
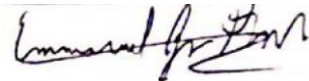



Proyecto Final de Carrera Ingeniería Informática	<b>Informe de estado del proyecto</b>	<b>FICH</b>	<b>UNL</b>	
---	---------------------------------------	-------------	------------	--

REALIZADO POR	FECHA	FIRMA
Facundo Salmerón	14/12/18	
REVISADO POR	FECHA	FIRMA
Emmanuel Rojas Fredini	05/12/18	
REVISADO POR	FECHA	FIRMA
Horacio Sagardoy	05/12/18	
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: 25/06/2018 – 15/11/2018**

**Alcance: Etapa 4 y parte de 5**

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

Estado del Proyecto					
Cronograma	Etapa 4: Desarrollo de la aplicación móvil.				Resultados obtenidos  Se deja un anexo con un detalle de las actividades realizadas.
		Actividad	Fecha realización		
			Estimada	Real	
		4.1.1. Desarrollo de las funcionalidades para el “usuario específico”.	14/02/18	22/10/18	
		4.1.2. Desarrollo de las funcionalidades para el “usuario común”.	01/03/18	01/11/18	
		4.2. Desarrollo de la interfaz gráfica.	09/03/18	25/10/18	
		4.3. Integración con el Web Service.	14/03/18	03/11/18	
	4.4. Pruebas unitarias.	18/03/18	10/11/18		
	Etapa 5: Testing.				Resultados obtenidos  Se deja un anexo con un detalle de las actividades realizadas.
		Actividad	Fecha realización		
			Estimada	Real	
		5.1. Integración de los distintos módulos.	20/03/18	03/11/18	
		5.2. Pruebas finales y experimentales sobre la aplicación en funcionamiento.	28/03/18	15/11/18	

*Resultados obtenidos*

Se deja un anexo con un detalle de las actividades realizadas.

*Resultados obtenidos*

Se deja un anexo con un detalle de las actividades realizadas.

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.	Si		Debido a cuestiones laborales tanto del director como del codirector del proyecto, se produjo un retraso en la corrección del informe de avance.	
	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 desarrollo de la aplicación se vio postergada.	
	R004: Pérdidas en la base de datos obtenida.		No		
	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					



**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 4 y 5: DESARROLLO DE LA APLICACIÓN MÓVIL Y TESTING**

Alumno: Salmerón Facundo

Director: Rojas Fredini Emmanuel

Co-Director: Sagardoy Horacio

Santa Fe, Diciembre de 2018

## **INFORME DE AVANCE Nº 4:**

En el siguiente anexo se detallan los resultados obtenidos de las tareas realizadas dentro de la cuarta etapa y parte de la quinta, que constituyen el ciclo de vida del proyecto, denominadas *Desarrollo de la aplicación móvil* y *Testing*, respectivamente.

### **DESARROLLO DE LA APLICACIÓN MÓVIL:**

Por cuestiones de funcionamiento, tanto de la aplicación en sí como del entorno de desarrollo integrado (IDE) “*Android Studio*”, las distintas tareas relacionadas al desarrollo de las funcionalidades, interfaz gráfica e integración con el Web Service, se realizaron de manera conjunta.

- **INTERFAZ GRÁFICA:**

En cuanto a la interfaz gráfica de usuario (UI), la misma fue resuelta mediante archivos XML, los cuales son un formato estándar de implementación de interfaces en Android. Los mismos se construyen usando una jerarquía de vistas (View) y grupos de vistas (ViewGroup). Las View son normalmente widgets de la UI, como pueden ser botones, campos de texto, etc. Los ViewGroup son contenedores invisibles de vistas que definen como se ordenan las vistas dentro de ellos, es decir para renderizar los distintos View de forma vertical, horizontal, alineados, etc.

Dichos archivos, denominados **Activities** o **Layouts**, son inyectados dentro de clases Java, al igual que todos sus componentes (Views), relacionando de esta forma la parte gráfica con la lógica de negocio. El IDE, presenta la posibilidad de diseñar un mismo *activity* mediante código xml o de forma gráfica, facilitando de esta forma la ubicación de los View, contraste de colores, tamaño de fuentes, etc.

- **INTEGRACIÓN CON EL WEB SERVICE:**

Para las distintas funcionalidades de la aplicación, es necesario consumir los EndPoints expuestos en el WebService (WS), presentados en el informe anterior. Para ello, se trabajó de forma local donde el WS fue configurado para trabajar en una dirección IP fija y privada, en la cual se encuentra alojado el servidor de aplicaciones. Para esto se configuró el DHCP del router para que se asigne siempre la misma IP a ese servidor, a partir de su MAC Address.

La dirección de IP configurada es 192.168.0.107, con el puerto 8080. Por lo tanto, los distintos EndPoints expuestos desde el WS son accedidos con una URL del tipo:

[http://192.168.0.170:8080/EncuestasFCM/\[controlador\]/\[servicio\]?\[parámetros\]](http://192.168.0.170:8080/EncuestasFCM/[controlador]/[servicio]?[parámetros])

Tal como se había diseñado anteriormente, el encargado de acceder a éstas URLs para el consumo de los servicios, es la clase **HttpAsyncTask** que extiende de **AsyncTask** perteneciente a la librería de Apache. Ésta clase permite realizar consultas HTTP a la URL especificada mediante un *HTTPClient*, obteniendo un *HTTPResponse* sobre los resultados encontrados en formato JSON, para luego ser plasmados en la

lógica de negocio de las distintas funcionalidades.

## DESARROLLO DE FUNCIONALIDADES Y PRESENTACIÓN DE LA APLICACIÓN MÓVIL.

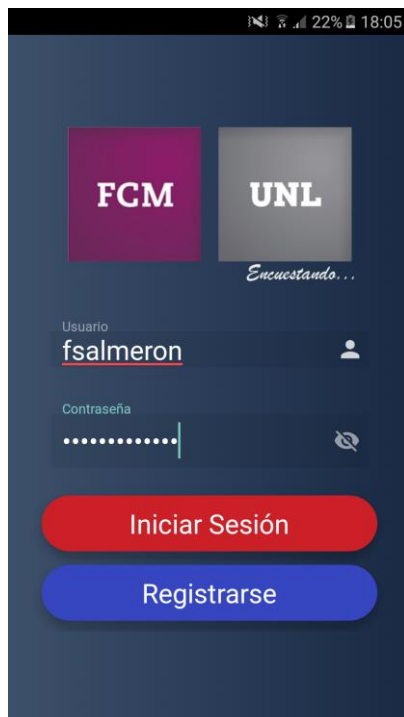
A continuación, se presentará la aplicación móvil denominada **Encuestando FCM** junto a todas las funcionalidades y librerías utilizadas para la realización de las mismas.



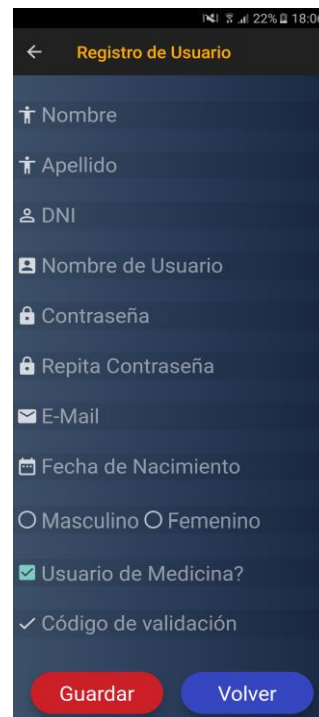
(Fig. 1 – Ícono de la aplicación)

### • PANTALLA DE LOGIN Y REGISTRO DE USUARIO:

Al instalar la aplicación y abrirla, se presentará hacia todos los usuarios una pantalla simple de logueo tal como se muestra en la Figura 2. Si ya se posee un usuario creado, simplemente se debe ingresar el *nombre*, *contraseña* e *Iniciar Sesión*. Se verificará consumiendo el servicio correspondiente en el WS si los datos ingresados son correctos y de esta forma redirigir a la pantalla principal adecuada (teniendo en cuenta el tipo de usuario del mismo), como ya se verá más adelante, caso contrario, se informarán los errores de logueo en la misma pantalla.



(Fig. 2 – Pantalla de Login)



(Fig. 3 – Pantalla de Registro de Usuarios).

Si no se posee un usuario para ingresar, se debe presionar sobre el botón *Registrarse*, el cual va a redirigir hacia la pantalla de *registro de usuario* como se presenta en la Figura 2. Aquí se deben colocar todos los datos personales y para el caso de aquellos “*Usuarios de Medicina*”, que serán los denominados anteriormente como *Usuarios Específicos*, se debe tildar el checkbox correspondiente y luego ingresar el código de validación, el cual debe ser solicitado a un superior de la facultad de medicina para obtener las funcionalidades de dicho tipo de usuario.

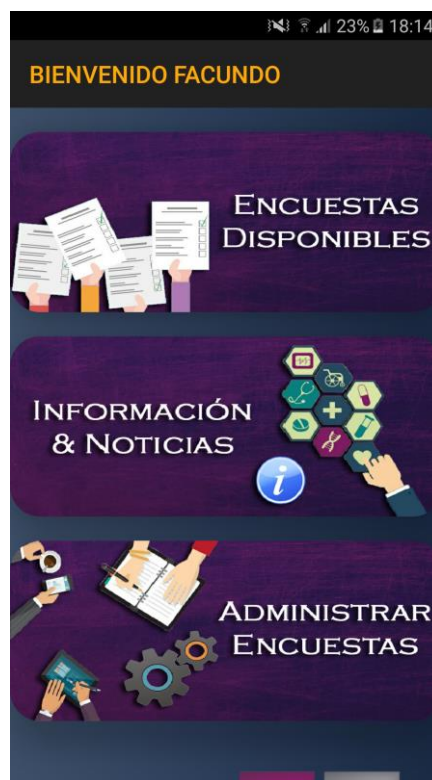
Al presionar guardar, inicialmente se verificará que todos los datos se encuentren cargados, ya que los mismos son de carácter obligatorio, caso contrario se informará. Una vez completados todos los datos se envían al WS para su registro, éste devolverá un *response* en formato JSON informando el éxito de la transacción o aquellos campos que se deben corregir. Por ejemplo: nombre de usuario o email ya en uso, password que no cumple la longitud, código de validación incorrecto, etc.

- **MENÚ PRINCIPAL:**

Como se mencionó anteriormente, al iniciar sesión se presenta un menú principal. Éste varía de acuerdo al tipo de usuario logueado. En el caso de los usuarios “*comunes*”, como se muestra en la Figura 4, se tendrán las opciones de *Encuestas Disponibles* e *Información & Noticias*. Como se verá más adelante, en ellas se podrán responder encuestas en una sola ocasión (siempre y cuando cumpla con las validaciones previas) y leer las noticias agregadas respectivamente. Cuando los usuarios son del tipo “*específicos*”, se presentará una pantalla como en la Figura 5. Aquí se tienen las mismas opciones que el otro tipo de usuario, con la diferencia de que se pueden responder ilimitadamente las encuestas, ver estadísticas sobre las mismas, exportar resultados, realizar AMBs de Informaciones, etc. Además, se haya la opción de *Administrar Encuestas*, para realizar los ABMs, habilitación e inhabilitación de ellas.



(Fig. 4 – Menú principal Usuario común)



(Fig. 5 – Menú principal Usuario Especial)



- **SECCIÓN NOTICIAS & INFORMACIÓN (I&N):**

Esta sección fue pensada para que los usuarios de la aplicación además de responder las encuestas, puedan informarse e instruirse acerca de distintas enfermedades, epidemiologías, informaciones relevantes, etc.

Para aquellos usuarios comunes que ingresan a la sección, se presentará en un *ListView* con un título y una breve descripción, las distintas I&N cargadas en base de datos (Figura 6). Para ello, se consumirá el correspondiente endpoint del WS al cargar las mismas. Al presionar sobre una de ellas, se abrirá un *WebView* en la aplicación misma, con el contenido de la URL correspondiente (Figura 9).



(Fig.6 – I&N U. común)



(Fig. 7 – I&N U. específico)



(Fig. 8 – Alta de I&N)



(Fig. 9 – Ejemplo I&N)

Para aquellos usuarios de tipo específico, como se puede ver en la Figura 7, se presentan las distintas opciones de ABM. Se encuentra disponible un botón para agregar I&N, el cual direcciona hacia la pantalla de alta (Figura 8). Aquí se debe colocar el título de la I&N, una breve descripción informativa y la URL a la que apuntará la misma. Al presionar sobre un ítem de la *ListView*, se abrirá un pop up que posee las opciones de *eliminar*, *modificar* (Figura 8) y *ver* (Figura 9). Todas estas opciones, consumen sus correspondientes servicios del WS para llevarlo a cabo.

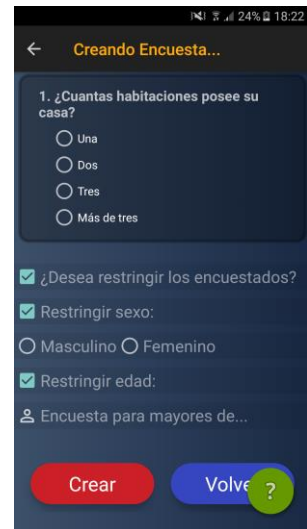
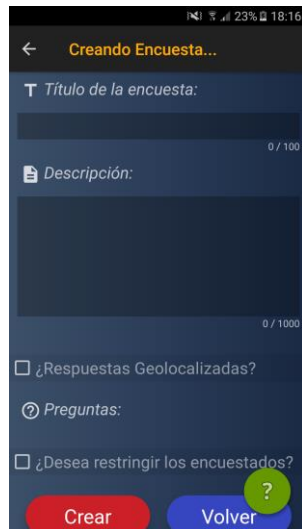
- **SECCIÓN ADMINISTRAR ENCUESTAS:**

Como se mencionó anteriormente, esta sección solamente estará disponible para aquellos usuarios de tipo específico. Al ingresar, se presentará un *ListView* de las encuestas existentes (con *activo = true*) traídas por WS. Cada ítem estará formado por un título y una descripción (Figura 10). Como se puede ver en la actual pantalla y en subsiguientes, se presenta un icono de ayuda (arriba a la derecha), el cual despliega un pop up informativo acerca de todas las funcionalidades presentadas, con la finalidad de colaborar en la experiencia de usuario.

Presionando sobre el botón rojo (+), se direcciona hacia la pantalla de *Nueva Encuesta* como se puede ver en la Figura 11. Aquí se deberá completar de carácter obligatorio el título y poseer al menos una pregunta. Luego se presentan una serie de validaciones



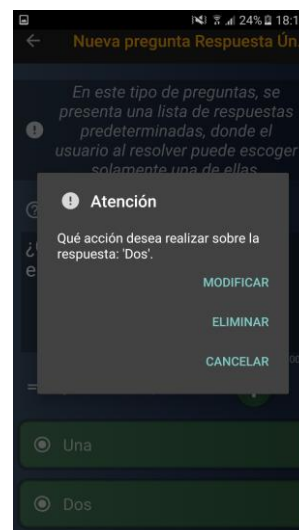
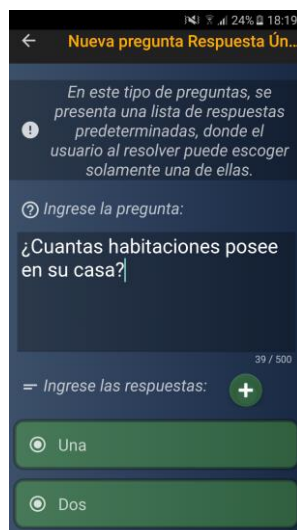
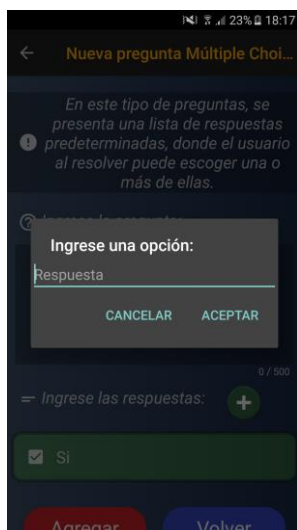
opcionales en forma de *CheckBox*. Por ejemplo, si se requiere que la encuesta sea *Geolocalizada* se debe tildar el mismo. Esto significa que cuando se resuelva la misma se almacenará la latitud y longitud de los encuestados, a modo de poseer referencias zonales para posteriores análisis. A su vez, se pueden agregar restricciones sobre los encuestados, es decir que los mismos sean de un sexo determinado y/o mayores a determinada edad (Figura 13).



(Fig. 10 – Lista Encuestas) (Fig. 11 – Nueva Encuesta) (Fig. 12 – Tipos Respuestas) (Fig. 13 – Validaciones)

Para agregar preguntas a la encuesta, se debe presionar sobre el botón verde (?). El mismo despliega una serie de opciones en relación al tipo de respuesta que tendrá y presionando sobre uno de ellos, se direccionará a la correspondiente alta.

Para aquellas preguntas que tendrán una serie de respuestas predefinidas, como por ejemplo las *Múltiple Choice* y *Respuesta Única*, se presenta un campo editable para ingresar la pregunta y un botón que despliega un pop up para ir ingresando las respuestas (Figura 14 y 15). Luego, al presionar sobre cada respuesta se presentará un *AlertDialog* (Figura 16) que permite modificar o eliminar la misma.



(Fig. 14 – Múltiple Choice) (Fig. 15 – Respuesta Única) (Fig. 16 – Opciones sobre respuesta)

Para aquellas preguntas que poseen una respuesta de tipo *Numérica* o *Textual*, el formato de alta es similar, donde el mismo presenta un campo editable para ingresar la

pregunta en cuestión (Figura 17 y 18). La diferencia entre ellas, es visible a la hora de responder la encuesta, ya que el teclado solamente habilitará los caracteres numéricos o alfanuméricos respectivamente.

En el caso de las preguntas con tipo de respuesta *Escala*, además de ingresar la pregunta en cuestión en el campo editable, posee otro campo que despliega un *NumberPicker*, eligiendo de ésta forma la escala máxima a responder, partiendo desde el 1 (Figura 19 y 20).

(Fig. 17 – Numéricas)

(Fig. 18 – Textual)

(Fig. 19 – Escala)

(Fig. 20 – Valores Escala)

Al ir agregando las preguntas como se mencionó anteriormente, aparecerán en la pantalla de nueva encuesta con su formato correspondiente, dependiendo del tipo de respuesta que contengan (Figura 11 y 13). Finalmente al presionar *Agregar*, se validará que todos los datos ingresados sean correctos y se procederá a persistir en base de datos consumiendo el WS, caso contrario se informarán los errores para corregirlos.

Nuevamente sobre la pantalla de *Administración de Encuestas*, como se presentó en la Figura 10, al presionar sobre un ítem específico se desplegará en un pop up las diferentes opciones a realizar, que se describirán a continuación:

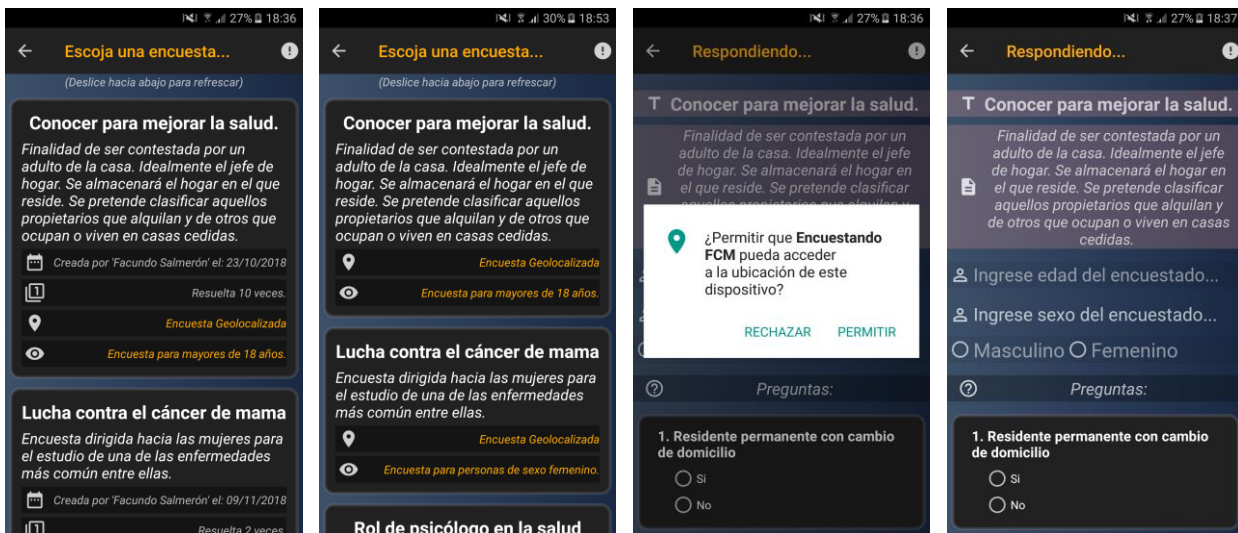
- **Eliminar:** procede a realizar una “baja lógica” de la encuesta. Es decir, la misma no será eliminada de la base de datos, sino que modificará los campos correspondientes para no ser cargada en la aplicación, es decir se colocará como *false* el campo ‘activo’ y se completarán los campos ‘usuario de baja’ y ‘fecha de baja’. Ésta metodología fue desarrollada a fin de tener una auditoría de la base de datos y mayor control sobre la misma.
- **Modificar:** ésta opción solamente estará visible para aquellas encuestas que aún no han sido resueltas. Se determinó esta metodología, para mantener integridad en los datos y no obtener resultados incorrectos o desactualizados. Al presionar sobre ésta opción, se abrirá una pantalla similar a la Figura 11, con todos los datos cargados correctamente, pudiendo eliminar, modificar o agregar preguntas y respuestas, cambiar validaciones, etc.
- **Habilitar:** al crear una encuesta, la misma queda en estado activa pero pendiente de habilitación. Es decir, para que ella quede disponible para responder, previamente debe ser habilitada. Ésta metodología está relacionada con la opción de modificar, ya que evitando que una encuesta quede automáticamente disponible de resolución, la misma puede modificarse cuantas veces quiera y cuando se considere que la misma está acorde para responderse, así habilitarla.

y empezar a recabar resultados.

- **Inhabilitar:** cuando se considere que una encuesta quedó desactualizada o no se necesiten más resultados de la misma, simplemente se inhabilita, haciendo que su resolución ya no esté más disponible en la correspondiente sección.

- **SECCIÓN ENCUESTAS DISPONIBLES:**

Al ingresar a la opción de *encuestas disponibles* desde el menú principal, se direcciona hacia una pantalla que presenta una lista de las mismas con una vista de tipo *Recycler View* por ítem. Como se mencionó anteriormente, aquí se cargarán todas aquellas encuestas que se encuentren en estado activas y habilitadas. Para ambos tipos de usuarios cada *Recycler* está compuesto por el título de la encuesta, descripción, una etiqueta que solamente será visible en encuestas geolocalizadas y las restricciones de resolución en caso de tenerlas. Únicamente para los usuarios de tipo especial se presentará un *TextView* que indicará el usuario creador y fecha, y otro indicando el número actual de resoluciones (Figura 21 y 22).



(Fig. 21 – Encuestas especial) (Fig. 22 – Encuestas común) (Fig. 23 – Permisos ubicación) (Fig. 24 – Resolución)

Al presionar sobre un ítem para responder se abrirá una pantalla con todos los datos de la encuesta. Si la misma tiene la característica de ser geolocalizada, previamente se le solicitarán los permisos al usuario para utilizar la ubicación del dispositivo (Figura 23). Éstos permisos serán requeridos solamente una vez en caso de aceptarse.

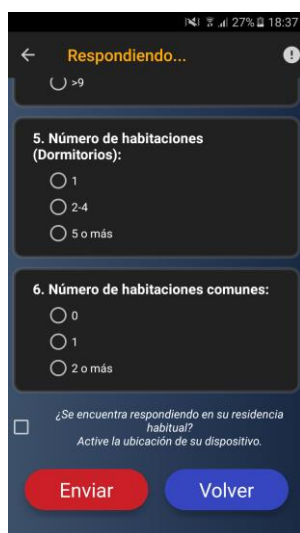
Cuando es resuelta por un usuario especial, al comienzo se presenta un campo para indicar la edad del encuestado y otro para indicar el sexo del mismo (Figura 24). Esto es para las denominadas “*salidas a terreno*”, cuando se encuesta a terceros. Para el caso de resolución en usuarios comunes, éstos campos no estarán presentes, ya que se obtendrán automáticamente desde el inicio de sesión.

Luego por cada pregunta a responder, se presentará una vista diferente dependiendo del tipo de respuesta de la misma, es decir:

- **Múltiple Choice:** respuestas como *Checkbox* pudiendo escoger una o más.
- **Respuestas Únicas:** respuestas como *RadioButton* pudiendo escoger solamente una.

- *Númericas*: teclado numérico disponible, plasmando la respuesta en un campo editable.
- *Textuales*: teclado alfanumérico libre de desarrollo, plasmando la respuesta en un campo editable.
- *Escala*: la respuesta debe ser elegida sobre un *NumberPicker* que presenta las escalas disponibles.

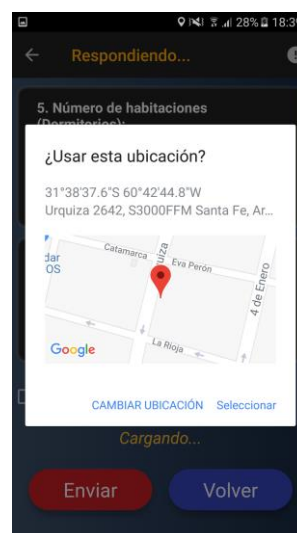
Al finalizar la resolución si la misma no fuera de carácter geolocalizada, simplemente se presiona enviar y se evaluará que todos los ítems hayan sido respondidos y no queden campos sin completar. Si hubiere algún tipo de error, se presentará en pantalla, caso contrario se almacenarán las respuestas en base de datos, consumiendo el correspondiente servicio del WS. Para el caso de aquellas encuestas de tipo geolocalizadas, se encontrará visible al final de la encuesta un checkbox para definir el lugar de residencia (Figura 25). Con el fin de obtener correctos datos de geolocalización sobre los usuarios encuestados, se les pregunta si al responder la encuesta se encuentran en su residencia habitual. Al tildar el checkbox y presionar enviar, se validarán que todos los datos sean correctos y se obtendrá la ubicación del encuestado (latitud y longitud) gracias a la funcionalidad de *LocationServices* de Google. En el caso de no tener activada la ubicación del dispositivo móvil o no haber aceptado los permisos para utilizarla (Figura 23), se informará al usuario como error y no se podrá responder la encuesta hasta realizar las habilitaciones correspondientes.



(Fig. 25 – Envío de Resolución)



(Fig. 26 – Elección de residencia)



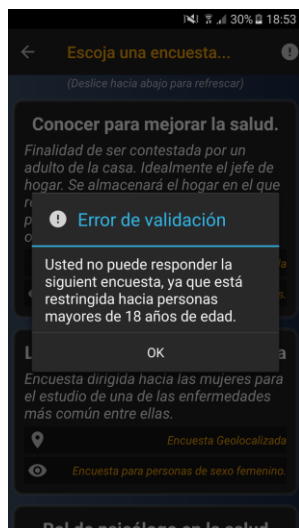
(Fig. 27 – Place Picker)

Si el usuario no se encuentra en su residencia habitual (checkbox sin tildar), se abrirá un *PlacePicker* (Figura 26), herramienta obtenida mediante la integración con el API de Google Maps, en donde se puede ingresar la dirección de residencia en la parte superior y luego buscar, o ir moviendo el *marker* para obtener la ubicación correspondiente. Finalmente, al presionar “*Seleccionar ésta ubicación*”, se mostrará un mensaje de comprobación (Figura 27) en donde se visualizará con mayor precisión la dirección escogida pudiendo volver a elegirla o confirmarla, invocando luego los servicios del WS para almacenar los resultados en base de datos.

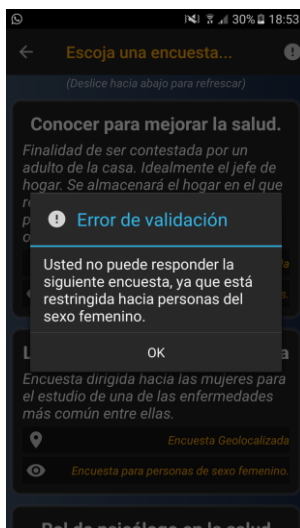
Para aquellos usuarios de tipo especial, todas aquellas restricciones escogidas inicialmente en la creación de las encuestas no serán tenidas en cuenta, así como también poder resolver las mismas de manera ilimitada. En cambio, para los usuarios comunes al momento de querer responder una encuesta, se tendrá en cuenta que cumpla con los requisitos y su resolución solamente estará disponible en una ocasión.



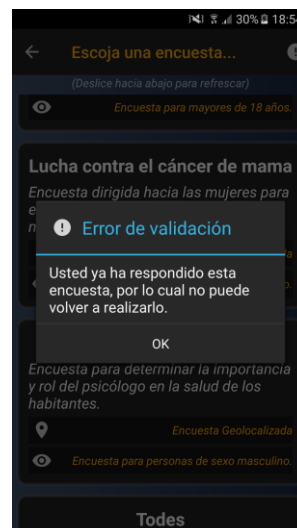
En el caso de presentarse algún inconveniente se informará y no se dejará resolver, como se puede ver en las Figuras 28, 29 y 30.



(Fig. 28 – Error de edad)



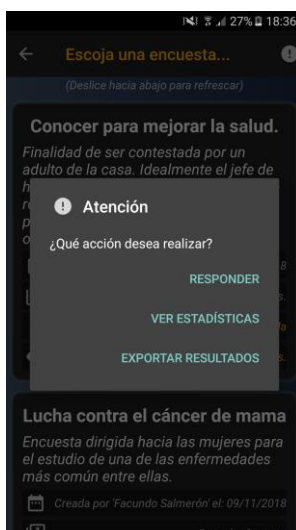
(Fig. 29 – Error de sexo)



(Fig. 30 – Encuesta ya resuelta)

## • EVALUACIÓN DE RESULTADOS:

Con la finalidad de obtener resultados y estadísticas de manera inmediata, dentro del menú de *encuestas disponibles*, para los usuarios especiales se colocaron dos opciones extras (además de responder) al presionar sobre un ítem como se puede ver en la Figura 31. Las mismas son: ver estadísticas de los resultados y exportar los mismos a un formato Excel.



(Fig. 31 – Opciones sobre una encuesta)

Al presionar sobre la opción “Ver estadísticas”, se cargará la encuesta donde se presentará el título y descripción de la misma, seguidos de un promedio de edad y sexo de los encuestados hasta el momento (Figura 32).

Una vez más, dependiendo del tipo de respuesta por cada pregunta, se renderizará la vista de manera distinta (Figura 33 y 34):

- *Múltiple Choice*: el promedio de resultados se presentará en gráficos de torta.
- *Respuesta Única*: el porcentaje de respuestas será presentado en gráficos de barras.
- *Núméricas*: se presentará el número promedio escogido por los encuestados en un *TextView* circular.
- *Textuales*: no presentarán estadística alguna, ya que al ser de libre desarrollo su análisis va más allá de resultados promediables.
- *Escalas*: se presentará el número promedio escogido por los encuestados en relación a la máxima escala posible en un *TextView* circular.



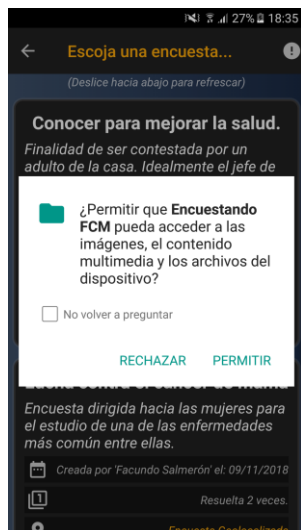
(Fig. 32, 33 y 34 Estadísticas Encuesta)

(Fig. 35 – Resultados Geolocalizados)

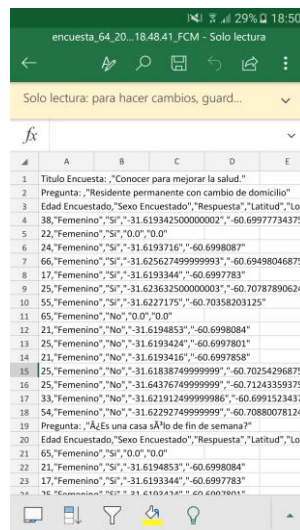
A su vez, presionando sobre una determinada respuesta, se mostrarán los resultados geolocalizados en caso de ser una encuesta de dicho tipo, mediante la integración con el API de Google Maps. Se distribuirán las respuestas en Google Maps gracias a la latitud y longitud almacenada en base de datos, clasificando por colores aquellas respuestas iguales en el caso de *Múltiple Choice* y *Respuestas Únicas*. Además, presionando sobre los marcadores se presentará una breve descripción que indicará la respuesta, sexo y edad del encuestado.

Por otro lado, si se escoge la opción de “*Exportar Resultados*”, se le solicitará previamente al usuario aceptar los permisos de almacenamiento (Figura 36). En caso de rechazarlos, se denegará la exportación informando al usuario de la misma manera que con los permisos de ubicación. Al aceptarlos, se procederá a convertir los datos a un formato *Excel* gracias a la librería de *OpenCSV*. Los mismos estarán disponibles para su posterior estudio por profesionales adecuados de la medicina, así como también bio-estadísticos.

La conversión de datos se realizará de manera asíncrona, es decir, en segundo plano para no entorpecer el funcionamiento normal de la aplicación. Es por ello, que puede tardar algunos segundos en producirse, para luego estar disponible dentro de las descargas del dispositivo móvil. En el archivo Excel se encuentran los datos principales de la encuesta como el título y descripción, y por cada pregunta se listan las respuestas obtenidas hasta el momento, con el sexo y edad del encuestado, latitud y longitud, etc.



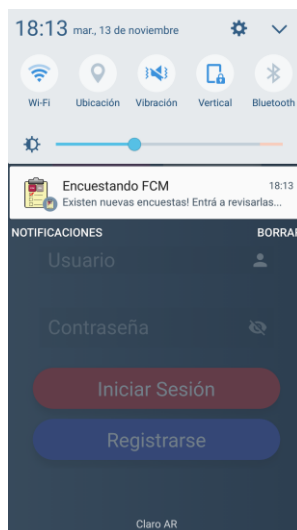
(Fig. 36 – Solicitud permisos almacenamiento)



(Fig. 37 – Resultados exportados)

## • ENVÍO DE NOTIFICACIONES:

Finalmente, se integró a la aplicación el API de Firebase el cual permite el envío de notificaciones. Este API provee de manera muy sencilla un servicio de mensajería, donde a la hora de enviar notificaciones el administrador de la aplicación debe ingresar a la web de desarrollo <https://console.firebase.google.com> y al enviar les debe colocar un título y una descripción (entre otras configuraciones opcionales), enviando de forma inmediata la notificación hacia todos los dispositivos que tengan instalado *Encuestando FCM* (Figura 38). Además, se puede enviar la notificación hacia determinados dispositivos, programar horarios y enviarlas de manera automática, etc.



(Fig. 38 – Notificaciones)

## • TESTING:

En esta etapa se realizaron distintas pruebas de verificación y validación tanto para el Web Service como para la aplicación Android, poniendo mayor énfasis en el primero, teniendo en cuenta que es el componente primordial para realizar las principales funcionalidades dentro de la aplicación.



Este proceso de análisis y pruebas tienen las siguientes finalidades:

- Comprobar que el sistema funciona tal y como se ha especificado en la etapa de requerimientos, satisfaciendo los mismos.
- Detección de comportamientos inesperados, defectuosos o que no cumplen la especificación inicial.
- Corresponder a las buenas prácticas de ingeniería de software, las cuales permiten desarrollar un producto que asegure calidad tanto en su despliegue como en la integración.

Logrando de ésta forma responder ¿Estamos construyendo el producto correcto?  
¿Estamos construyendo el producto correctamente?

Para ello se realizaron pruebas de software inicialmente sobre el Web Service, mediante la utilización de distintas tecnologías que permitan testear cada uno de las funcionalidades implementadas y posteriormente la correcta integración con la aplicación Android.

Se implementaron testing automatizados principalmente sobre los distintos controladores del WS, ya que por la arquitectura definida inicialmente, cada uno de ellos invocará internamente a un servicio correspondiente, y éste a su vez a un dao (capa de datos). Mediante la utilización de los frameworks **JUnit** y **Mockito** se realizaron pruebas unitarias de los controladores, lo cual tiene como característica evaluar el correcto funcionamiento de un módulo en forma aislada, es decir se va probando de a pequeñas porciones de código donde la ejecución es más rápida. Para ello se establece un escenario posible con datos de entrada ejecutando las acciones de cada módulo en forma separada y corroborando el comportamiento esperado en las salidas de cada una de ellas.

La forma de trabajar que posee JUnit sobre cada prueba unitaria es mediante la anotación **@Test**. Dentro de cada uno de ellos se invoca a una funcionalidad de un determinado controlador, pasándole datos estáticos de entrada en la request y simulando objetos gracias a Mockito. El valor de retorno esperado se evalúa mediante afirmaciones "**asserts...**" (Ej. *assertNull*, *assertFalse*, *assertThrows*), informando si la clase cumple con la especificación y retornando JUnit si el método pasó el caso de prueba o falló.

Luego sobre el WS se utilizó el servicio de **Jacoco** (*Java Code Coverage*), el cual permite cuantificar el porcentaje de código fuente ejecutado en los distintos test en conjunto. De esta forma se determina la cantidad de código que es cubierto en las pruebas, lo cual es útil para la depuración de errores o detectar malas definiciones en los módulos del software. Por lo tanto, en las distintas pruebas JUnit se tuvieron en cuenta las distintas entradas a colocar para que la cobertura cumpla con la mayor cantidad posible.

Como se puede ver en la figura 39 se logró llegar a un 85% de cobertura en el total del software, donde las principales funcionalidades dentro de los controladores, servicios y daos se encuentran totalmente cubiertos.

## EncuestasFCM

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods	Missed	Classes
com.salmeron.encuestasfcm.serializer		70%		33%	12	21	24	92	6	15	0	3
com.salmeron.encuestasfcm.dap.impl		92%		84%	7	47	12	195	0	23	0	2
com.salmeron.encuestasfcm.utils		60%		0%	3	6	12	39	1	4	0	2
com.salmeron.encuestasfcm.config		63%		16%	3	6	6	25	0	3	0	1
com.salmeron.encuestasfcm.filter		81%		n/a	0	8	2	29	0	8	0	1
com.salmeron.encuestasfcm.base		83%		n/a	4	8	3	15	4	8	0	2
com.salmeron.encuestasfcm.enum		81%		n/a	2	6	0	5	2	6	0	1
com.salmeron.encuestasfcm.model		85%		50%	2	12	2	21	1	11	0	2
com.salmeron.encuestasfcm.controller		100%		88%	3	26	0	80	0	13	0	3
com.salmeron.encuestasfcm.service.impl		100%		n/a	0	10	0	31	0	10	0	2
com.salmeron.encuestasfcm.dap		100%		100%	0	17	0	26	0	15	0	2
com.salmeron.encuestasfcm.service		100%		n/a	0	2	0	3	0	2	0	1
Total	407 of 2.767	85%	25 of 96	73%	36	169	61	561	14	118	0	22

Created with JaCoCo 0.7.9.201702052155

(Fig. 39 – Code coverage Web Service).

Como ya se ha mencionado anteriormente, el SDK de Android está basado en Java, por lo tanto, también fue posible realizar pruebas unitarias mediante la utilización de JUnit. En éste caso el foco principal de testing se basó en la integración de los distintos componentes tanto para la comunicación interna de los eventos en la aplicación como para la externa (comunicación con el WS). La metodología de trabajo es exactamente igual que la realizada anteriormente, colocando distintas entradas en las ejecuciones y evaluando los distintos resultados.

Finalmente se llevaron a cabo distintas pruebas manuales a “*nivel usuario*” para las cuales se simuló un servidor en un computador personal utilizando una red local de trabajo, como se explicó anteriormente y se utilizaron varios smartphones Android con diferentes versiones y resoluciones.

Se procedió a realizar testing sobre todos los componentes ya sea de manera individual e integrada, verificando el funcionamiento conjunto de los mismos. Se tuvieron en cuenta los distintos escenarios posibles a la hora de las pruebas, como puede ser la falta de conexión móvil, geolocalización, almacenamiento, vinculación y funcionamiento del WS, informando al usuario ante cada falla producida. También se verificó su uso con la utilización de WiFi y datos móviles, comparando y verificando el tiempo de respuesta en conexiones.

Además, se realizaron pruebas sobre distintos dispositivos móviles Android, en diferentes versiones y resoluciones para verificar la adaptabilidad a cada uno de ellos, se produjo la utilización de la aplicación sobre distintos dispositivos de forma simultanea para verificar el correcto funcionamiento y conexión con el WS.