



**DEPARTAMENTO
DE COMPUTACION**

Facultad de Ciencias Exactas y Naturales - UBA

Big Tiza

Ingeniería de Software II

Integrante	LU	Correo electrónico
Agustina Ciraco	630/06	agusciraco@gmail.com
Alejandro Rebecchi	15/10	alejandrorebecchi@gmail.com
Maria Lara Gauder	27/10	marialaraa@gmail.com
Martin Heredia	146/11	martin.herediaf@gmail.com



Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

Ciudad Universitaria - (Pabellón I/Planta Baja)

Intendente Güiraldes 2160 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (54 11) 4576-3359

<http://www.fcen.uba.ar>

Índice

1. Introducción	3
2. Atributos de Calidad	4
2.1. Listado atributos de calidad pedidos	4
2.2. Escenarios de Atributos de calidad	7
3. Arquitecturas	8
3.1. Arquitectura TP1	8
3.2. Arquitectura TP2	9
3.2.1. Vista Componentes y conectores	9
3.2.2. Vista Alocaación	13
3.3. Programming in the Large vs Programming in the Small	14
4. Discusión	16
5. Conclusiones	17

1. Introducción

En este trabajo práctico se presentan las arquitecturas del sistema correspondiente a la primera entrega (versión para escuela de villa urquiza) y la correspondiente al sistema pedido por el ministro, cuyo alcance es a nivel país. Se detallan los atributos de calidad pedidos, junto con sus respectivos escenarios. Por otro lado se presenta una discusión sobre las diferencias entre las metodologías utilizadas en cada tp, y por último conclusiones.

2. Atributos de Calidad

2.1. Listado atributos de calidad pedidos

Performance - Deadline	<p>No se acepta ningún tipo de demoras ante el monitoreo del estado de las campañas.</p> <p>Fuente: Externa Estímulo: El usuario desea ver el estado actual de la campaña. Requiere visualización de las estadísticas generadas hasta el momento, de los mensajes enviados y los no enviados. Entorno: Normal Artefacto: Sistema Respuesta: Se procesa el pedido de visualizar el estado de la campaña. Medición de la respuesta: El pedido debe resolverse en menos de un segundo.</p>
Disponibilidad	<p>Se pueden generar fallas de comunicación durante la transición de servidores.</p> <p>Fuente: Interna Estímulo: Falla por omisión Entorno: Normal, el sistema funciona en su modo habitual Artefacto: Canal de comunicación Respuesta: Se deberá pasar a un entorno degradado, donde se solucionen los pedidos realizados durante la transición, sin importar la performance del sistema. Medición de la respuesta: Se permanecerá en modo degradado hasta que se termine la transición.</p>
Modificabilidad	<p>Cambio de servidores entre etapa de Córdoba a etapa país.</p> <p>Fuente: Administrador Estímulo: Se quiere aumentar la cantidad de servidores para poder almacenar datos de todos los usuarios del país' Entorno: En ejecución Artefacto: Sistema Respuesta: Los nuevos servidores son agregados al sistema y estan listos para ser usados Medición de la respuesta:</p>
Seguridad	<p>Proteger datos.</p> <p>Fuente: Gestor Mensajes Estímulo: Intenta enviar un mensaje Entorno: Online Artefacto: Sistema de envíos de Mensajes Respuesta: Se logra enviar el mensaje encriptado por alguno de los canales de comunicación disponibles Medición de la respuesta: La probabilidad de que una persona logre decodificar el algoritmo es de 0.0001 %</p>
Seguridad	<p>Auditar</p> <p>Fuente: Estímulo: Entorno: Artefacto: Respuesta: Medición de la respuesta:</p>
Disponibilidad	<p>Envío de mensajes. (Patagonia)</p> <p>Fuente: Gestor de Mensajes Estímulo: Se intenta enviar un mensaje a un usuario en la Patagonia</p>

Entorno: Normal

Artefacto: Sistema de envío de Mensajes

Respuesta: El mensaje es enviado al usuario en cuestión por medio del uso de los drones

Medición de la respuesta: El mensaje a la Patagonia llega el 98 % de las veces

Certeza de Datos Para la estimacion de campañas

Fuente: Usuario

Estímulo: Pide ver el estado actual de todas las campañas

Entorno: Normal

Artefacto: Cliente resultados

Respuesta: Se muestra por una interfaz un listado (de manera incremental) de todas las campañas con datos aproximados de sus estados actuales

Medición de la respuesta: El 80 % del estado de las campañas mostradas es correcto

Modificabilidad-

Scalabilidad Volumen de datos

Fuente:

Estímulo:

Entorno:

Artefacto:

Respuesta:

Medición de la respuesta:

Performance-

Escalabilidad Mantener la performance en el envío de mensajes Se quiere expandir el sistema de la provincia de Cordoba a Nivel nacional manteniendo la performance en el envío de mensajes.

Fuente: Sistema

Estímulo: Envío de mensaje

Entorno: Funcionamiento con 40 millones de posibles destinatarios.

Artefacto: Sistema de envío

Respuesta: Se mantiene en valores similares el tiempo de envío

Medición de la respuesta: El tiempo de envío de mensajes aumenta en menos de un 20 %

Modificabilidad Campañas privadas

A futuro se quiere poder permitir a empresas privadas crear campañas de promoción de productos

Fuente: Representante empresa privada

Estímulo: Carga de campaña

Entorno: Funcionamiento normal

Artefacto: Aplicación web

Respuesta: Se adapta la aplicación para permitir la carga de campañas privadas de productos

Medición de la respuesta: En menos de 2 semanas se deben hacer los cambios y

poner en producción la nueva funcionalidad para la carga de campañas por empresas privadas

Flexibilidad política nacional de métricas

Permitir que diferentes proveedores de contenido adapten campañas nacionales a las condiciones locales de cada región

Fuente: Proveedor de contenido

Estímulo: Adaptación de campaña

Entorno: Funcionamiento normal

Artefacto: Aplicación web

Respuesta: Se adapta la campaña a las condiciones de la región

Medición de la respuesta: Se adapta la campaña en tiempo de ejecución (sin necesidad de bajar el servidor o detener la aplicación)

Usabilidad Crear campañas

Se desea que los usuarios puedan crear campañas correctamente de forma fácil y rápida

Fuente: Usuario

Estímulo: Pedido de creación de campaña

Entorno: Funcionamiento normal

Artefacto: Aplicación web

Respuesta: El usuario crea una campaña correctamente

Medición de la respuesta: En un test de usabilidad el 95 % de los usuarios logra crear una campaña correctamente en menos de 5 minutos

Usabilidad Visualizar campañas

Se desea poder ver toda la información referida a una campaña de forma clara.

Fuente: Usuario

Estímulo: Pedido de visualización de campaña

Entorno: Funcionamiento normal

Artefacto: Aplicación web

Respuesta: Se visualiza toda la información disponible relacionada con la campaña

Medición de la respuesta: En un test de usabilidad el 95 % de los usuarios logran extraer la información esperada en menos de 30 segundos

Usabilidad Visualizar Resultados

Se desea poder comparar resultados de varias campañas de forma comparable, fácil de entender.

Fuente: Usuario

Estímulo: Pedido de comparación de resultados de 2 campañas

Entorno: Funcionamiento normal

Artefacto: Aplicación web

Respuesta: Se muestran los resultados de ambas campañas en la misma pantalla

Medición de la respuesta: En un test de usabilidad el 95 % de los usuarios logran comprender los resultados en menos de 5 minutos

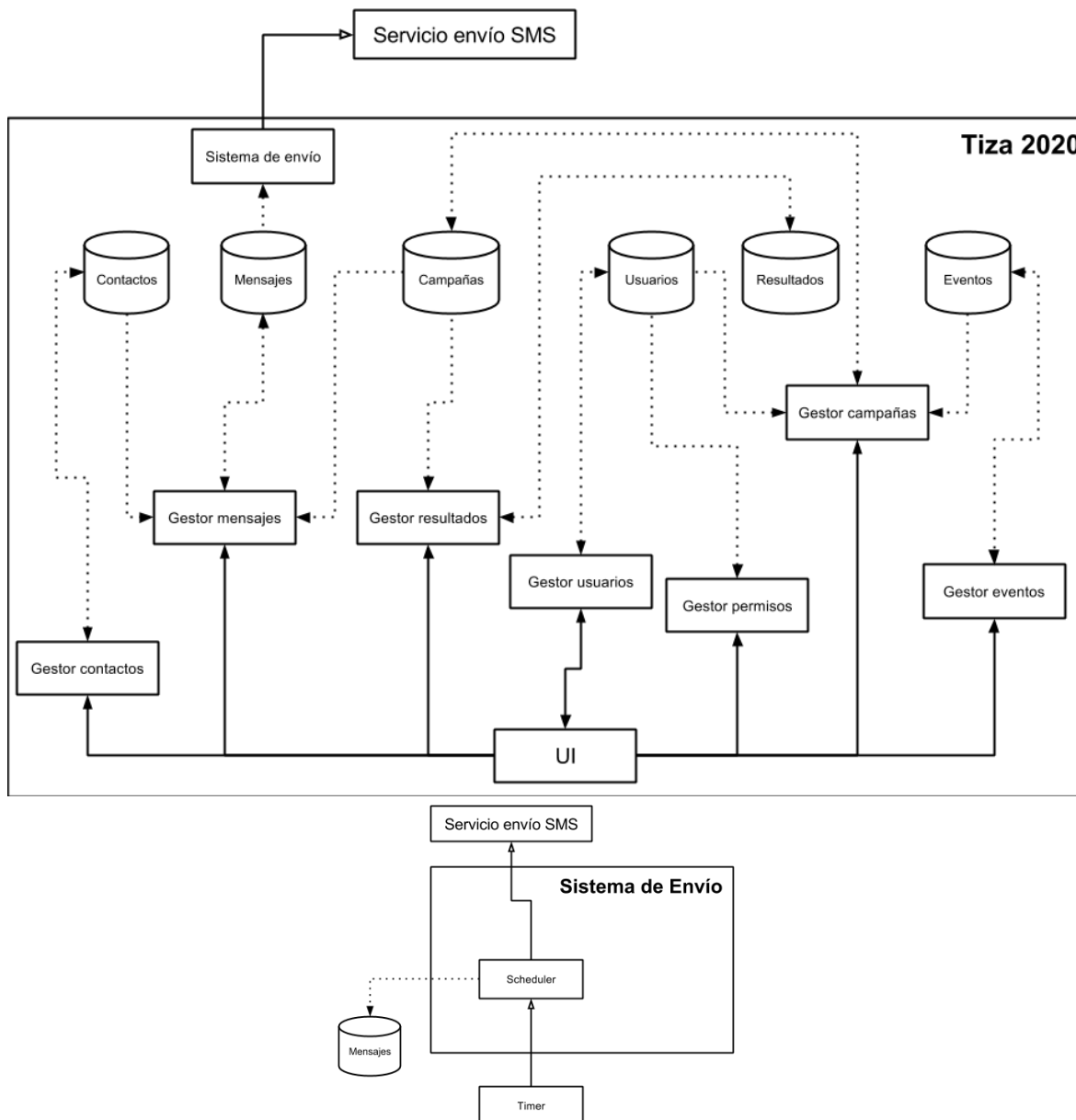
2.2. Escenarios de Atributos de calidad

Clasificación	Caso de Uso	Horas H.
Usuarios	Ingresando al sistema	36
	Cargando usuario	13

3. Arquitecturas

3.1. Arquitectura TP1

A partir del proyecto Aula al 2020, de los objetivos planteados y el prototipo desarrollado, se define el siguiente diagrama de arquitectura:



En primer lugar se presenta la UI, que es la encargada de la interacción del usuario con el sistema de Tiza 2020. La misma recibe los pedidos por parte del usuario. Para comenzar, se deberá validar el acceso del mismo, es decir, verificar su previa inscripción en el sistema. Para eso el UI indica al **Gestor usuarios** el id del usuario que intenta acceder. Este último lee la base de datos, buscando los datos del mismo. En caso de no existir, retorna al UI un aviso de usuario inválido, el cual se encargará de notificarle a la persona. Caso contrario, responde que se permite el acceso al usuario al sistema. En caso de que el usuario sea válido, el UI envía un pedido al **Gestor campañas**, pidiendo las campañas correspondientes al usuario para poder mostrarlas como el siguiente menú, luego del inicio de sesión. El componente **Gestor campañas**, pide del repositorio las campañas que tienen en el campo de *usuario* con *permiso de acceso* al id del usuario. Luego, envía la información de todas las campañas,

completándolo con los datos requeridos en los demás repositorios, como podría ser el caso de los teléfonos de los contactos. El proceso se repite para cada menú que el usuario desee ver. Además, el gestor se encargará de recolectar la información necesaria y traducirla a una interfaz para que sea comprensible por el usuario y se le enviará al UI, quien se encarga de mostrarla. En el caso de que se desee mostrar la agenda correspondiente a un usuario, para poder incluirlo en la creación de un mensaje, el UI pedirá al **Gestor contactos**, la información necesaria. Este último se encargará de acceder a la base de datos y recolectar los datos pedidos. El proceso es el mismo para el **Gestor mensajes**, **Gestor resultados**, **Gestor eventos**, siendo cada uno el encargado de interpretar para el UI los datos necesarios y de armar la interfaz necesaria.

Por otro lado, cuando un usuario, tanto Docente como personal de la Secretaría o Dirección, desean crear o modificar algún objeto del sistema, lo indicarán al UI, utilizando los accesos que correspondan, y este se encargará de avisar al gestor que corresponda con el objeto. El gestor correspondiente se encargará de agregar el nuevo objeto al repositorio necesario. En caso de que se desee modificar uno existente, también levantará de la base de datos el objeto, le aplicará la modificación indicada por el usuario y lo volverá a cargar en el repositorio.

El **Sistema de envío** es el encargado de enviar los mensajes. El *Timer* le envía cada hora un mensaje asincrónico al *Scheduler* con la hora y fecha actuales. El *Scheduler* se encargará de leer en la base de datos aquellos mensajes que cumplan con esa hora y fecha de envío. Al obtener todos los mensajes, los traducirá para poder ser interpretados por el **Servicio envío SMS**. Este último es un sistema externo que se contratará para cumplir la función de enviar los SMS necesarios.

3.2. Arquitectura TP2

3.2.1. Vista Componentes y conectores

La arquitectura de esta parte se realizó contemplando los atributos de calidad nombrados en la sección anterior. A continuación se muestra el diseño realizado para esta parte, seguido de un detalle de decisiones tomadas.

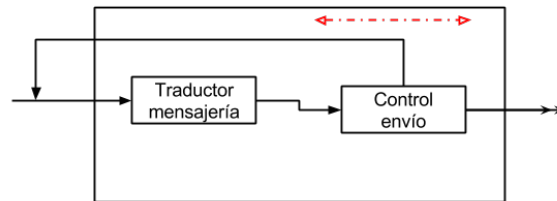
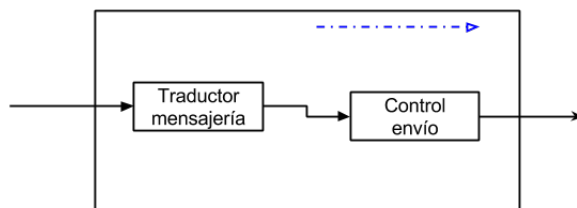


Diagrama de flujo de un sistema de envío (Sistema envío). El sistema incluye un Scheduler y un Selector de servicio. El Scheduler recibe entrada de un Timer y envía datos al Selector de servicio. El Selector de servicio envía datos al Scheduler.

```

graph LR
    Input(( )) --> Scheduler
    subgraph SistemaEnvio [Sistema envío]
        Scheduler[Scheduler]
        Selector[Selector de servicio]
        Scheduler --> Selector
        Selector --> Scheduler
    end
    Timer[Timer] --> Scheduler
    Selector --> Output(( ))

```



Viendo en detalle el conector rojo del camino sin fallas se puede observar que existe un traductor para poder comunicarse con cada tipo de servicio (Facebook, Twiter etc) ya que cada uno tiene un formato distinto. En el caso del conector azul de camino alternativo, que contempla la falta de conexion, al ver el conector más en detalle se puede observar que se tiene el componente servicio alternativo de mensajería, quién se encarga de tomar la decición correspondiente y conectarse con el servicio de la empresa de telefonía privada para poder concretar el envío.

Además en este caso se contempló un tercer caso que coincide con el análisis de riesgo, tomado para la parte de planificación, donde se considera una conexión directa con un servicio de correo (OCA), para poder contar con él en el caso de no tener ningún tipo de conexión con los otros servicios.

Al considerar las situaciones de envío de camino sin fallas, donde como hay que evitar el uso de la empresa de telefonía privada, ya que esta puede generarle costos a los receptores, lo cual se desea evitar, por lo que se van a hacer varios intentos, antes de tomar la Decisión de cambiar de conector.

Decisión de Diseño Al momento de contemplar en las respuestas de aquellas campañas que esperar respuestas a los mensajes enviados, se asume que los mensajes se envían con un código para vincular con dicha campaña, para de esta manera poder generar los resultados correspondientes.

Por otro lado, se optó por tener un componente **Servicio de evaluacion de estadisticas**, el cual es externo con el cual hay que conectarse al momento de evaluar los resultados, el mismo es definido por cada municipio acorde a lo que se desee evaluar (campaña de saludo, de educacion vial, etc)

Otra parte importante a la hora de hablar de la arquitectura de Big Tiza, es la parte del procesado de resultados de las campañas, ya que esta parte involucra 2 partes importantes, la de procesado de campañas y la parte de visualizarlos en por ejemplo un dashboard. En cuando al procesado de los resultados, primero hay que considerar cómo llegan al sistema, una opción posible es la carga manual de dichos resultado. Esta opción es considerada dado que hay campañas, como por ejemplo una campaña de vacunación, que requiere de requiere de un proceso de que la misma llegue a la población, se considere y tenga efecto, en este ejemplo el efecto es que la persona se vacune y cuando lo haga eso suma un resultado a la campaña, como que 100 personas recibieron la vacuna para la gripe en el hospital de villa urquiza el día 4 de mayo de 2015, que luego será agregado al sistema por personal reponsable. Respecto a la arquitectura considerada para este punto, quién recibe los resultados dichos anteriormente es el componente gestor de resultados. Por otro lado, se contemplan campañas interactivas, es decir que envían mensajes cuyo efecto es una respuesta el mensaje enviado,

por lo cual este tipo de resultados, se reciben directamente en el sistema, el encargado de recibirlo nuevamente es el componente gestor de resultados.

Al recibir un resultado de una campaña, éste debe ser procesado inmediatamente para poder ser visualizado lo antes posible, ya que se desea obtener el dashboard siempre actualizado (el dashboard es el componente que se encargará de contemplar el requerimiento de que las campañas se puedan monitorear de manera ágil). por ejemplo se puede considerar el resultado 4 personas recibieron la vacuna contra la gripe en el hospital Posadas, estos valores deben ser evaluados acorde a las métricas definidas para la campaña definida por el ministerio de salud para la prevención de la gripe. Dependiendo de la definición de la métrica este valor (4 personas) puede ser bueno o malo. ya que por ejemplo si se espera que haya 100 personas vacunadas en el posadas, este valor es bajo. Luego de evaluar dicho resultado, el mismo está listo para ser visualizado en el dashboard, también se guarda para una vez que se haya finalizado la campaña el mismo pueda ser reunido con todos los resultados de la misma y obtener así un resultado final certero de la campaña.

Otro aspecto que fue considerado al momento de definir la interfaz para que un usuario pueda definir campañas es el de seguridad, ya que se hizo hincapié en que van a haber distintos tipos de usuario, y no todos pueden ver información de todas las campañas, en particular a la hora de tener la posibilidad de agregar campañas para empresas privadas, como por ejemplo addidas, se quiere tener la posibilidad de restringir la información a la que accede el usuario representante a addidas. *Decisión de diseño* Para cubrir este requerimiento se decidió agregar un web browser.

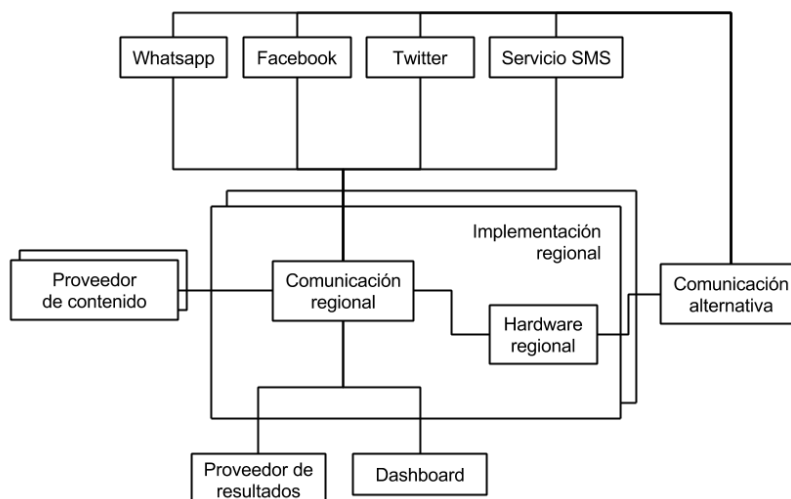
Otras consideraciones que se tienen respecto de la seguridad de la información, es que los otros canales de acceso al sistema, por ejemplo la carga de resultados, o como el componente de recepción de mensajes, tengan revisiones de seguridad previo a acceder a la base de datos para persistir información, para evitar un potencial ataque de seguridad.

s Volviendo al requerimiento de visualizar el resultado de campañas de manera ágil, como se presento anteriormente, se consideró un dashboard. *Decisión de diseño* Para ello se optó por realizar un componente **Servidor Dashboard** el cual se responsabiliza de recibir y mostrar los resultados, constantemente. Además este componente tiene dos formas de recibir resultados. Dado que se desea obtener en tiempo real resultados procesados una forma de que el componente reciba es el camino de los valores que se evalúan constantemente de la campaña, y otra es cuando se tienen resultados finalizados de las misma los puede seguir visualizando a pedido del componente **Cliente Dashboard** del lado del usuario.

Se tomó esta decisión teniendo en cuenta el volumen de información a procesar y los recursos con los que se cuentan, ya que durante la vigencia de una campaña se van a obtener volúmenes muy grandes de resultados parciales, que los mismos se quieren ver reflejados en el dashboard, pero también se desea tener la información correcta del resultado final conseguido a través de la campaña concretada. Al considerar resultados parciales se desea conseguir valores representativos al volumen total de incidencia de la campaña, por lo que esos resultados parciales se deben evaluar a nivel porcentaje de incidencia de dicho volumen para que al visualizar se pueda reflejar la métrica utilizada por la campaña. Teniendo en cuenta los

escenarios de performance se tomó la Decisión dicha anteriormente.

3.2.2. Vista Alocación



En esta vista se muestran dónde está contenido cada componente externo presentado en la vista anterior. En este diagrama se puede observar que los servicios de mensajería se separan y muestran cada uno en particular, ya que se considera que cada servicio corre por separado ya que son empresas diferentes.

Por otro lado se muestran los proveedores de contenido que es aquel componente responsable de agregar a las campañas aquellos logos, mensajes o lo que se desee mostrar, esto se decidió en consideración a que las campañas, tanto en esta primer etapa como en la siguiente, van a ser de distintas índoles ya sea para creación de una campaña por el ministerio de salud, como por el ministerio de seguridad vial, y en particular en un futuro se consideran las campañas de empresas privadas como NIKE.

Por último, se puede destacar el componente **Empresa de telecomunicaciones** a quién se le asigna en el conector de camino con falla de conexión, la responsabilidad de enviar los mensajes utilizando su infraestructura, por ello dicho sistema está por fuera de Big Tiza.

3.3. Programming in the Large vs Programming in the Small

En esta sección se compararán la Arquitectura correspondiente al Diseño Orientado a Objetos (small programming) y la nueva Arquitectura de gran escala (large programming) correspondientes a este trabajo.

Comunicaciones con entidades externas

Diseño O. O.	Arquitectura
Un solo componente externo, correspondiente al Sistema de Envío de Mensajes. Esto se debió a que era el único canal de comunicación. Con respecto a los resultados, estos eran cargados manualmente al sistema.	La cantidad de canales de comunicación aumentaron, no sólo abarcando al anterior Sistema de Envío de Mensajes, sino también a redes sociales como Facebook y Twitter, y la posibilidad de envío por Correo Argentino. Para los resultados, como la cantidad de mensajes y respuestas son masivas, hacerlo manualmente sería imposible. Para esto se cuenta con la opción de poder ser cargadas automáticamente por medio de Web Services externos (como por ejemplo, un Web Service ofrecido por el Ministerio de Salud, para obtener resultados de campañas de salud)

Seguridad de datos

Diseño O. O.	Arquitectura
La comunicación con el servicio de Mensajería SMS era insegura. Cualquier persona podría capturar el tráfico y obtener datos privados de los usuarios.	Ahora nuestro sistema al ser a escala Provincial (y posteriormente a escala Nacional), debe garantizar la seguridad de los datos de cada usuario. Esto se logra por medio de algoritmos de encriptación.

Disponibilidad

Diseño O. O.	Arquitectura
No siempre estaba garantizado que los mensajes enviados efectivamente llegaran. Si luego de 3 intentos no se lograba enviar algún mensaje, se lo descartaba.	Para intentar suplir la anterior falla, los mensajes persisten en el tiempo, y mientras tenga sentido el mensaje en el contexto de su campaña y aún no haya llegado, se prueba el envío por otros medios previamente mencionados o se reenintenta en caso de haber ya agotado todos los canales posibles.

Alocación

Diseño O. O.	Arquitectura
Al no ser muy grande el sistema y la cantidad de datos no era demasiada, con una máquina alcanzaba para alocar cada componente del mismo y almacenar las campañas, mensajes, usuarios y destinatarios.	Ahora aparece el concepto de BigData. Se cuenta con una masiva cantidad de datos y a la vez está la necesidad de implementar sistemas performante. Es por esto que ahora el sistema está alocado en varios servidores, aprovechando la capacidad de paralelizar trabajo y distribuir estratégicamente los datos entre todos estos lugares.

Tácticas

Diseño O. O.	Arquitectura
En esta arquitectura no se ven reflejadas ninguna de las tácticas de atributos de calidad, ya que no fueron tenidas en cuenta.	Se vio en la necesidad de aplicación de algunas tácticas para la resolución de problemas típicos de Disponibilidad, Performance, Seguridad entre otros.

Big Data

Diseño O. O.	Arquitectura
<p>Como se mencionó previamente, en esta etapa la cantidad de datos es mínima, con lo cual esto permitía que para poder obtener el estado de todas las campañas, con un simple listado de las mismas se lograba visualizarlas, y no se consumía mucho tiempo.</p>	<p>En cambio, y como también se mencionó, en esta etapa al haber una inmensa cantidad de datos, la visualización de las campañas pasa a ser un reto. Aparece la necesidad de aplicar heurísticas usando estadísticas para dar una idea del estado de las mismas, debido a la imposibilidad de brindar un servicio exacto tiempo real.</p>

4. Discusión

5. Conclusiones