

Proyecto Final de Carrera Ingeniería Informática	Informe de estado del proyecto	FICH	UNL	
---	---------------------------------------	-------------	------------	--

REALIZADO POR	FECHA	FIRMA
Facundo Salmerón	19/03/18	
REVISADO POR	FECHA	FIRMA
Emmanuel Rojas Fredini	19/03/18	
REVISADO POR	FECHA	FIRMA
Horacio Sagardoy	19/03/18	
APROBADO POR	FECHA	FIRMA
Lucila Romero		

Nombre del Proyecto: DESARROLLO DE APLICACIÓN MÓVIL PARA EL DISEÑO Y PROCESAMIENTO DE ENCUESTAS PÚBLICAS SOBRE SALUD.

Periodo del Informe: 26/02/2018 – 14/03/2018

Alcance: Etapa 1

Proyecto Final de Carrera Ingeniería Informática	Informe de estado del proyecto	FICH	UNL	
---	--------------------------------	------	-----	--

Estado del Proyecto					
Cronograma	Etapa 1: Análisis.	Actividad	Fecha realización		<i>Resultados obtenidos</i> Se deja un anexo con un detalle de algunas de las actividades realizadas.
			Estimada	Real	
		1.1. Identificación y análisis de los requerimientos funcionales y no funcionales, con la realización de los diagramas de casos de uso.10	07/09/17	02/03/18	
		1.2. Identificación de riesgos y análisis de posibles soluciones.5	08/09/17	---	
		1.3. Estudio de plataformas de solución y ventajas de la implementación en Android.5	12/09/17	03/03/18	
		1.4. Estudio del lenguaje de programación Java y la SDK de Android.30	22/09/17	05/03/18	
		1.5. Investigación del funcionamiento de los Web Services y almacenamiento en las bases de datos.15	26/10/17	8/03/18	
		1.6. Análisis del acceso a la funcionalidad del GPS en los dispositivos Android.5	27/10/17	09/03/18	
		1.7. Desarrollo del documento de requerimientos.25	06/10/17	14/03/18	

Proyecto Final de Carrera Ingeniería Informática	Informe de estado del proyecto	FICH	UNL	
---	---------------------------------------	-------------	------------	--

Riesgos	Riesgo	Se efectivizó		Impacto	Mitigación
	R001: Falta de disponibilidad del director.		No		
	R002: Retrasos en los entregables.	Si		Debido que el ejecutor del proyecto tomó un empleo de 8 horas diarias y debido a situaciones de cursado en la carrera, el inicio del proyecto se postergó unos meses.	
	R003: Indisponibilidad de los recursos.		No		
	R006: Cambios en los requerimientos.		No		
	Riesgos futuros			Probabilidad	Impacto
	Falta de disponibilidad del recurso humano del proyecto			Alta	Medio
Notas	Como mitigación para solventar los retrasos producidos en el inicio del proyecto, se comenzó a trabajar más horas de las previstas por día y tiempo completo los fines de semana y feriados.				



UNIVERSIDAD NACIONAL DEL LITORAL
Facultad de Ingeniería y Ciencias Hídricas

PROYECTO FINAL DE CARRERA
INGENIERÍA EN INFORMÁTICA

**Desarrollo de aplicación móvil para el
diseño y procesamiento de encuestas
públicas sobre salud.**

ETAPA 1: ANÁLISIS

Alumno: Salmerón Facundo

Director: Rojas Fredini Emmanuel

Co-Director: Sagardoy Horacio

Santa Fe, Marzo de 2018

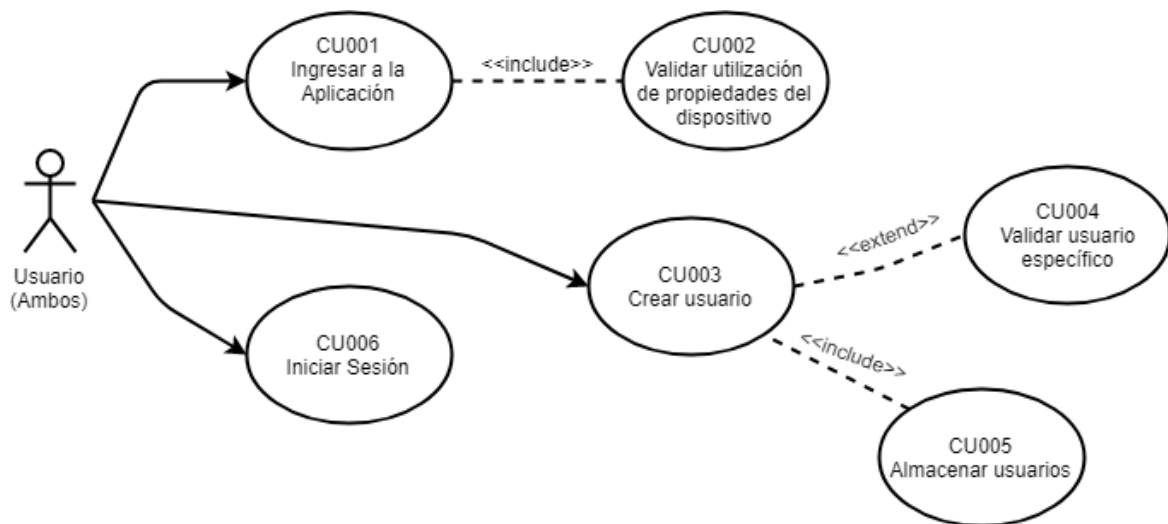
ANEXO:

En el siguiente anexo se detallan los resultados obtenidos de las tareas realizadas dentro de la primera etapa que constituye el ciclo de vida del proyecto, denominada de Análisis.

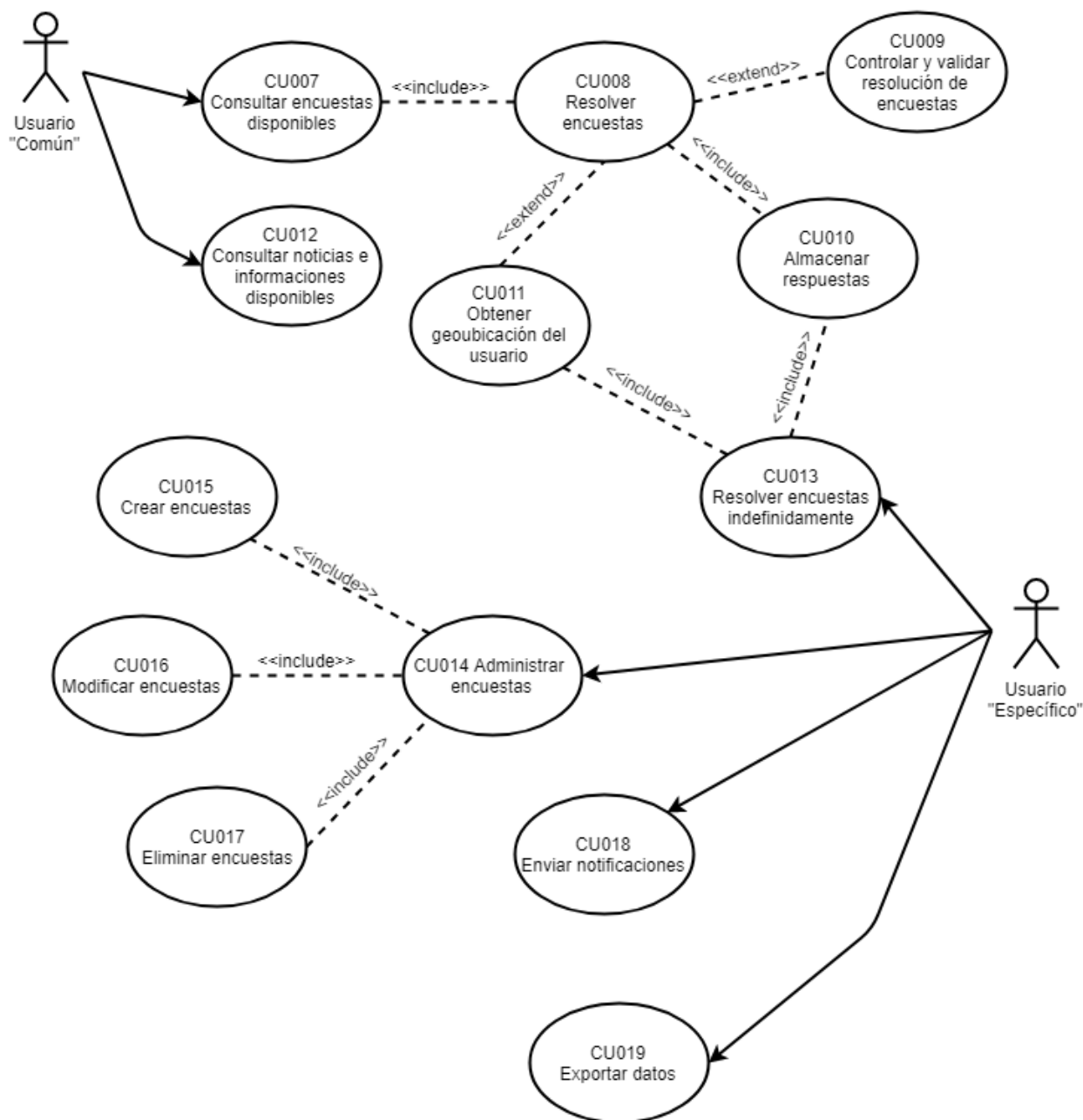
IDENTIFICACIÓN Y ANÁLISIS DE REQUERIMIENTOS FUNCIONALES Y NO FUNCIONALES DE LA APLICACIÓN:

DIAGRAMAS DE CASOS DE USO:

Mediante el diagrama de caso de uso que se presenta a continuación, se describe en forma gráfica la secuencia de operaciones que son llevadas a cabo por el sistema en respuesta a los eventos ejecutados por los distintos actores. Ésta técnica fue utilizada para la recolección de requerimientos funcionales de la aplicación, mientras que los requerimientos No Funcionales serán descriptos a continuación de los Casos de uso.



(Fig. 1 Diagrama de Casos de Usos)



(Fig. 2 Diagrama de Casos de Uso)

DEFINICIÓN DE CASOS DE USO:

Caso de uso	CU001 Ingresar a la aplicación
Actores	Cualquier tipo de usuario
Descripción	El usuario ingresa por primera vez a la aplicación de Encuestas FCM UNL presionando sobre el ícono representativo de la misma.
Precondiciones	Tener la aplicación descargada e instalada en el dispositivo.
Postcondiciones	Ejecutar validación de permisos sobre el dispositivo.
Flujo Principal	1)- El usuario descarga la aplicación del Play Store de Google. 2)- La aplicación se instala en el dispositivo. 3)- El usuario presiona sobre el ícono de la aplicación.

	4)- La aplicación se inicia por primera vez.
Flujo Alternativo	---

Caso de uso	CU002 Validar utilización de propiedades del dispositivo.
Actores	Cualquier tipo de usuario.
Descripción	Se validará que con la primer ejecución de la aplicación se otorgarán los permisos para utilizar los distintos recursos del dispositivo como por ejemplo la geoubicación mediante la utilización de los sensores móviles, datos móviles para la conectividad y almacenamiento.
Precondiciones	Iniciar la aplicación por primera vez.
Postcondiciones	El dispositivo acepta o no que la aplicación consuma sus recursos.
Flujo Principal	1)- En la interfaz gráfica aparecerá un popup informando los recursos del dispositivo a consumir por parte de la aplicación. 2)- El usuario permite que el dispositivo utilice los recursos. 3)- Se habilitan los recursos a utilizar. 4)- Se presenta la pantalla principal de la aplicación.
Flujo Alternativo	2)- a)- El usuario no permite que el dispositivo utilice los recursos. 2)- b)- Se cierra la aplicación y no dejará utilizarla correctamente hasta que no se acepte la utilización de los mismos.

Caso de uso	CU003 Crear Usuario
Actores	Cualquier tipo de usuario.
Descripción	El actor al utilizar la aplicación deberá crearse un usuario para poder utilizar las funcionalidades de la misma (luego dependiendo del tipo de usuario las funcionalidades son distintas).
Precondiciones	Iniciar la aplicación y tener los recursos habilitados.
Postcondiciones	El actor posee un usuario para realizar actividades dentro de la app.
Flujo Principal	1)- El usuario presiona sobre el botón de “Registrar nueva cuenta”. 2)- Se abre la pantalla de creación de usuario. 3)- El usuario ingresa los datos para el registro. Entre ellos el nombre de usuario, contraseña, mail, sexo y edad (datos importantes en la posterior resolución de encuestas). 4)- El usuario presiona sobre el botón “Crear cuenta”. 5)- El sistema ejecuta validaciones sobre los datos ingresados. 6)- Se informa la correcta creación del usuario. 7)- Se vuelve a la pantalla de inicio de la aplicación.
Flujo Alternativo	4)- a)- El usuario presiona sobre el botón “Cancelar”. 4)- b)- Se informa que los datos no fueron cargados. 4)- c)- Se regresa a la pantalla principal 6)- a)- El sistema informa los datos incorrectos. 6)- b)- Se vuelve a la pantalla de creación de usuario.

Caso de uso	CU004 Validar usuario específico
Actores	Cualquier tipo de usuario.
Descripción	Para la creación de usuarios específicos se deberá completar un campo extra para poder obtener todas las funcionalidades de dicho usuario.

Precondiciones	Ingresar a la creación de usuarios.
Postcondiciones	Creación de usuario específico y obtención de funcionalidades propias.
Flujo Principal	1)- El usuario coloca un código de validación en un campo no obligatorio para la registración. 2)- El sistema valida el código ingresado. 3)- El sistema permite la creación del usuario específico.
Flujo Alternativo	3)- El sistema no permite la creación del usuario específico.

Caso de uso	CU005 Almacenar Usuarios.
Actores	---
Descripción	Los datos ingresados en la registración del nuevo usuario son almacenados en la base de datos mediante la sincronización por Web Service.
Precondiciones	Los datos ingresados sean correctos, es decir hayan pasado las validaciones previas.
Postcondiciones	Los datos de los usuarios se encontrarán almacenados en la base de datos.
Flujo Principal	1)- Se comprueba la conexión a la web. 2)- Se envían los datos a la base de datos mediante el web Service. 3)- Se almacenan los datos en la base de datos.
Flujo Alternativo	2)- a)- Se informa sobre la no conexión a la web. 2)- b)- No se permite crear el usuario y por ende no se almacenan los datos.

Caso de uso	CU006 Iniciar Sesión.
Actores	Cualquier tipo de usuario.
Descripción	Se ingresarán los datos del usuario y se podrá acceder a la pantalla principal de la aplicación para poder utilizar las funcionalidades permitidas al mismo.
Precondiciones	Tener una cuenta de usuario creada.
Postcondiciones	Acceder a la pantalla principal de la aplicación.
Flujo Principal	1)- El usuario ingresa sus datos en los campos "Nombre de usuario" y "Contraseña". 2)- El usuario presiona sobre el botón "Ingresar". 3)- El sistema ejecuta validaciones sobre los datos ingresados. 4)- Se presenta la pantalla principal de la aplicación.
Flujo Alternativo	2)- El usuario sale de la aplicación. 4)- a)- Se informa al usuario que los datos ingresados son incorrectos. 4)- b)- Se limpian los campos de "Nombre de usuario" y "Contraseña" para volver a ingresarlos.

Caso de uso	CU007 Consultar encuestas disponibles.
Actores	Usuario común.
Descripción	El usuario ingresa a ver todas las encuestas disponibles para su resolución, las cuales se encuentran en forma de lista y son

	informadas mediante el título de la misma y si ya fueron resueltas o no por el usuario en cuestión.
Precondiciones	Haber iniciado sesión y tener conectividad.
Postcondiciones	Conocer las encuestas que se encuentran activas y poder resolverlas.
Flujo Principal	1)- Presionar sobre el campo “Encuestas”. 2)- Mediante Web Service traer todas aquellas encuestas activas. 3)- Se muestran las encuestas mediante su título, condición y una breve descripción de la misma.
Flujo Alternativo	3)- Se informa la falta de conectividad y la imposibilidad de cargar las encuestas.

Caso de uso	CU008 Resolver encuestas
Actores	Usuario común
Descripción	El usuario podrá escoger sobre la lista de encuestas disponibles aquella que desee responder.
Precondiciones	Tener conectividad y por lo tanto poseer la lista de encuestas disponibles.
Postcondiciones	Aportar datos a la estadística, sobre la encuesta en cuestión.
Flujo Principal	1)- El usuario presiona sobre la encuesta a resolver. 2)- El sistema valida que el usuario todavía no la haya resuelto. 3)- Se abre la encuesta a resolver. 4)- El usuario responde las preguntas. 5)- El sistema obtiene automáticamente los datos de interés del usuario. 6)- El usuario presiona sobre el botón “Enviar encuesta”. 7)- El sistema valida que los campos cumplan con las condiciones. 8)- El sistema informa la correcta resolución de la misma. 9)- Se retorna a la pantalla principal de la aplicación.
Flujo Alternativo	3)- a)- Se informa que el usuario ya respondió la encuesta anteriormente. 3)- b)- Se regresa a la pantalla de encuestas disponibles. 6)- a)- El usuario presiona sobre el botón “Cancelar”. 6)- b)- Se retorna a la pantalla de encuestas disponibles. 8)- a)- El sistema informa los campos incorrectos. 8)- b)- Se retorna a la pantalla de resolución de encuestas.

Caso de uso	CU009 Controlar y validar resolución de encuestas.
Actores	Usuario común.
Descripción	Se ejecutarán una serie de validaciones sobre la encuesta a resolver para verificar que el usuario pueda hacerla efectivamente. Por ejemplo verificar la edad de los destinatarios, la geoubicación, el sexo, etc.
Precondiciones	Tener conectividad e ingresar a resolver una determinada encuesta.
Postcondiciones	Se permite al usuario responder la encuesta.
Flujo Principal	1)- Se obtienen las precondiciones de la encuesta. 2)- Se validan las precondiciones en relación a los datos del usuario. 3)- El sistema habilita al usuario a responder la encuesta.

	4)- Se abre la pantalla de resolución de encuesta.
Flujo Alternativo	3)- a)- El sistema no habilita al usuario a responder la encuesta por no cumplir alguna precondition. 3)- b)- Se informa al usuario que no puede resolver la misma. 3)- c)- Se regresa a la pantalla de encuestas disponibles.

Caso de uso	CU010 Almacenar Respuestas.
Actores	---
Descripción	Las respuestas obtenidas por parte del usuario se almacenarán en la base de datos mediante la utilización del web Service.
Precondiciones	El usuario respondió correctamente la encuesta superando todas las validaciones previas.
Postcondiciones	Los datos de resolución son almacenados en la base de datos.
Flujo Principal	1)- Se comprueba la conexión a la web. 2)- Se envían los datos a la base de datos mediante el web Service. 3)- Se almacenan los datos en la base de datos.
Flujo Alternativo	2)- a)- Se informa sobre la no conexión a la web. 2)- b)- No se permite responder la encuesta y por lo tanto no se almacenan los datos en la base de datos.

Caso de uso	CU011 Obtener geoubicación del usuario.
Actores	---
Descripción	Al resolver la encuesta se guardará el posicionamiento del usuario para luego poder armar un mapa estadístico por parte de los profesionales a cargo de estudiar los resultados.
Precondiciones	Tener conectividad y activado el sensor de posicionamiento (GPS).
Postcondiciones	Se almacenará con los datos de resolución, la ubicación del usuario.
Flujo Principal	1)- El usuario tilda la opción que informa que se encuentra en su residencia habitual. 2)- El sistema geolocaliza la posición del usuario. 3)- Se almacena junto a los datos de la encuesta la latitud y longitud geolocalizada.
Flujo Alternativo	1)- a)- El usuario no tilda la opción que informa que se encuentra en su residencia habitual. 2)- a)- El sistema pide al usuario que ingrese la localización donde reside habitualmente. 2)- b)- El usuario ingresa las coordenadas de residencia. 3)- a)- Se informa al usuario que no se pudo acceder a su geolocalización y que verifique que el sensor se encuentre activado.

Caso de uso	CU012 Consultar noticias e informaciones disponibles.
Actores	Usuario común.
Descripción	Dentro de las funcionalidades de la aplicación el usuario podrá instruirse sobre distintas epidemiologías o noticias de interés en relación a la medicina y salud, a través de la sección "Noticias e informaciones".
Precondiciones	Tener conectividad.

Postcondiciones	Obtener información de interés sobre distintos tópicos relacionados a la medicina y la salud.
Flujo Principal	1)- Presionar sobre el campo “Noticias e informaciones”. 2)- Mediante web Service traer todas aquellas noticias e informaciones activas. 3)- Se muestran las distintas noticias a modo de lista, presentadas por un título y una breve descripción.
Flujo Alternativo	3)- Se informa la falta de conectividad y la imposibilidad de cargar las noticias.

Caso de uso	CU013 Resolver encuestas indefinidamente.
Actores	Usuario específico.
Descripción	Los usuarios específicos podrán resolver una misma encuesta indefinidamente, saltando las validaciones presentadas en el CU008 y CU009.
Precondiciones	Ser tipo de usuario específico y tener conectividad.
Postcondiciones	Aportar datos sobre la encuesta en cuestión.
Flujo Principal	1)- El usuario presiona sobre la encuesta a resolver. 2)- Se abre la encuesta a resolver. 3)- El usuario responde las preguntas. 4)- Se ingresan manualmente datos sobre la persona encuestada. 5)- El usuario presiona sobre el botón “Enviar encuesta”. 6)- El sistema valida que los campos cumplan con las condiciones. 7)- El sistema informa la correcta resolución de la misma. 8)- El sistema brinda la posibilidad de responder nuevamente la misma encuesta. 9)- El usuario presiona sobre “Resolver nuevamente”. 10)- Se vuelve al punto 2.
Flujo Alternativo	5)- a)- El usuario presiona sobre el botón “Cancelar”. 5)- b)- Se retorna a la pantalla de encuestas disponibles. 7)- a)- El sistema informa los campos incorrectos. 7)- b)- Se retorna a la pantalla de resolución de encuestas. 9)- a)- El usuario presiona sobre “Volver a pantalla inicio”. 9)- b)- Se retorna a la pantalla inicial de la aplicación.

Caso de uso	CU014 Administrar encuestas.
Actores	Usuario específico.
Descripción	Se podrá llevar a cabo una administración de las encuestas desde la misma aplicación. Ya sea para crear una nueva, modificar el contenido de una existente o dar de baja una encuesta para que no se siga respondiendo.
Precondiciones	Haber ingresado a la aplicación como usuario específico.
Postcondiciones	Llevar una administración sobre las encuestas.
Flujo Principal	1)- El usuario ingresa a la sección de administración de encuestas. 2)- Se presentan las distintas funcionalidades a llevar a cabo (ABMs de encuestas).
Flujo Alternativo	---

Caso de uso	CU015 Crear encuestas.
Actores	Usuario específico.
Descripción	Mediante el ABM de encuestas el usuario podrá dar de alta nuevas encuestas para luego poder ser resueltas por los distintos tipos de usuarios.
Precondiciones	Haber ingresado a la aplicación como usuario específico.
Postcondiciones	Nuevas encuestas almacenadas para ser resueltas.
Flujo Principal	1)- El usuario ingresa a “Crear nueva encuesta”. 2)- Se carga el formulario de alta. 3)- El usuario ingresa distintas preguntas a realizar y tipos de respuestas en cada una de ellas (por ejemplo, multiple choice, desarrollo, “si o no”, etc.). 4)- El usuario ingresa las distintas validaciones a tener en cuenta para que las mismas puedan ser resueltas (Ej. CU009). 5)- El usuario presiona sobre “Dar de alta”. 6)- El sistema valida que todos los datos ingresados sean correctos. 7)- El sistema se conecta con la base de datos mediante Web Service. 8)- Se almacena correctamente la nueva encuesta. 9)- Se retorna a la pantalla de Administración de encuestas.
Flujo Alternativo	5)- a)- El usuario presiona en “Cancelar”. 5)- b)- Se retorna a la pantalla de Administración de encuestas. 6)- a)- Se informan los datos incorrectos. 6)- b)- Se vuelve a la pantalla de Alta de encuestas. 8)- a)- Se informa la falta de conectividad y la imposibilidad de almacenar la nueva encuesta.

Caso de uso	CU016 Modificar encuestas.
Actores	Usuario específico.
Descripción	Mediante el ABM de encuestas, el usuario podrá modificar aquellas encuestas existentes, ya sea agregando o corrigiendo preguntas y tipos de respuestas, validaciones previas, etc.
Precondiciones	Haber ingresado a la aplicación como usuario específico y tener conexión para que se carguen las encuestas disponibles.
Postcondiciones	Modificar los datos de una encuesta.
Flujo Principal	1)- El usuario ingresa a “Modificar encuesta”. 2)- Se cargan la lista de encuestas activas. 3)- El usuario presiona sobre la encuesta a modificar. 4)- Se carga el formulario de modificación. 5)- El usuario corrige los datos y validaciones que considere necesario. 6)- El usuario presiona sobre “Guardar cambios”. 7)- El sistema valida que todos los datos ingresados sean correctos. 8)- El sistema se conecta con la base de datos mediante Web Service. 9)- Se actualiza correctamente la encuesta modificada. 10)- Se retorna a la pantalla de Administración de encuestas.
Flujo Alternativo	2)- a)- Se informa la falta de conectividad para cargar las encuestas. 2)- b)- Se retorna a la pantalla de administración de encuestas.

	6)- a)- El usuario presiona en “Cancelar”. 6)- b)- Se retorna a la pantalla de Administración de encuestas. 7)- a)- Se informan los datos incorrectos. 7)- b)- Se vuelve a la pantalla de Alta de encuestas. 9)- a)- Se informa la falta de conectividad y la imposibilidad de almacenar las modificaciones sobre la encuesta.
--	--

Caso de uso	CU017 Eliminar encuestas.
Actores	Usuario específico.
Descripción	Mediante el ABM de encuestas, el usuario podrá dar de baja aquellas encuestas existentes. Haciendo de esta forma que las mismas no estén más disponibles para su resolución por parte de ninguno de los usuarios.
Precondiciones	Haber ingresado a la aplicación como usuario específico y tener conexión para que se carguen las encuestas disponibles.
Postcondiciones	Eliminación de una encuesta específica.
Flujo Principal	1)- El usuario ingresa a “Eliminar encuestas”. 2)- Se cargan la lista de encuestas activas. 3)- El usuario presiona sobre la encuesta a eliminar. 4)- El sistema genera un mensaje informativo sobre la encuesta a eliminar. 5)- El usuario presiona sobre “Aceptar”. 6)- El sistema se conecta con la base de datos mediante Web Service. 7)- Se elimina correctamente la encuesta de la base de datos. 8)- Se retorna a la pantalla de Administración de encuestas.
Flujo Alternativo	2)- a)- Se informa la falta de conectividad para cargar las encuestas. 2)- b)- Se retorna a la pantalla de Administración de encuestas. 5)- a)- El usuario presiona en “Cancelar”. 5)- b)- Se retorna a la pantalla de Eliminar encuestas. 7)- a)- Se informa la falta de conectividad y la imposibilidad de almacenar las modificaciones sobre la encuesta.

Caso de uso	CU018 Enviar notificaciones
Actores	Usuario específico.
Descripción	Se podrán enviar notificaciones a todos aquellos usuarios que tengan instalada la aplicación, por ejemplo para informar sobre la disponibilidad de nuevas encuestas.
Precondiciones	Haber ingresado a la aplicación como usuario específico.
Postcondiciones	Enviar notificaciones a los usuarios activos.
Flujo Principal	1)- El usuario ingresa a la sección “Enviar notificaciones”. 2)- El sistema presenta mediante un web view la Api de Firebase. 3)- El usuario ingresa los datos de la notificación. 4)- Se presiona sobre el botón enviar notificación. 5)- La notificación es enviada a todos los usuarios activos.
Flujo Alternativo	2)- a)- Se informa la falta de conectividad para acceder al web view. 2)- b)- Se retorna a la pantalla de inicio de la aplicación.

Caso de uso	CU019 Exportar datos.
Actores	Usuario específico.
Descripción	Todos aquellos datos almacenados en la base de datos podrán ser exportados en distintos formatos (por ejemplo en PDF o Excel), para que luego distintos profesionales puedan hacer uso de los mismos de la forma que deseen.
Precondiciones	Haber ingresado a la aplicación como usuario específico y tener conectividad para acceder a la base de datos.
Postcondiciones	Archivo generado con los datos de la base de datos.
Flujo Principal	1)- El usuario ingresa a la sección exportar datos. 2)- Se informa el tipo de archivo a exportar. 3)- El usuario informa los datos que desea exportar. 4)- Se presiona sobre el botón "Exportar datos". 5)- El sistema se conecta a la base de datos mediante el web Service y genera el archivo de exportación.
Flujo Alternativo	5)- a)- El sistema informa la falta de conectividad hacia la base de datos. 5)- b)- Se retorna a la pantalla principal de la aplicación.

REQUERIMIENTOS NO FUNCIONALES:

- El dispositivo debe poseer sistema operativo Android con una versión mayor o igual a la 4.4 (KitKat).
- El dispositivo debe tener GPS, tenerlo activo y con una localización válida.
- El dispositivo debe poseer conexión a internet.
- El sistema debe presentar una interfaz de uso intuitiva y sencilla.
- El sistema debe ser capaz de procesar las transacciones en segundos.
- El sistema debe ser capaz de actuar con varios usuarios concurrentes.
- El sistema debe presentar mensajes informativos y orientados al usuario final.
- Las encuestas modificadas deben actualizarse instantáneamente en la base de datos para que todos los usuarios posean la versión correcta.
- La aplicación estará disponible en el Play Store de Google para su libre descarga.

ESPECIFICACIÓN HERRAMIENTAS DE TRABAJO:

En esta tarea se llevó a cabo una investigación acerca de las posibles herramientas de trabajos y tecnologías a utilizar para llevar a cabo el desarrollo del proyecto. En este aspecto, cabe destacar que se pueden diferenciar tres desarrollos distintos, como lo serán el Web Service, la aplicación en sí y la base de datos.

En cuanto al Web Service, el mismo será desarrollado bajo el lenguaje de programación Java. La elección del mismo se debe a que dicho lenguaje soporta todas las necesidades del proyecto y presenta una homogeneidad en el lenguaje con Android, ya que el SDK (Software Development Kit) de éste último está basado en

Java. A su vez, es uno de los lenguajes más utilizados en la actualidad y con gran cantidad de documentación, así como también existen numerosos frameworks desarrollados bajo este lenguaje que pueden ser integrados y muy útiles para el desarrollo del proyecto, como por ejemplo el de Hibernate para el mapeo entre las clases Java y la base de datos.

El IDE (entorno de desarrollo integrado) a utilizar para el desarrollo del Web Service, será el Eclipse. El mismo es una de las herramientas más utilizadas para la programación en Java y es de carácter gratuito. A su vez, por el estudio desarrollado en la carrera, se posee mayor familiaridad hacia dicho IDE, así como también sobre Java.

En cuanto a la aplicación, la misma será desarrollada en Android Studio, ya que es el IDE de desarrollo oficial para ésta tecnología. Al igual que Eclipse, es de carácter gratuito y posee gran documentación en la web, así como también manuales y comunidades activas que facilitan el desarrollo y la utilización de todas las funcionalidades ofrecidas. Dentro de las ventajas de utilizar Android Studio por sobre Eclipse para el desarrollo de la aplicación, se puede mencionar que el primer IDE es puramente de Android (creado para programar Android), trae incluido la SDK de Android y presenta constantemente actualizaciones, posee una compilación más rápida, ejecución de la aplicación en tiempo real, ya sea a través del móvil como de emuladores propios, renderizado en tiempo real, entre otras cosas.

Como fue mencionado dentro de los requerimientos no funcionales, la versión mínima a utilizar de Android para el desarrollo de la aplicación será la 4.4 (KitKat). La elección de dicha versión se debe a que las funcionalidades que presenta la misma se encuentran acorde a los alcances del proyecto, además según las estadísticas brindadas por Android Studio, aproximadamente el 90% de los usuarios activos posee versiones superiores o iguales a la mencionada, logrando un mayor alcance de los usuarios a utilizar la aplicación.

ANDROID PLATFORM VERSION	API LEVEL	CUMULATIVE DISTRIBUTION
4.0 Ice Cream Sandwich	15	
4.1 Jelly Bean	16	99,2%
4.2 Jelly Bean	17	96,0%
4.3 Jelly Bean	18	91,4%
4.4 KitKat	19	90,1%
5.0 Lollipop	21	71,3%
5.1 Lollipop	22	62,6%
6.0 Marshmallow	23	39,3%
7.0 Nougat	24	8,1%
7.1 Nougat	25	1,5%

(Fig. 3 Versiones de Android)

En cuanto a la gestión de la base de datos, se llevará a cabo bajo MySQL que es una base de datos gratuita y de las más utilizadas en la actualidad. De la misma forma que en los casos anteriores, se posee gran cantidad de documentación, lo cual es de gran ayuda para resolver cualquier inconveniente que se presente. Este gestor brinda la posibilidad de desarrollar una base de datos de tipo OLTP (On Line Transaction Processing), también llamadas bases de datos dinámicas, la cual es acorde al proyecto, ya que permite que la información sea modificada en tiempo real, pudiendo insertar, eliminar, modificar o consultar datos durante la operación del sistema, a diferencia de las bases de datos de tipo OLAP (On Line Analytical Processing) que son estáticas, es decir no se insertan, eliminan, ni modifican datos, sino que solamente se pueden realizar consultas sobre datos ya existentes.

Dentro de las ventajas a mencionar en cuanto a la elección de éste gestor por sobre SQLServer u Oracle por ejemplo (dos de los gestores de base de datos más populares), MySQL es un software totalmente open source a diferencia de los mencionados y posee gran velocidad a la hora de realizar operaciones (siendo uno de los mejores rendimientos), posee bajos costos en los requerimientos para la elaboración de la base de datos, pudiendo ejecutarse en máquinas de escasos recursos gracias a su bajo consumo y existen diversos drivers para efectuar conexiones a la misma. Pero sin dudas, una de las mayores ventajas se debe a que por su conectividad, velocidad y seguridad es altamente apropiada para acceder a bases de datos a través de internet como lo será en este proyecto.

Dentro de los frameworks basados en Java se mencionó la utilización de Hibernate. El mismo es una herramienta de mapeo objeto-relacional (ORM) que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación, ya sea mediante archivos declarativos (xml) o anotaciones en los beans de las entidades que permiten establecer estas relaciones. En otras palabras, Hibernate agiliza la relación entre la aplicación y la base de datos. Provee muchas facilidades para realizar consultas a la base de datos y crear objetos en función de ellos, así como también ejecutar todo tipo de sentencias a nivel Java. Cabe mencionar que éste framework presenta soporte para conexiones sobre gestores de bases de datos MySQL, lo cual es un punto a favor para la integración de todas las herramientas a utilizar.

INVESTIGACIÓN WEB SERVICE Y ALMACENAMIENTO EN BASE DE DATOS:

Como ya se mencionó anteriormente, la aplicación necesitará de conexión mediante datos móviles para el acceso a la base de datos. Es por ello, que al estar trabajando con una arquitectura de tipo Cliente – Servidor, en donde el intercambio de información que se produce entre la aplicación y la base de datos, se manejará mediante un Web Service a través de distintos protocolos y estándares, sin importar el lenguaje de programación en el que fueron programadas ambas partes.

Para la creación del Web Service, es fundamental definir inicialmente que arquitectura será la más apropiada para la aplicación. Es por ello que nos encontramos con las siguientes:

- SOAP (*Simple Object Access Protocol*): éste protocolo estándar define como dos objetos en diferentes procesos pueden comunicarse por medio de intercambios

de datos XML. Los servicios SOAP funcionan por lo general bajo el protocolo HTTP (más común), aunque no está limitado solamente a éste, pudiendo trabajar con protocolos FTP, POP3, TCP, entre otros. SOAP es una arquitectura muy robusta y con gran seguridad, pero a su vez bastante pesada, tanto en tamaño como en procesamiento.

- REST (*Representational State Transfer*): ésta arquitectura es más nueva en relación a la anterior. Se puede mencionar que la misma es mucho más flexible y transporta datos mediante el protocolo HTTP, pudiendo utilizar los distintos métodos que proporciona el mismo para la comunicación como el GET, POST, PUT, etc. Gracias a su flexibilidad permite transportar cualquier tipo de datos, como pueden ser XML, JSON, binarios, etc.

Por lo tanto, se estableció que el tipo de Web Service a implementar sea REST, ya que se adapta a las necesidades del proyecto y es más sencillo y rápido de implementar.

Como se mencionó, SOAP solamente permite el formato XML lo cual lo hace más pesado en relación a REST, donde en éste último se pueden enviar los datos mediante JSON (JavaScript Object Notation) por ejemplo, brindando una mayor flexibilidad y mejor performance, ya que éstos tipos de datos son mucho más livianos en peso y más rápidos para su procesamiento por no requerir de etiquetas como en XML, además de poseer mayor soporte y ser de más fácil lectura.

Una de las ventajas que posee el protocolo SOAP es su mayor seguridad para el transporte de datos, lo cual puede ser subsanado en gran parte por la arquitectura REST, mediante el protocolo de transferencia HTTPS (Protocolo seguro de transferencia de hipertexto). Como ya se mencionó anteriormente, al estar basados en este protocolo, se pueden utilizar los distintos métodos que proporciona para el intercambio de datos, como por ejemplo, las peticiones GET realizando consultas al servidor y enviando las respuestas por el mismo canal, peticiones POST para almacenamientos, PUT para actualizaciones, entre otras, que van a ser de gran utilidad a lo largo del proyecto.

INVESTIGACIÓN FUNCIONALIDAD GPS EN DISPOSITIVOS ANDROID:

En la resolución de las encuestas por parte de los usuarios, junto con esas respuestas se obtendrá la ubicación del mismo gracias al geoposicionamiento del dispositivo móvil. Esto permitirá en posteriores análisis la distribución de zonas en la ciudad, realizar mapas estadísticos y tomar medidas de acuerdo a los datos obtenidos. Es por ello, que se necesitará la utilización de componentes hardware del dispositivo móvil, como lo es el GPS.

Actualmente, prácticamente la gran mayoría de los dispositivos móviles Android vienen equipados con un GPS y por la elección mínima de versión de Android, esto no sería un impedimento a la hora de instalar la aplicación. La SDK brindada por Android facilita el acceso al GPS del dispositivo en forma sencilla, clara y con gran precisión.

Por lo tanto, a la hora de resolver encuestas como fue determinado previamente en los requerimientos, la ubicación del usuario se obtendrá automáticamente gracias a éste recurso del dispositivo, donde es necesario tenerlo activado (si no fuera así, se solicitaría al usuario que lo encienda). De tal forma, se podrá obtener la ubicación de

respuesta, almacenando la latitud y longitud del encuestado. A su vez, se brinda la posibilidad de ingresar manualmente la ubicación, en situaciones en las que el usuario no se encuentre en su residencia habitual. Aquí se ingresará la dirección habitual, para luego ser buscada con la funcionalidad del GPS y de tal forma obtener las coordenadas. La importancia de las coordenadas es fundamental, ya que con sólo tener latitud y longitud asociadas a una respuesta se pueden ubicar los puntos en el mapa donde luego en futuras investigaciones se pueden generar los mapas estadísticos y sectorizaciones gracias a las mismas.

Las ubicaciones obtenidas a su vez tendrán gran importancia a la hora de ejecutar validaciones sobre resoluciones de encuestas. Por ejemplo, cuando se desee realizar encuestas para determinadas zonas, si una ubicación no se encuentra contenida en la misma, no se le permitirá la resolución al usuario.

BIBLIOGRAFÍA:

Smith, N. "Android Studio Development Essentials". Android 6 Edition.

Besteiro M. & Rodríguez M. "Web Services".

Dept. Ciencia de la Computación e IA. "Servicios Web y SOA".

Web Services – REST vs. SOAP. Recuperado de: <http://qode.pro/blog/web-services-rest-vs-soap/>

Características y tipos de bases de datos. Recuperado de: https://www.ibm.com/developerworks/ssa/data/library/tipos_bases_de_datos/index.html