ubicación del vehículo.	
RF21		La aplicación web deberá obtener y mostrar la ubicación del vehículo enviada desde el Smartphone.	HU7, HU11

Tabla N° 16 - SPRINT 2: Requerimientos Funcionales
Fuente: Elaboración Propia

Ver Anexo 4 – Diseño de Prototipos - Placa y GPS

2.3.3 SPRINT 3: Aplicación web

En este Sprint se desarrolla las actividades necesarias para registrar un usuario y generaciones notificaciones.

Dónde: RF= Requerimiento Funcional, HU= Historia de Usuario.

Código	Nombre	Descripción	Código HU
RF25		La aplicación web debe contar con una página de inicio que muestre el logo de la aplicación móvil y el login de ingreso.	HU12, HU14
RF26	Aplicación Web	La aplicación web debe estar a disponibilidad de la Policía Nacional del Perú.	HU12, HU14
RF27		La aplicación web debe mostrar opciones de búsqueda para consultar los reportes realizados por los Agentes ciudadanos.	HU12, HU14
RF28		La aplicación web debe mostrar la ubicación enviada desde el Smartphone, fecha y hora de registro.	HU12, HU14

Código	Nombre	Descripción	Código HU
RF29		La aplicación web deberá permitir exportar en Excel lo mostrado en pantalla.	HU13
RF30		El usuario deberá contar con una cuenta de correo que se encuentre asociada a Google Play.	HU1, HU6
RF31	Generales	Al recuperarse el vehículo reportado por el Agente Ciudadano, el sistema deberá mostrar un mensaje de agradecimiento al agente.	HU5

Tabla N° 17 - SPRINT 3: Requerimientos Funcionales
Fuente: Elaboración Propia

Ver Anexo 5 – Diseño de Prototipos - Aplicación web y Reportes

2.3.4 Diseño de Tablas

En esta sección se muestra el diagrama del modelo de datos:

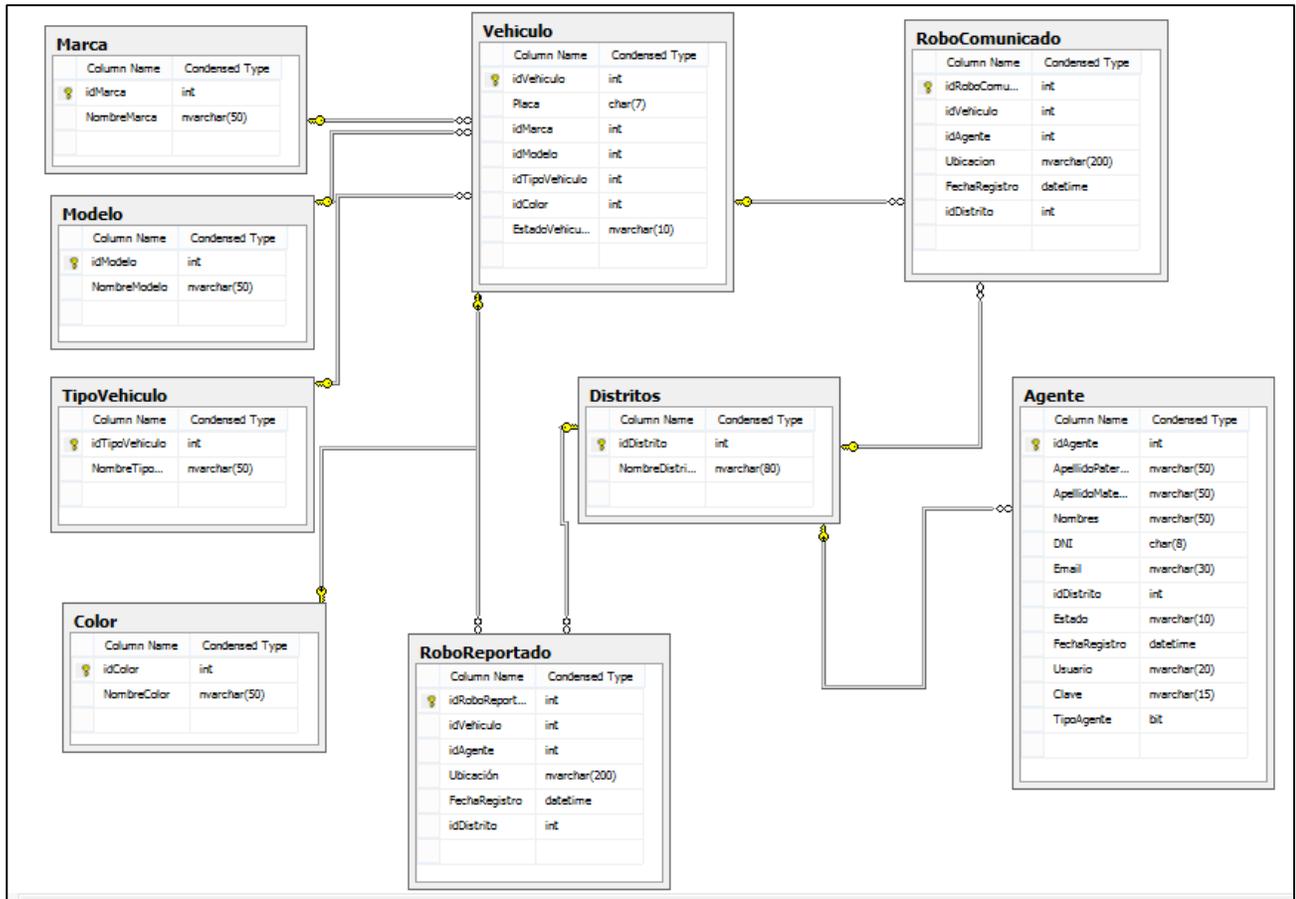


Figura N° 21 - Modelo de Datos
Fuente: Elaboración Propia

2.3.5 Diccionario de tablas

A continuación se describen las tablas implicadas en el desarrollo de la solución:

Tablas de Agente Ciudadano	
Columna	Descripción
Agente	Tabla que almacena los datos básicos de los agentes, como Nombre, Apellido, DNI.
Color	Tabla que almacena el universo de colores.
Distritos	Tabla que almacena los distritos de lima.
Marca	Tabla donde se registra el universo de las marcas de vehículos.
Modelo	Tabla donde se registran los modelos de las marcas previamente registradas.
Robo Comunicado	Tabla donde se almacena la información brindada por el agente ciudadano de las características del vehículo que ha sido siniestrado.
Robo Reportado	Tabla donde se almacena la información consultada por el agente ciudadano sobre los vehículos robados previamente registrados en el sistema de la Policía.
Vehículo	Tabla donde se almacena los datos básicos del vehículo robado.
Tipo de Vehículo	Tabla que almacena los diferentes tipos de vehículos, ejm. Sedan, Camioneta, Station Wagon.

Tabla N° 18 - Tablas Agente Ciudadano
Fuente: Elaboración Propia

2.3.6 Diccionario de datos

En esta sección el diccionario de datos, el cual agrupado por tablas, lista y detalla cada uno de los campos considerados para la solución.

Agente		
Columna	Tipo	Descripción
idAgente	Int	Identificador del Agente Ciudadano, es un número incremental. Campo Autogenerado.
Apellido Paterno	Nvarchar(50)	Apellido Paterno del Agente Ciudadano registrado en el sistema. Campo Obligatorio.
Apellido Materno	Nvarchar(50)	Apellido Materno del Agente Ciudadano registrado en el sistema. Campo Obligatorio.
Nombres	Nvarchar(50)	Nombres con los cuales se va a registrar el usuario para el uso de la aplicación. Campo Obligatorio.
DNI	Nchar(8)	DNI del agente, al ser único se toma como el identificador del agente. Campo Obligatorio.
Email	Nvarchar(30)	Correo del usuario necesario para realizar el registro de la aplicación. Campo Obligatorio.
idDistrito	Int	Identificador del Distrito donde reside el agente ciudadano. Campo obligatorio.
Estado	Nvarchar(20)	Estado de la cuenta del Agente Ciudadano. Estados: Activos, Inactivo, Bloqueado.
FechaHoraRegistro	Datetime	Fecha que el agente entra por primera vez a la aplicación y realiza el registro de

Agente		
Columna	Tipo	Descripción
		su usuario.
Usuario	Nvarchar(20)	Usuario con el que se registrará el Agente Ciudadano para hacer uso de la aplicación Web. Campo Obligatorio.
Clave	Nvarchar(15)	Clave ligada al usuario con la cual se podrá acceder a la aplicación Web. Campo Obligatorio.
TipoAgente	Bit	Flag que identificar si es Agente Ciudadano registrado es una personal natura o un miembro de la Policía.

Tabla N° 19 - Tabla Agente
Fuente: Elaboración Propia

Color		
Columna	Tipo	Descripción
idColor	Int	Identificador de la tabla Color, el cual tiene el detalle de los diversos colores de autos.
NombreColor	Nvarchar(50)	Detalle de los colores para un vehículo.

Tabla N° 20 - Tabla Color
Fuente: Elaboración Propia

Distrito		
Columna	Tipo	Descripción
idDistrito	Int	Identificador de los Distritos de Lima. Es incremental.
NombreDistrito	Nvarchar(80)	Lista de los diversos distritos de Lima Metropolitana.

Tabla N° 21 - Tabla Distrito
Fuente: Elaboración Propia

Marca		
Columna	Tipo	Descripción
idMarca	Int	Identificador de la Marca del Vehículo.
NombreMarca	Nvarchar(50)	Nombre de las diversas marcas de vehículos.

Tabla N° 22 - Tabla Marca
Fuente: Elaboración Propia

Modelo		
Columna	Tipo	Descripción
idModelo	Int	Identificador del Modelo, según la marca del vehículo.
NombreModelo	Nvarchar(50)	Nombre de los diversos modelos de vehículos, asociados a las Marcas.

Tabla N° 23 - Tabla Modelo
Fuente: Elaboración Propia

RoboReportado		
Columna	Tipo	Descripción
idRoboReportado	Int	Identificador del Reporte generado a través de la aplicación móvil Agente Ciudadano.
idVehiculo	Int	Identificador de los vehículos registrados, es un número incremental. Campo Autogenerado.
idAgente	Int	Identificador del Agente Ciudadano, es un número incremental. Campo Autogenerado.

Ubicación	Nvarchar(200)	DNI del agente quien realiza la consulta si el vehículo es robado o no.
FechaHoraRegistro	Datetime	Fecha y Hora en que se registró el vehículo en el sistema,
idDistrito	Int	Identificador de los Distritos de Lima. Es incremental.

Tabla N° 24 - Tabla RoboReportado
Fuente: Elaboración Propia

TipoVehiculo		
Columna	Tipo	Descripción
idTipoVehiculo	Int	Identificador del tipo del vehículo.
NombreTipoVehiculo	Nvarchar(50)	Detalle de los diversos tipos de vehículos. Ej. Van, Station Wagon, Sedan, etc.

Tabla N° 25 - Tabla TipoVehiculo
Fuente: Elaboración Propia

Vehículo		
Columna	Tipo	Descripción
idVehiculo	Int	Identificador de los vehículos registrados, es un número incremental. Campo Autogenerado.
Placa	Char(7)	Placa del vehículo. Campo Obligatorio.
idMarca	Int	Identificador de la Marca del Vehículo.
idModelo	Int	Identificador del Modelo, según la marca del vehículo.
idTipoVehiculo	Int	Identificador del tipo del vehículo. Ej.: Camioneta, Van, Station, Sedan,
idColor	Int	Identificador de la tabla Color, el cual tiene el detalle de los diversos colores de autos.

EstadoVehiculo	Nvarchar(20)	Estado del Vehículo, dato brindado por los sistemas de la Policía. Ej.: Robado, No Robado.
----------------	--------------	--

Tabla N° 26 - Tabla Vehículo

Fuente: Elaboración Propia

VehiculoRobado		
Columna	Tipo	Descripción
idRoboComunicado	Int	Identificador del Reporte de Robo Comunicado generado a través del aplicación móvil Agente Ciudadano.
idVehiculo	Int	Identificador de los vehículos registrados, es un número incremental. Campo Autogenerado.
idAgente	Int	Identificador del Agente Ciudadano, es un número incremental. Campo Autogenerado.
Placa	Char(7)	Placa del vehículo. Campo Obligatorio.
idMarca	Int	Identificador de la Marca del Vehículo.
idModelo	Int	Identificador del Modelo, según la marca del vehículo.
idTipoVehiculo	Int	Identificador del tipo del vehículo. Ej.: Camioneta, Van, Station, Sedan,
idColor	Int	Identificador de la tabla Color, el cual tiene el detalle de los diversos colores de autos.
Característica	Nvarchar(50)	Característica especial del vehículo robado que brindará el agente para que pueda ser reconocido.
Ubicación	Nvarchar(200)	DNI del agente quien realiza la consulta

		si el vehículo es robado o no.
FechaHoraRegistro	Datetime	Fecha y Hora en que se registró el vehículo en el sistema,
idDistrito	Int	Identificador de los Distritos de Lima. Es incremental.

Tabla N° 27 - Tabla VehiculoRobado

Fuente: Elaboración Propia

2.3.7 Arquitectura del sistema

En esta sección se muestra la arquitectura para el desarrollo del proyecto Agente Ciudadano:



Figura N° 22 – Arquitectura del Sistema

Fuente: Elaboración Propia

Nota: El SQLite es una base de datos que fue usada de soporte para la construcción en el Android Studio la cual fue reemplaza posteriormente por la Base de Datos MySQL para la integración de la aplicación móvil y su página web.

CAPÍTULO III. PRUEBAS Y RESULTADOS

En el presente capítulo se muestran las pruebas y el resultado de pruebas ejecutados para asegurar la calidad del producto “Agente Ciudadano”.

3.1 Casos de Pruebas

La siguiente tabla muestra la lista de casos de pruebas ejecutados y las historias de usuarios (HU) que aplican.

#	CASO DE PRUEBA	DESCRIPCION	HU
1	Aceptar Términos y Condiciones	Verifica que el usuario pueda aceptar y o rechazar los términos y condiciones de la aplicación móvil.	HU1, HU6
2	Registrar Agente	Verifica que el usuario pueda registrar sus datos. Al concluir el sistema muestre un mensaje de bienvenida y el menú principal.	HU2
3	Comunicar Robo	Verifica que el usuario pueda ingresar un robo.	HU3
4	Actualizar datos	Verifica que el usuario pueda actualizar sus datos personales.	HU2
5	Consultar Placa	Verifica que el usuario pueda realizar la consulta de una placa.	HU8, HU9
6	Reportar Vehículo	Verifica que el usuario pueda reportar la ubicación específica de un vehículo.	HU7, HU14
7	Ingresar a la Página web	Verifica que el usuario pueda ingresar a la página web y consultar los reportes disponibles.	HU4, HU12
8	Consultar	Verifica que el usuario pueda realizar	HU5,

	Reporte	la consulta de los reportes y el sistema muestre el resultado.	HU10, HU11
9	Exportar reporte	Verifica que el usuario pueda exportar los reportes consultados.	HU13

Tabla N° 28 – Listado de Casos de Prueba

Fuente: Elaboración Propia

Ver Anexo 6 – Detalle de casos de prueba

3.2 Resultado de pruebas

Al culminar las pruebas y subsanar todas las observaciones presentadas durante el proceso de pruebas, se puede concluir que la aplicación Agente Ciudadano cumple con los requerimientos previamente identificados. Las pruebas se realizaron en equipos con sistema operativo Android con versiones igual o mayores a la versión Android 4.1.2:

- Tablet - Samsung – Modelo: Galaxy Tab II - v4.1.2
- Tablet - Samsung – Modelo: Galaxy Tab III - v4.2.2
- Smartphone – Samsung – Modelo: Galaxy 3 – v4.3

Ver Anexo 7– Resultado de Pruebas



Figura N° 23 –Pruebas: Muestra 1
Fuente: Elaboración Propia

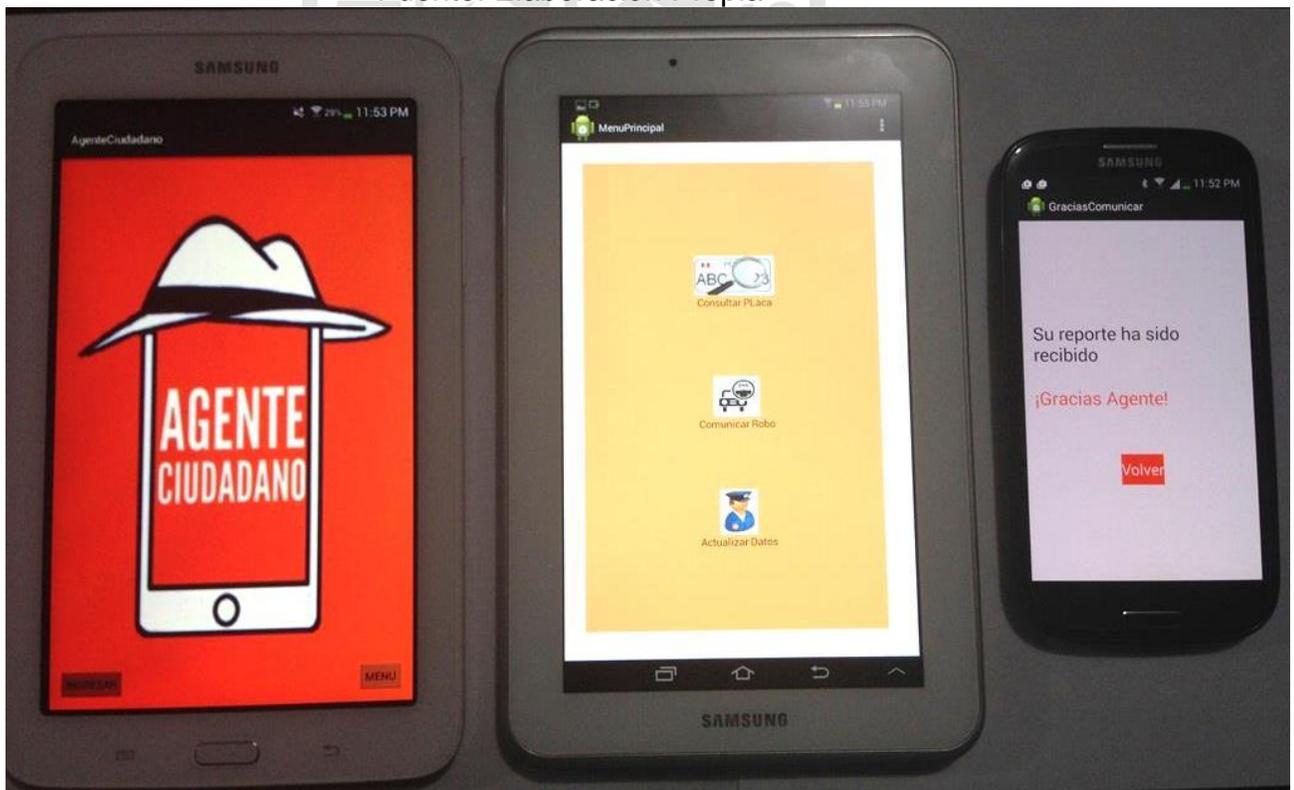


Figura N° 24 –Pruebas: Muestra 2
Fuente: Elaboración Propia

CAPÍTULO IV. DISCUSIÓN Y APLICACIONES

4.1 Diferenciación de la aplicación Agente Ciudadano

Habiendo analizado las aplicaciones existentes con las que cuenta las municipalidades, las cuales fueron presentadas como casos de éxito en el capítulo I; podemos señalar que la aplicación Agente Ciudadano se diferencia de estas en:

- Es una aplicación especializada en el reporte de ubicaciones de vehículos robados.
- No compromete la integridad personal de los usuarios ya que la comunicación de la información es de manera indirecta y privada.
- Confirma el estado de un vehículo sospechoso, consultando la placa.
- La aplicación móvil no requiere un conocimiento adicional para consultar la placa de un vehículo y reportar su ubicación.

A continuación se muestra un cuadro que muestra las semejanzas y diferencias entre las funcionalidades de la aplicación móvil Agente Ciudadano y otras aplicaciones (casos de éxito presentados en el Capítulo anterior).

Dónde:

(-) Significa que no cuenta con la funcionalidad

(✓) Significa que si cuenta con la funcionalidad



FUNCIONALIDADES	OTRAS APLICACIONES	AGENTE CIUDADANO
Bienvenida personalizada	—	✓
Consultar estado de placa	—	✓
Muestra reporte de datos ingresados	—	✓
Reporta ubicación de Vehículos	—	✓
Registrar robo	—	✓
Solicitar datos personales	✓	✓
Solicitar DNI	✓	✓
Términos y condiciones	✓	✓

Tabla N° 29 – Comparaciones y semejanzas entre otras aplicaciones y el Agente Ciudadano

Fuente: Elaboración Propia

De la tabla mostrada podemos identificar claramente que el resto de aplicaciones no cuentan con la mayoría de aplicaciones que ya se encuentran a disposición del usuario. Donde el resto de aplicaciones son los casos de éxito mencionados en el Capítulo I:

- Alerta Surco
- Bellavista emergencia
- JM emergencias
- Lince Apps
- Miraflores en tu corazón
- SOS alto al crimen

➤ Vecino Vigilante

a. Semejanzas

La mayoría de aplicaciones cuentan muestran los términos y condiciones de uso de la aplicación, solicitan datos personales durante la instalación de la aplicación y solicitan DNI como dato relevante para proceder con el uso de la aplicación.

b. Diferencias

La aplicación móvil cuenta con funcionalidades, que a diferencia del resto de aplicación, muestra una bienvenida personalizada indicando el nombre del usuario, permite consultar la placa en la misma aplicación, permite comunicar robos de manera extraoficial, permite reportar la ubicación de un vehículo robado, sin comprometer la integridad personal del usuario y muestra reportes de los datos ingresados a través del móvil en una página web.

4.2 Elección del sistema operativo

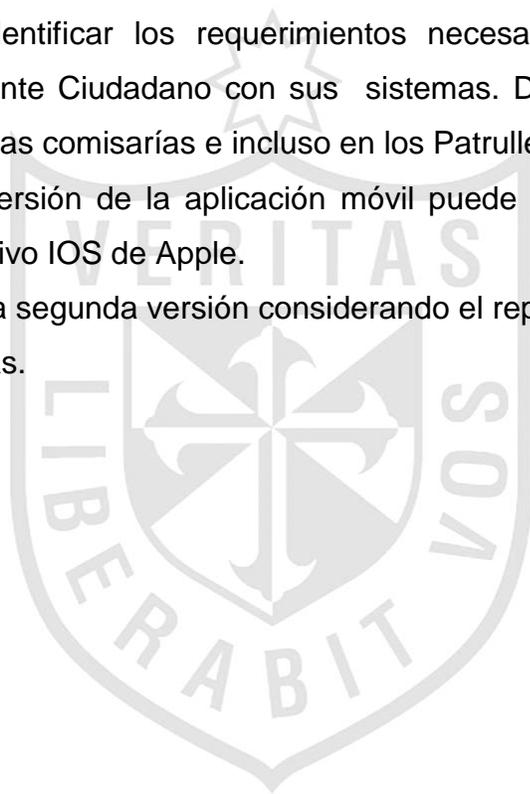
La aplicación móvil Agente Ciudadano ha sido desarrollada para uso de la comunidad, por tanto es necesario se encuentre a su alcance. La aplicación móvil se desarrolló en base al sistema operativo Android, dado que este sistema operativo se viene usando en el Perú de manera masiva e incremental durante los últimos años, como bien lo expone StatCounter (s/f) en sus datos estadísticos realizados en el Perú en cuanto al uso de sistemas operativos.

CONCLUSIONES

1. Se desarrolló una aplicación móvil que almacena los datos personales del usuario, la cual no es compartida, por lo tanto se concluye que no compromete la integridad del usuario.
2. La aplicación móvil cuenta con opciones que permiten consultar si un vehículo sospechoso es robado, de ser robado permite reportar la ubicación del vehículo.
3. La aplicación estará a disposición de la Policía, con la finalidad de que puedan contar con mayor información sobre el paradero de un vehículo robado.
4. La aplicación móvil Agente Ciudadano se complementa con una página web, que permitirá centralizar la información de los vehículos robados por distrito.
5. La aplicación móvil Agente Ciudadano permite involucrar a la ciudadanía en la lucha contra los robos de vehículos, convirtiéndola en un ente activo para la lucha contra la delincuencia.

RECOMENDACIONES

1. Integrar la aplicación móvil Agente Ciudadano con el sistema del Reniec, para validar los datos del agente.
2. Usar la aplicación móvil Agente Ciudadano como fuente oficial en el proceso de investigación realizada por la Policía, en la detección de un vehículo robado.
3. Coordinar con la Policía un estudio de la arquitectura de sistemas actual, para poder identificar los requerimientos necesarios para integrar la aplicación Agente Ciudadano con sus sistemas. De manera que pueda ser usado por las comisarías e incluso en los Patrulleros Inteligentes.
4. La siguiente versión de la aplicación móvil puede ser desplegada en el sistema operativo IOS de Apple.
5. Desarrollar una segunda versión considerando el reporte de vehículos con placas clonadas.



ANEXOS

<i>ANEXO 1. Promedio de Instalaciones de aplicaciones móviles realizadas a noviembre 2014</i>	<i>97</i>
<i>ANEXO 2. Estado de Ganancias y pérdidas</i>	<i>98</i>
<i>ANEXO 3. Diseño de Prototipos - Registro de Usuario y Comunicar Robo</i>	<i>99</i>
<i>ANEXO 4. Diseño de Prototipos - Placa y GPS</i>	<i>100</i>
<i>ANEXO 5. Diseño de Prototipos - Aplicación Web y Reportes</i>	<i>100</i>
<i>ANEXO 6. Detalle de casos de prueba</i>	<i>102</i>
<i>ANEXO 7. Resultado de Pruebas</i>	<i>107</i>

ANEXO 1. Promedio de Instalaciones de aplicaciones móviles realizadas a noviembre 2014

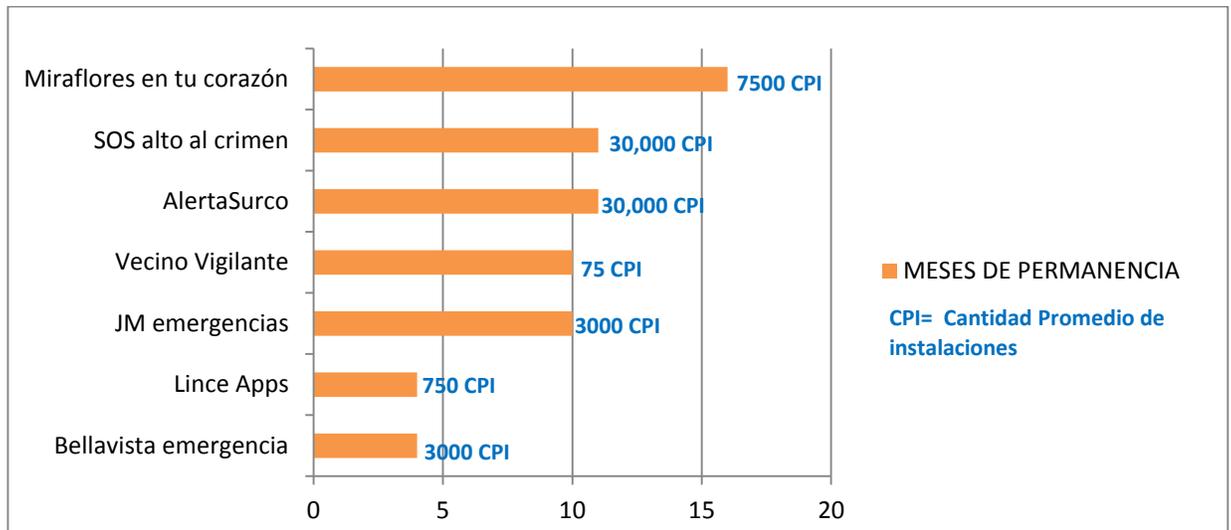


Figura N° 25 –Promedio de Instalaciones de aplicaciones móviles realizadas a noviembre 2014

Fuente: Municipalidad de Bellavista, Municipalidad de Lince, Municipalidad de Jesús María, Municipalidad de Miraflores, Municipalidad de Santiago de Surco y Municipalidad de La Molina (2014).

#	APLICACIONES MÓVILES	MES DE CREACIÓN	MESES A NOV 2014	INSTALACIONES EFECTUADAS	CANTIDAD PROMEDIO DE INSTALACIONES (PI)
1	Alerta Surco	dic-13	11	10,000 a 50,000	30,000
2	Bellavista emergencia	jul-14	4	1,000 a 5,000	3000
3	JM emergencias	ene-14	10	1,000 a 5,000	3000
4	Lince Apps	jul-14	4	500 a 1,000	750
5	Miraflores en tu corazón	jul-13	16	5,000 a 10,000	7500
6	SOS alto al crimen	dic-13	11	10,000 a 50,000	30,000
7	Vecino Vigilante	ene-14	10	50 a 100	75

Tabla N° 30 – Detalle de instalación de aplicaciones móviles

Fuente: Municipalidad de Bellavista, Municipalidad de Lince, Municipalidad de Jesús María, Municipalidad de Miraflores, Municipalidad de Santiago de Surco y Municipalidad de La Molina (2014).

ANEXO 2. Estado de Ganancias y pérdidas

Agente Ciudadano
Estado de Ganancias y Pérdidas
Expresado en S/.

	Año 1	Año 2	Año 3
Ingresos	28,800	34,560	42,768
Costos	0	0	0
Utilidad Bruta	28,800	34,560	42,768
Gastos			
- Gastos Operativos	-4,320	-5,184	-6,415
- Gastos Administrativos	-1,440	-1,728	-2,138
- Depreciación	-2,000	-2,000	-2,000
Utilidad Operativa	21,040	25,648	32,214
Gastos Financieros	2,104	2,565	3,221
Utilidad Antes de IR	23,144	28,213	35,436
Impuesto a la Renta	-6,943	-8,464	-10,631
Utilidad Neta	16,201	19,749	24,805

Activos Fijos	Costo	Vida Util(Anual)	Depreciac. Anual
Muebles	2000	3	667
Equipos	4000	3	1,333
	6,000		2,000

Tabla N° 31 – Cuadro de Gastos y Pérdidas
Fuente: Elaboración Propia

ANEXO 3. Diseño de Prototipos - Registro de Usuario y Comunicar Robo

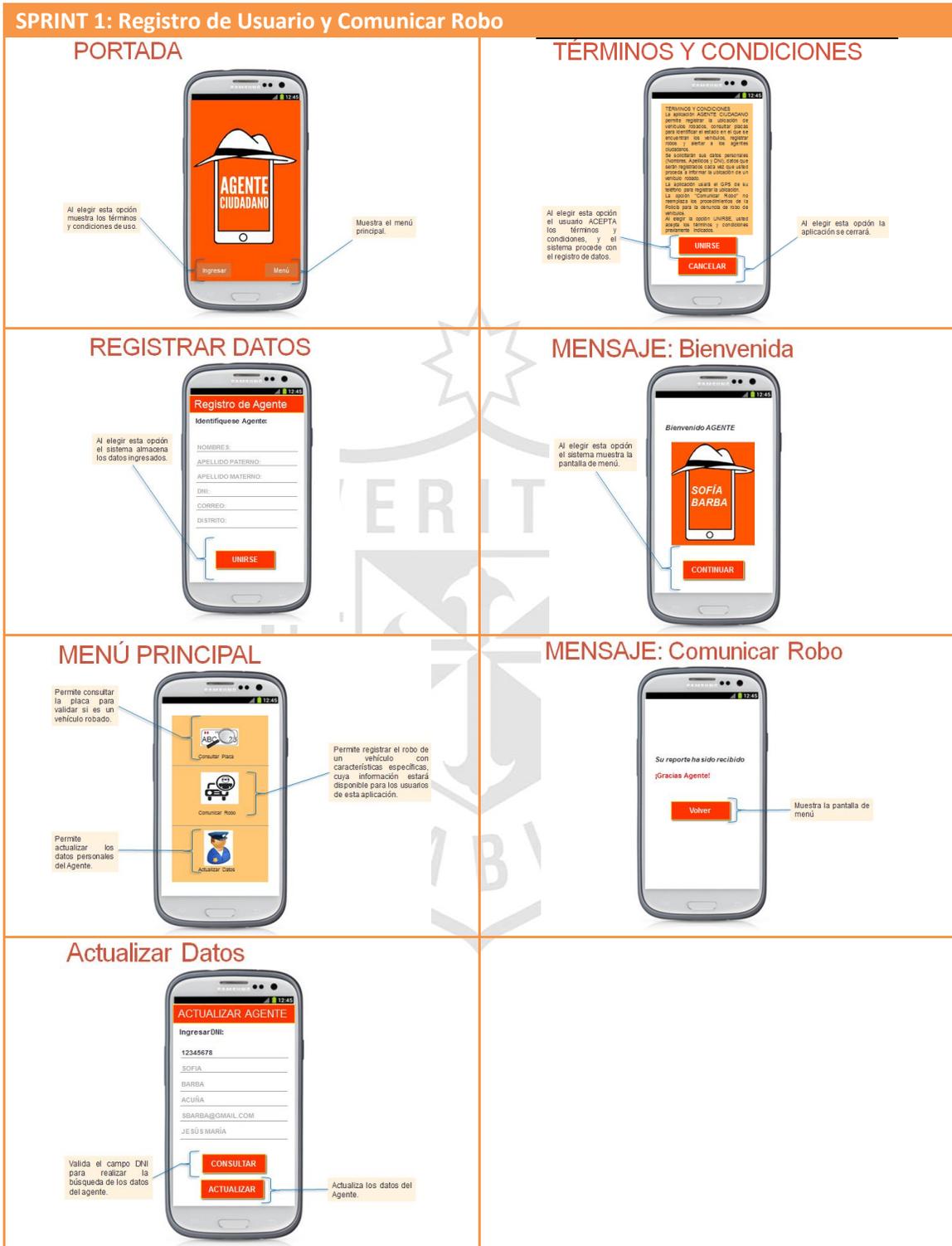


Tabla N° 32 – Prototipos SPRINT 1

Fuente: Elaboración Propia

ANEXO 4. Diseño de Prototipos - Placa y GPS

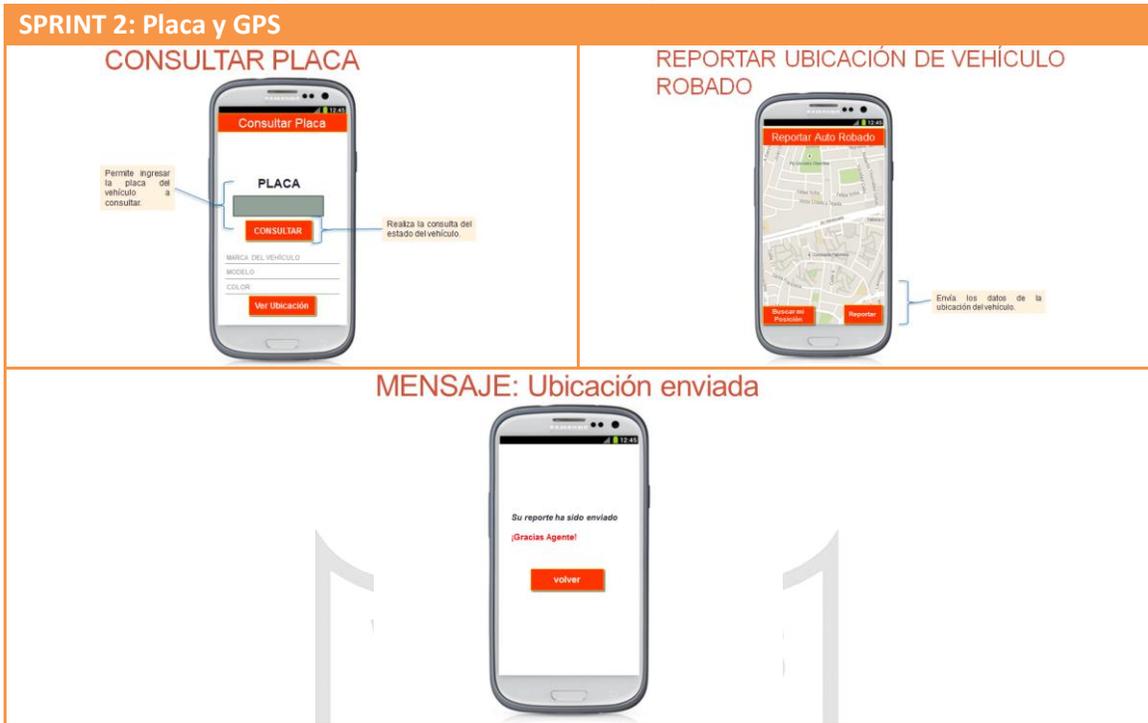


Tabla N° 33 – Prototipos SPRINT 2

Fuente: Elaboración Propia

ANEXO 5. Diseño de Prototipos - Aplicación Web y Reportes



SPRINT 3: Página Web y Reportes

[Inicio Sesión](#)

[Página Principal](#) [Acerca de](#)

Iniciar Sesión

Especifique su nombre de usuario y contraseña.

Identifíquese Agente

Nombre de Usuario:

Contraseña:

INGRESAR



Valida los datos y muestra la pantalla para seleccionar reporte

[Inicio Sesión](#)

[Página Principal](#) [Acerca de](#) Bienvenido Agente Ciudadano, Sofía Barba



Seleccione su consulta:

- Robos Comunicados por los Agentes
- Ubicaciones Reportadas

CONSULTAR

Muestra el nombre y apellido del usuario

Muestra el reporte seleccionado

[Inicio Sesión](#)

[Página Principal](#) [Acerca de](#) Bienvenido Agente Ciudadano, Sofía Barba

Placa:

Agente Ciudadano:

Fecha  **CONSULTAR**

Muestra el reporte según los filtros

#	PLACA	AGENTE CIUDADANO	FECHA Y HORA	UBICACIÓN
1	C2P315	45654987	09/09/2014 20:11:01	
2				
3				
4				

< < 1-1 > >

Exporta el reporte en Excel

EXPORTAR

SPRINT 3: Página Web y Reportes

Tabla N° 34 – Prototipos SPRINT 3

Fuente: Elaboración Propia

ANEXO 6. Detalle de casos de prueba

En esta sección se muestra las funcionalidades probadas por caso de uso:

CP1: Aceptar Términos y Condiciones		
#	Funcionalidad	Resultado esperado
1	Mostrar logo al ingresar a la aplicación	Debe mostrarse el logo en toda la pantalla del móvil.
2	Términos y condiciones	Debe mostrar los términos y condiciones de forma clara y sencilla para el usuario. Debe mostrarse 2 botones: UNIRSE: Permitirá ingresar a la pantalla para registrar nuevo Agente. CANCELAR: Cerrará la aplicación.

Tabla N° 35 – CP1: Aceptar Términos y Condiciones

Fuente: Elaboración Propia

CP2: Registrar Agente		
#	Funcionalidad	Resultado esperado
1	Registrar datos del	Debe permitir registrar:

CP2: Registrar Agente		
#	Funcionalidad	Resultado esperado
	agente	Nombres, Apellido materno, Apellido paterno, DNI, EMAIL, Distrito. Mostrar un botón UNIRSE, que al ser usado inserte los datos en la BD.
2	Mostrar bienvenida	Al ser ingresados los datos en la BD debe mostrarse un mensaje de bienvenida con el logo del agente, el nombre y apellido.
3	Menú Principal	Deben mostrarse con gráficos las opciones del sistema: Consultar Placa, al usar esta opción debe mostrar la pantalla de Consultar Placa. Comunicar Robo, al usar esta opción debe mostrar la pantalla de Comunicar Robo. Actualizar Datos, al usar esta opción debe mostrar la pantalla de Actualizar Datos.

Tabla N° 36 – CP2: Registrar Agente

Fuente: Elaboración Propia

CP3: Comunicar Robo		
#	Funcionalidad	Resultado esperado
1	Comunicar robo	Debe permitir registrar los datos: Placa, marca, modelo, tipo, color, característica específica, distrito y fecha de robo. Mostrar un botón “comunicar” que registre los datos en la base de datos.
2	Mensaje de confirmación	Debe mostrarse un mensaje de confirmación “Su reporte ha sido recibido. ¡Gracias Agente!”, así como una opción volver que permita regresar al menú principal.

Tabla N° 37 – CP3: Comunicar Robo

Fuente: Elaboración Propia

CP4: Actualizar Datos		
#	Funcionalidad	Resultado esperado
1	Actualizar datos	Al elegir la opción actualizar agente del menú principal, debe mostrarse la pantalla Actualizar Datos con los campos editables: Nombres, Apellido Paterno, Apellido Materno, DNI, EMAIL, Distrito y el botón actualizar que permita efectivamente cambiar los datos del usuario en la BD.

Tabla N° 38 – CP4: Actualizar Datos
Fuente: Elaboración Propia

CP5: Consultar Placa		
#	Funcionalidad	Resultado esperado
1	Consultar Placa	Debe mostrarse un solo campo que permite registrar una placa a la vez, y un botón que permita consultar el estado del vehículo.
2	Resultado de búsqueda exitoso	Al usar la opción de “consultar”, y solo si el vehículo tiene ingresada una denuncia de robo; el sistema debe mostrar: Placa, Marca, Modelo y color del vehículo consultado; así como un mensaje en pantalla “Robado”. También deberá mostrarse habilitado el botón “Ver ubicación”, que al ser elegido mostrará la pantalla “Reportar Vehículo Robado”

Tabla N° 39 – CP5: Consultar Placa
Fuente: Elaboración Propia

CP6: Reportar Vehículo		
#	Funcionalidad	Resultado esperado
1	Reportar Vehículo	En la pantalla “Reportar Vehículo Robado” se mostrará en toda la pantalla el mapa de ubicación con una opción para ingresar datos adicionales y una opción reportar que grabará las coordenadas en la BD.
2	Notificaciones	El sistema debe mostrar en pantalla mensajes: <ul style="list-style-type: none"> - Cuando el auto no es un vehículo robado: “No

CP6: Reportar Vehículo		
#	Funcionalidad	Resultado esperado
		<p>requisitoriado”</p> <ul style="list-style-type: none"> - Cuando el GPS no se encuentra activo: “Por favor activar el GPS” - Cuando la ubicación ha sido correctamente enviada: “Su reporte ha sido enviado ¡Gracias Agente!”

Tabla N° 40 – CP6: Reportar Vehículo
Fuente: Elaboración Propia

CP7: Ingresar a la página Web		
#	Funcionalidad	Resultado esperado
1	Crear página web	<p>La página web debe contar con los siguiente:</p> <ul style="list-style-type: none"> - Inicio de sesión. - Pantallas que muestren las funcionalidades principales de la aplicación móvil. - Logo de Google Play para descargar la aplicación. - Recomendaciones brindadas por las DIROVE, la policía. - Enlaces de interés. - Número de emergencias de la DIROVE
2	Inicio de sesión	<p>Deber mostrar el logo de Agente Ciudadano y las opciones de logueo (nombre del usuario y contraseña), página principal, acerca de y la opción ingresar que validará el usuario y clave.</p>

Tabla N° 41 – CP7: Ingresar a la Página Web
Fuente: Elaboración Propia

CP8: Consultar Reporte		
#	Funcionalidad	Resultado esperado
1	Ingresar a bienvenida	<p>Debe mostrarse el logo de Agente Ciudadano con los nombres del usuario y una bienvenida al usuario “Bienvenido Agente Ciudadano, Nombre Apellido”. La aplicación deberá distinguir entre un usuario civil y un usuario de la policía, en caso sea un policía deberá mostrar su código en vez del nombre.</p>

CP8: Consultar Reporte		
#	Funcionalidad	Resultado esperado
2	Opciones de consulta de reporte	Deben mostrarse habilitadas 2 opciones: Robos Comunicados por los Agentes, Robos Reportados a la Policía. Las opciones deben ser radio button y solo debe permitir elegir una a la vez. Mostrar un botón consultar.
3	Consultar Ubicaciones reportadas	Mostrará filtros de búsqueda por placa, agente ciudadano (DNI), fecha y un botón de búsqueda. Como resultado deberá mostrarse la placa, agente ciudadano (DNI), fecha y hora de registro y la ubicación. Mostrar habilitada una opción de exportar.
4	Consultar Robos Comunicados	Mostrará filtros de búsqueda por Distrito, Marca, Tipo, Fecha, Agente Ciudadano (DNI) y un botón de búsqueda. Como resultado deberá mostrarse la placa, marca, color, característica específica, distrito, agente ciudadano (DNI), fecha y hora de registro. Mostrar habilitada una opción de exportar.

Tabla N° 42 – CP8: Consultar Reporte
Fuente: Elaboración Propia

CP9: Exportar Reporte		
#	Funcionalidad	Resultado esperado
1	Exportar	La página web debe mostrar una opción de “Exportar” en los reportes consultados.
2	Resultado	El documento exportado debe ser tipo Excel y contar con los mismos campos mostrados en los reportes.

Tabla N° 43 – CP9: Exportar Reporte
Fuente: Elaboración Propia

ANEXO 7. Resultado de Pruebas



Figura N° 26 – Resultado de Pruebas: Portada, Términos y Condiciones
Fuente: Elaboración Propia

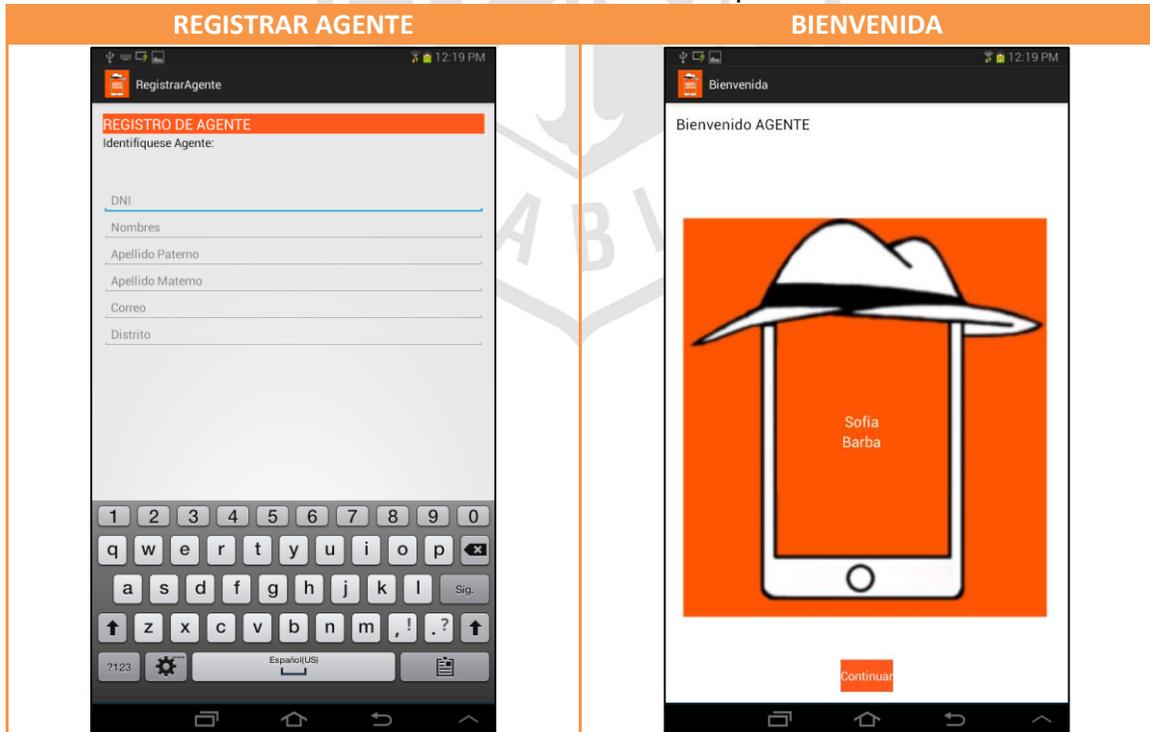


Figura N° 27 – Resultado de Pruebas: Registrar Agente y Bienvenida
Fuente: Elaboración Propia



Figura N° 28 –Resultado de Pruebas: Menú Principal y Comunicar Robo
Fuente: Elaboración Propia

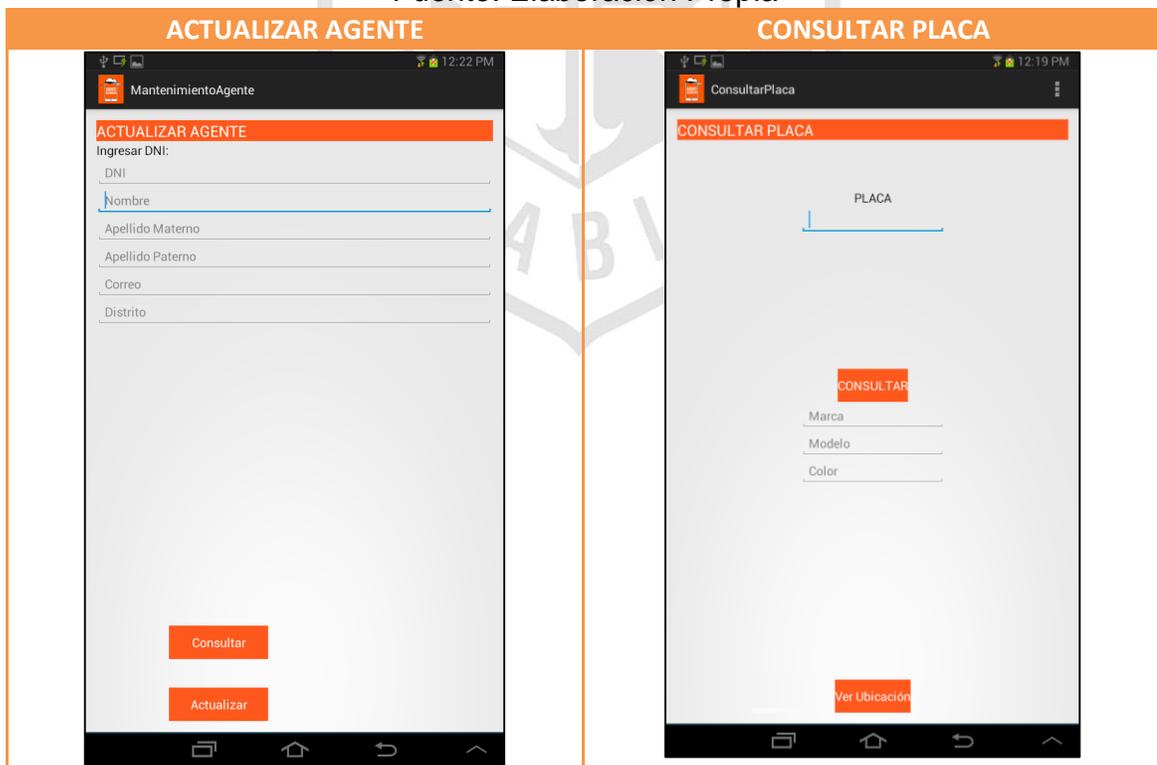


Figura N° 29 – Resultado de Pruebas: Placa y GPS
Fuente: Elaboración Propia

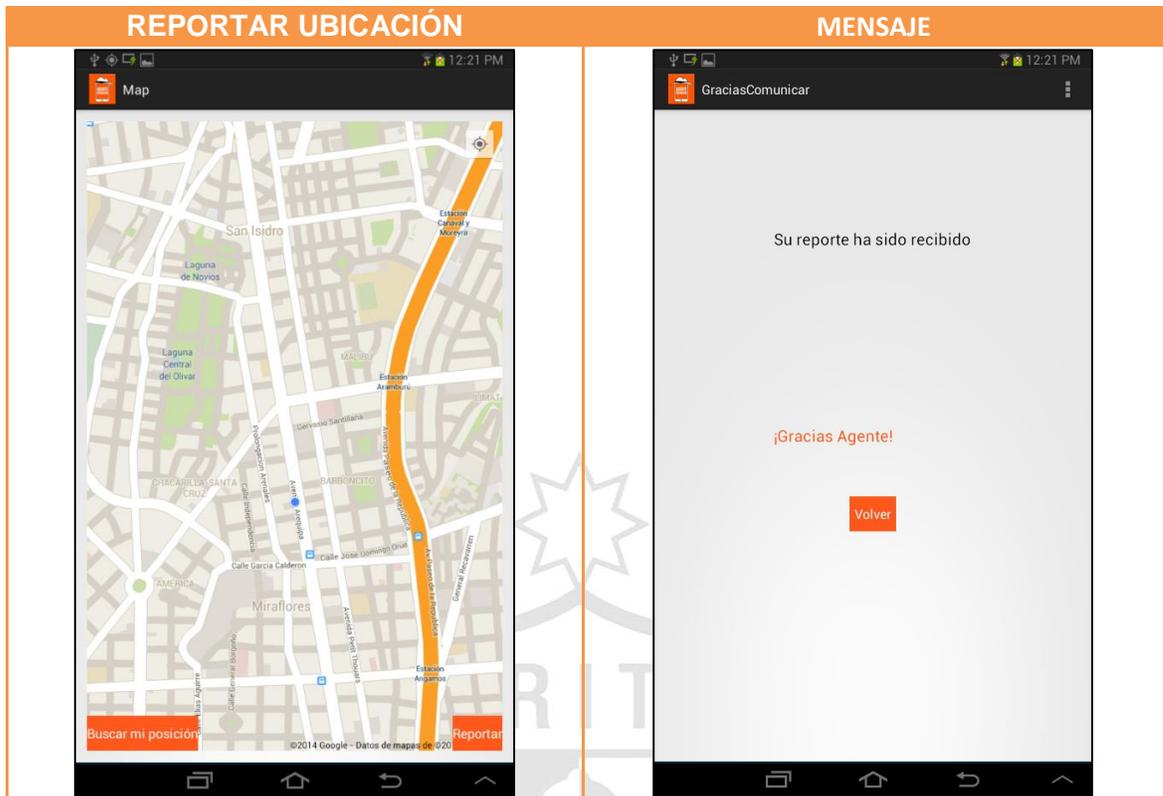
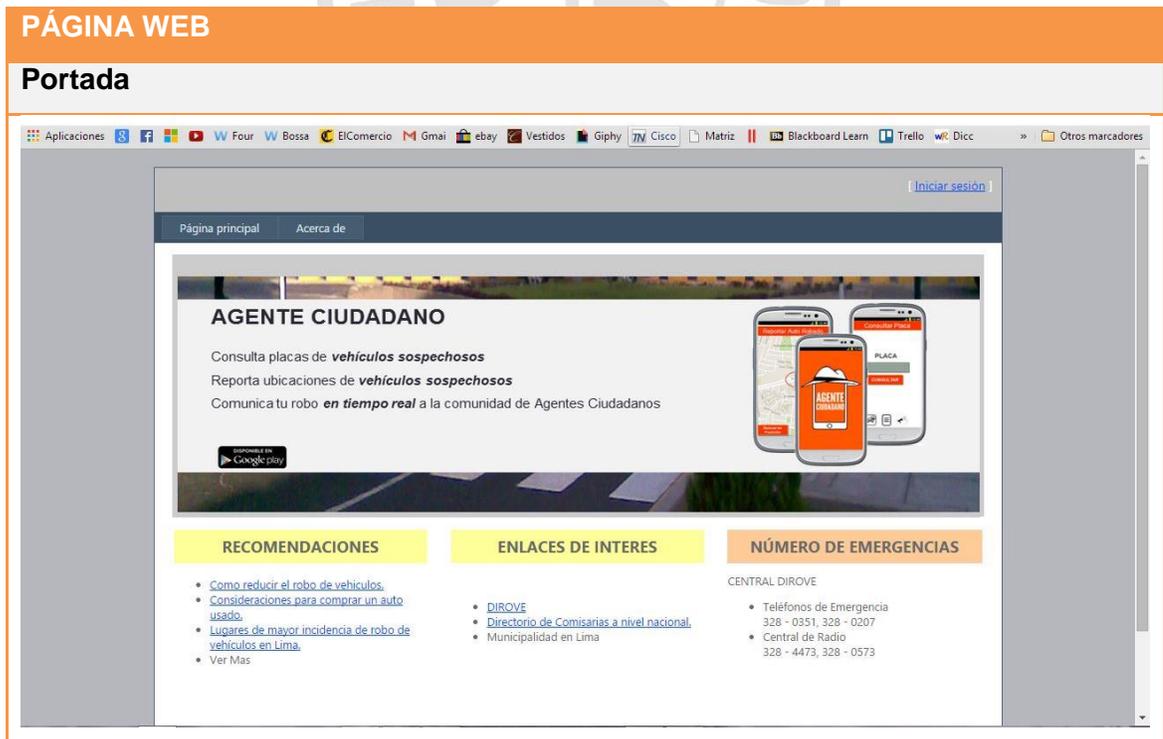


Figura N° 30 – Resultado de Pruebas: Reportar Ubicación y Mensajes
Fuente: Elaboración Propia



Inicio

Aplicaciones Facebook YouTube Four Boss ElComercio Gmail ebay Vestidos Giphy Cisco Matriz Blackboard Learn Trello Dicc Otros marcadores

[Iniciar sesión]

Página principal Acerca de

Especifique su nombre de usuario y contraseña. [Registrar](#)

Identifíquese Agente

Nombre de usuario:

Contraseña:

INGRESAR

AGENTE CIUDADANO

VERITAS

Registro de Usuario

Aplicaciones Facebook YouTube Four Boss ElComercio Gmail ebay Vestidos Giphy Cisco Matriz Blackboard Learn Trello Dicc Otros marcadores

[Iniciar sesión]

Página principal Acerca de

REGISTRESE AGENTE:
Ingrese sus datos y forme parte de la comunidad de Agentes Ciudadanos.
Las contraseñas deben tener una longitud mínima de 8 caracteres.

Agente Ciudadano:

Nombre de usuario: Debe escribir su nombre

Apellido Paterno: Debe escribir su Apellido Paterno

Apellido Materno: Debe escribir su Apellido Materno

DNI: Debe escribir su DNI

Email: Debe escribir su Email

Usuario: Debe escribir un usuario

Contraseña: Debe escribir a Contraseña

Distrito
Arequipa

Registrar

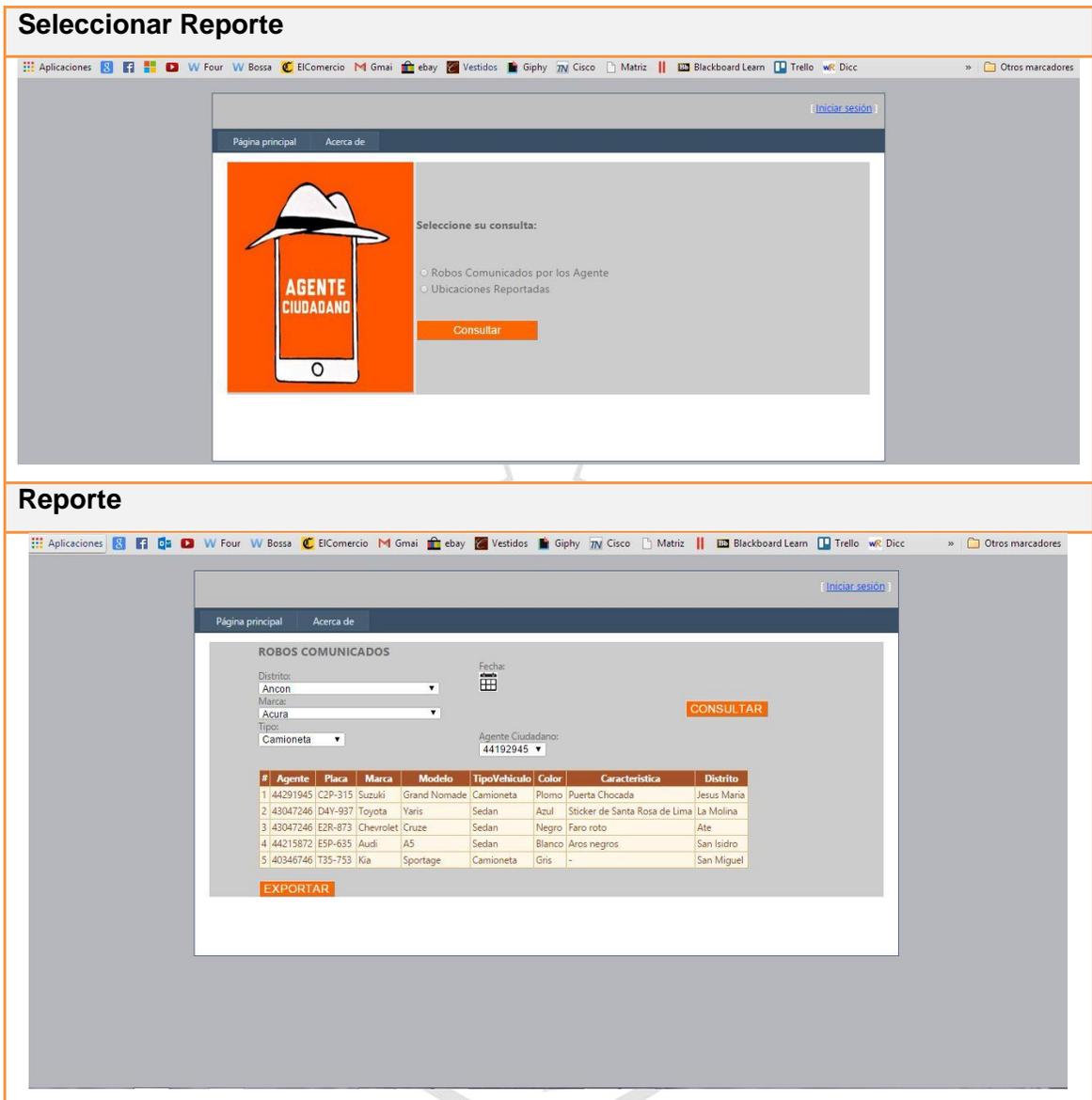


Figura N° 31 –Resultado de Pruebas: Página Web y Reportes
Fuente: Elaboración Propia

FUENTES DE INFORMACIÓN

- Android (s/f). The Android Source Code. Publicado s/f. Obtenido el 13 de Agosto de 2014, desde <https://source.android.com/source/>.
- Android Developers (s.f.). Get the Android SDK. Publicado s/f. Obtenido el 13 de Agosto de 2014, desde <http://developer.android.com/sdk/index.html>.
- Android Developers (2013). Android Studio. Publicado en Mayo 2013. Obtenido el 22 de Setiembre de 2014, desde <https://developer.android.com/sdk/installing/studio.html>
- Apache Tomcat (s/f). Apache Tomcat. Publicado s/f. Obtenido el 28 de Agosto de 2014, desde <http://tomcat.apache.org/>.
- Apple Inc.(2013). *About the iOS Technologies*. Publicado el 18 de Setiembre de 2013. Obtenido el 28 de Agosto de 2014, desde <https://developer.apple.com/library/ios/documentation/miscellaneous/conceptual/iphoneostechoverview/Introduction/Introduction.html>.
- Apple Inc. (2013). *El Sistema Operativo Móvil Más Avanzado Sigue Avanzando*. Publicado el 18 de Setiembre de 2013. Obtenido el 28 de Agosto de 2014, desde <https://www.apple.com/es/ios/ios7-update/>.
- Arroyo, Á. (2013). *Preocupante Cifra: Roban 20 Autos al Día en Lima*. Publicado el 14 de Abril de 2013. Obtenido el 13 de Setiembre de 2014, desde <http://peru21.pe/actualidad/preocupante-cifra-roban-20-autos-al-dia-lima-2126416>.
- Basterra, J., Berteza, C., Borello, A., Castillo, A., y Venturi, F. (2012). *Android OS*. Publicado el 12 de Mayo de 2012. Obtenido el 13 de Agosto de 2014, desde <http://androidos.readthedocs.org/en/latest/>.
- Blackberry (2014). *Software For Blackberry Devices*. Publicado el 12 de Enero de 2014. Obtenido el 13 de Agosto de 2014, desde <http://us.blackberry.com/software/smartphones.html>.
- Blackberry (2014). *Sistema Operativo Blackberry 10 OS*. Publicado el 12 de Enero de 2012. Obtenido el 13 de Agosto de 2014, desde <http://co.blackberry.com/software/smartphones/blackberry-10-os.html>.

- Cobol A., Gómez P., Pérez D. y Rocha R. (2005). *PHP y MySQL Tecnologías para el Desarrollo de Aplicaciones Web*. España: Díaz de Santos.
- Delgado, A. (2013). *Alerta Surco*. *Google Play*. Publicado el 16 de Diciembre de 2013. Obtenido el 13 de Agosto de 2014, desde https://play.google.com/store/apps/details?id=app.mss.surcoalerta&hl=es_419.
- DIROVE (s/f). *Nuestros Logros*. Publicado s/f. Obtenido el 15 de Agosto de 2014, desde <https://www.pnp.gob.pe/direcciones/dirove/logros.html>.
- El Comercio (2014). *Aplicaciones Móviles De Distritos Limeños: ¿Qué Ofrecen?*. Publicado el 9 de Marzo de 2014. Obtenido el 13 de Agosto de 2014, desde <http://elcomercio.pe/lima/seguridad/aplicaciones-moviles-distritos-limenos-que-ofrecen-noticia-1714679>.
- Espinoza, A. (2014). *Tres De Cada Diez Teléfonos Móviles En Perú Son Smartphones*. Publicado el 7 de Marzo de 2014. Obtenido el 13 de Agosto de 2014, desde http://elcomercio.pe/economia/peru/importacion-smartphones-peru-crecio-15-ano-noticia-1714442?ref=nota_lima&ft=mod_leatambien&e=titulo.
- Fernández, A. (2014a). *Seguridad Ciudadana: Tres Mil Alertas Al Mes Vía Aplicaciones*. Publicado el 09 de Marzo de 2014. Obtenido el 13 de Agosto de 2014, desde <http://elcomercio.pe/lima/ciudad/seguridad-ciudadana-tres-mil-alertas-al-mes-via-aplicaciones-noticia-1714678>.
- Fernández, B. (2014b). *¿Sabías Que Puedes Combatir La Inseguridad Con Tu Celular?*. Publicado el 5 de Marzo de 2014). Obtenido el 13 de Agosto de 2014, desde http://elcomercio.pe/lima/seguridad/sabias-que-puedes-combatir-inseguridad-tu-celular-noticia-1713894?ref=nota_lima&ft=mod_leatambien&e=titulo.
- Figueredo, O. B. (2006). *Sistemas Operativos para Dispositivos Móviles*. Publicado 2006 . Obtenido el 13 de Agosto de 2014, desde <http://www.negotium.org.ve/pdf/19/>.

- Gobierno de los Estados Unidos (s/f). *Sistema de Posicionamiento Global al Servicio del Mundo*. Publicado s/f. Obtenido el 22 de Agosto de 2014, desde <http://www.gps.gov/spanish.php>.
- Gobierno de los Estados Unidos (s/f), *How GPS Works*. Publicado s/f. Obtenido el 22 de Agosto de 2014, desde <http://www.gps.gov/multimedia/poster/poster-web.pdf>.
- Gutiérrez, J. (2008). *Eclipse 2.1 Java*. Publicado 2008. Obtenido el 22 de Agosto de 2014, desde http://www.uv.es/~jgutierr/MySQL_Java/TutorialEclipse.
- Hispa Linux (s/f). *¿Qué es el Software Libre?*. Publicado s/f. Obtenido el 22 de Agosto de 2014, desde <http://hispalinux.es/SoftwareLibre>.
- Huidrobo, J., y Conesa, R. (2006). *Sistemas de Telefonía*. Publicado 2006. Obtenido el 22 de Agosto de 2014, desde http://books.google.com.pe/books/about/Sistemas_de_telefon%C3%ADa.html?id=ckrl3mtUtwQC.
- IBM (s/f). *Iniciándose en la Plataforma Eclipse*. Publicado s/f. Obtenido el 22 de Agosto de 2014, desde <http://www.ibm.com/developerworks/ssa/library/os-ecov/>.
- Beck, K. A. (2004). *Extreme Programming: A gentle introduction*. Publicado el 2004. Obtenido el 22 de Agosto de 2014, desde <http://www.extremeprogramming.org/>.
- Lima Cómo Vamos (2014). *Informe de Percepción Sobre Calidad de Vida*. (Informe Número. 2014-03850) Ciudad Editorial: Lima.
- León, J. (2014). *Cada Tres Horas Se Roban Dos Autos En Lima Metropolitana*. Publicado el 22 de Abril de 2014. Obtenido el 22 de Agosto de 2014, desde <http://www.larepublica.pe/22-04-2014/cada-tres-horas-se-roban-dos-autos-en-lima-metropolitana>.
- Martínez, C. (2013). *¡Alarmante!: Un robo cada 5 minutos en Lima*. Publicado el 5 de Mayo de 2013. Obtenido el 23 de Agosto de 2014, desde <http://peru21.pe/actualidad/alarmante-robo-cada-5-minutos-lima-2129570>.

- Municipalidad de Bellavista (2014). *Google Play - Bellavista Emergencias*. Publicado el 07 de Julio de 2014. Obtenido el 23 de Agosto de 2014, desde https://play.google.com/store/apps/details?id=air.com.munibellavista.bemergencias&hl=es_419.
- Municipalidad de Lince (2014). *Google Play - Lince Apps*. Publicado el 22 de Junio de 2014. Obtenido el 23 de Agosto de 2014, desde https://play.google.com/store/apps/details?id=com.me.vistas&hl=es_419.
- Municipalidad de Jesús María (2014). *Municipalidad de Jesús María Lima - Perú*. Publicado s/f. Obtenido el 23 de Agosto de 2014, desde <http://www.munijesusmaria.gob.pe/index.php/jm-apps?highlight=WyJqbSlslmVtZXJnZW5jaWFzliwiam0gZW1lcmdlbmNpYXMiXQ==>.
- Municipalidad de Jesús María (2014). *Vecino Vigilante*. Publicado s/f. Obtenido el 23 de Agosto de 2014, desde <http://www.munijesusmaria.gob.pe/index.php/vecino-vigilante>.
- Municipalidad de Jesús María (2014). *Google Play - Vecino Vigilante*. Publicado el 14 de Enero de 2014. Obtenido el 23 de Agosto de 2014, desde https://play.google.com/store/apps/details?id=air.com.munijesusmaria.vecinovigilante&hl=es_419.
- Municipalidad de Miraflores (2014). *Google Play - Miraflores En Tu Corazón*. Publicado el 25 de Junio de 2014. Obtenido el 23 de Agosto de 2014, desde https://play.google.com/store/apps/details?id=pe.gob.miraflores.movil&hl=es_419.
- Municipalidad de Miraflores (2013). *Tecnología Al Servicio Del Vecino*. Publicado el 20 de Noviembre de 2013. Obtenido el 23 de Agosto de 2014, desde http://www.miraflores.gob.pe/_contenTempl3.asp?idcontenido=7505.

- Municipalidad de Santiago de Surco (2014). *Aplicación Alerta Surco Al Servicio De Nuestros Vecinos*. Publicado el 15 de Agosto de 2014. Obtenido el 23 de Agosto de 2014, desde http://www.munisurco.gob.pe/surco_portal/municipio/app/agosto/agosto_134_surco_alerta.html.
- Municipalidad de La Molina (2014). *S.O.S Alto Al Crimen - La Molina*. Publicado el 15 de Agosto de 2014. Obtenido el 23 de Agosto de 2014, desde <http://www.munimolina.gob.pe/index.php/descargar-sos-alto-al-crimen>.
- Municipalidad de La Molina (2014). *"Bellavista Emergencias", Novedoso Aplicativo para Móviles*. Publicado el 04 de Julio de 2014. Obtenido el 23 de Agosto de 2014, desde <http://diario16.pe/noticia/50349-bellavista-emergencias-novedoso-aplicativo-para-moviles>.
- Pavón, J. (2012). *Java EE-Apache Tomcat, Aplicaciones Web/ Sistemas Web*. Publicado 2012. Obtenido el 30 de Agosto de 2014, desde <http://www.fdi.ucm.es/profesor/jpavon/web/42-tomcat.pdf>.
- Pele System y Lola-Tech (2014). *SOS Alto al Crimen*. Publicado el 27 de Marzo de 2014. Obtenido el 30 de Agosto de 2014, desde https://play.google.com/store/apps/details?id=com.lolatech.android.crimestoppe&hl=es_419.
- Pérez, O. (2011). *Cuatro enfoques metodológicos para el desarrollo de Software RUP – MSF – XP – SCRUM*. Publicado 2011. Obtenido el 30 de Agosto de 2014, desde <https://drive.google.com/?tab=mo&authuser=0#folders/0Bzy1tGaAChahR2FrSTJacEJtYjA>.
- Ramirez, G. (2013). *Seguridad en Aplicaciones Moviles. Universidad Nacional Abierta y a Distancia*. Publicado 2013. Obtenido el 30 de Agosto de 2014, desde http://datateca.unad.edu.co/contenidos/233016/EXE_SAM/index.html.
- Rodríguez, S. (2013). *Inseguridad en Lima: Percepción vs. Realidad*. Publicado el 3 de Octubre de 2013. Obtenido el 30 de Agosto de 2014, desde

<http://publimetro.pe/actualidad/noticia-inseguridad-lima-percepcion-vs-realidad-17139?ref=ecr>.

Rospigliosi Avila, E. (2014). *Google Play - JM Emergencias*. Publicado el 13 de Enero de 2014. Obtenido el 30 de Agosto de 2014, desde https://play.google.com/store/apps/details?id=air.air.AlertaPanico&hl=es_419.

Scrum.org (s/f). *What is Scrum?*. Publicado s/f Obtenido el 30 de Agosto de 2014, desde <https://www.scrum.org/resources/what-is-scrum/>.

StatCounter (s/f). *Market Share (July 2013 to July 2014)*. Publicado s/f. Obtenido el 30 de Agosto de 2014, desde http://gs.statcounter.com/#mobile_os-PE-monthly-201307-201407-bar.

SQLite (s/f). *About SQLite*. Obtenido el 26 de Setiembre 2014, desde <http://www.sqlite.org/about.html>

UCAD (s/f). *Servidores Web*. Publicado s/f. Obtenido el 30 de Agosto de 2014, desde <http://es.scribd.com/doc/27519905/8/Caracteristicas-de-IIS>.

UNAD (s/f). *¿Qué es una Aplicación Móvil?*. Publicado s/f. Obtenido el 30 de Agosto de 2014, desde https://plus.google.com/hangouts/_/gsi3qijdwfujcatv7tdxavbkyqa?hl=es-419.