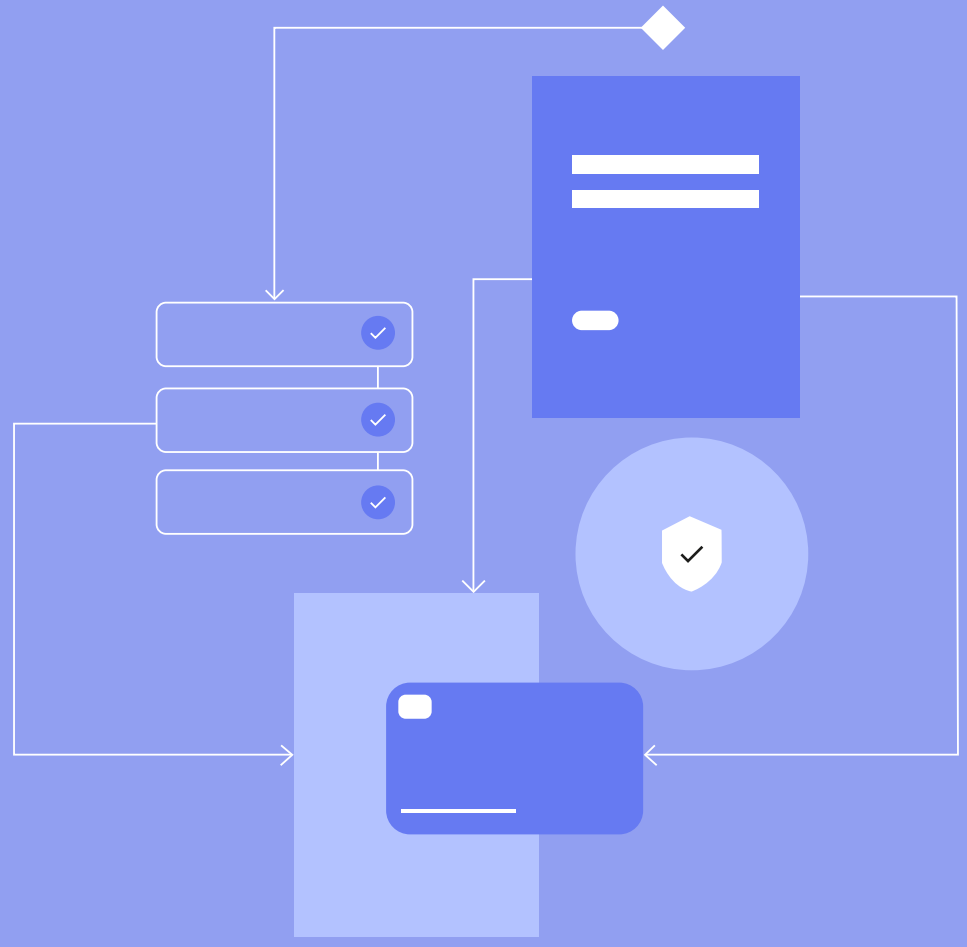
Seguridad en Aplicaciones de Misión Crítica: Caso ‘Uruguay se vacuna’

Whitepaper



GeneXus™

El 1º de marzo del año 2021 se activó en Uruguay la primera etapa del Plan de Vacunación contra el **COVID-19**, bajo el lema #Uruguaysevacuna.

Como en ese momento las vacunas eran limitadas, las autoridades gubernamentales decidieron priorizar a los grupos de riesgo, constituido por pacientes con morbilidades y por trabajadores que pudieran estar más expuestos a contraer el virus, como el personal de salud y de educación, policías, bomberos, entre otros. Luego seguirían con el resto de la población.

Para que la vacunación ocurriera de forma ordenada y segura, se creó con GeneXus una agenda digital accesible desde la web, desde la aplicación móvil CoronavirusUY (Google Play - App Store) y desde un chat.

El sistema debía desarrollarse en tiempo récord. Los casos positivos aumentaban, así como la tasa de mortalidad relacionada con esta enfermedad. Tener vacunas y no poder administrarlas, por fallas en el sistema, no era una opción. Por eso el software desarrollado con Low-Code para #Uruguaysevacuna se considera una **Aplicación de Misión Crítica.**

Fue así que en solo 2 semanas, un equipo multidisciplinario creó la primera versión de esta solución que permitió que 2 millones de personas, mayores de 18 años, y elegibles para recibir la vacuna de COVID-19, pudieran solicitar su agenda, vacunarse en la hora, lugar y fecha asignada y luego ser notificada para aplicarse la segunda dosis.

"El sistema debía soportar un alto nivel de transacciones. Lo que se esperaba era que en momentos puntuales se recibieran muchísimos pedidos a la vez, luego esta cantidad de solicitudes bajaría, pero podría subir ante distintos eventos y generar picos. Nuestro desafío consistió en desarrollar un sistema seguro y acorde con las legislaciones del Sistema Nacional Integrado de Salud de la República Oriental del Uruguay", explica Gerardo Canedo, ingeniero en Computación, Especialista en Seguridad Informática y Gerente de Seguridad en GeneXus Consulting.

Gente, ideas, herramientas ¿Qué hay detrás del Coronavirus UY?

Leer más

Un enfoque orientado a riesgos

El equipo:

✓Este sistema fue construido por personas que trabajaron en modalidad remota.

✓El tema de la seguridad recayó en todos, desde analistas de negocio, hasta testers, desarrolladores y arquitectos de software.

La estrategia:

Debían ser eficaces y eficientes. Para lograrlo, y mucho antes de empezar el proceso de la codificación, se enfocaron en la seguridad informática, identificando:

✓ Los riesgos mayores que tenían que mitigar.

✓ Los riesgos que no iban a mitigar, pero que debían saber cómo iban a gestionarlos.

Las tareas:

"De todas las actividades que podíamos hacer pensando en la seguridad, decidimos en esta primera iteración tomar en cuenta las cuatro con más valor agregado, que son el Modelado de Amenazas, el Análisis de Riesgo de la Arquitectura, la Definición de Requerimientos de Seguridad y el Testeo de Seguridad", detalla Canedo.

Modelado de Amenazas

Es el proceso por el cual se identifican las amenazas, las vulnerabilidades que pueden existir y cómo se podrían utilizar esas vulnerabilidades para llevar a cabo algún ataque.

Para esto, contestaron las siguientes preguntas:

-¿A quién le puede interesar atacar este sistema?
-¿Cuál o cuáles podrían ser los objetivos del ataque?
-¿Qué tipo de atacantes podría tener el sistema?
-¿Qué técnicas pueden ser utilizadas para atacar el sistema y cómo resolverlo?

"Debíamos implementar controles para mitigar o reducir la capacidad de impacto o de potencialidad de un ataque exitoso. En el juego de Modelado de Amenazas, buscamos entender quién estaría del otro lado intentando dañar el sistema".

De este análisis se desprendieron las siguientes conclusiones:

✓ El sistema debía restringirse al territorio nacional.

✓ Se debían implementar mecanismos de anti-automatización para las interfaces públicas hacia el mundo.

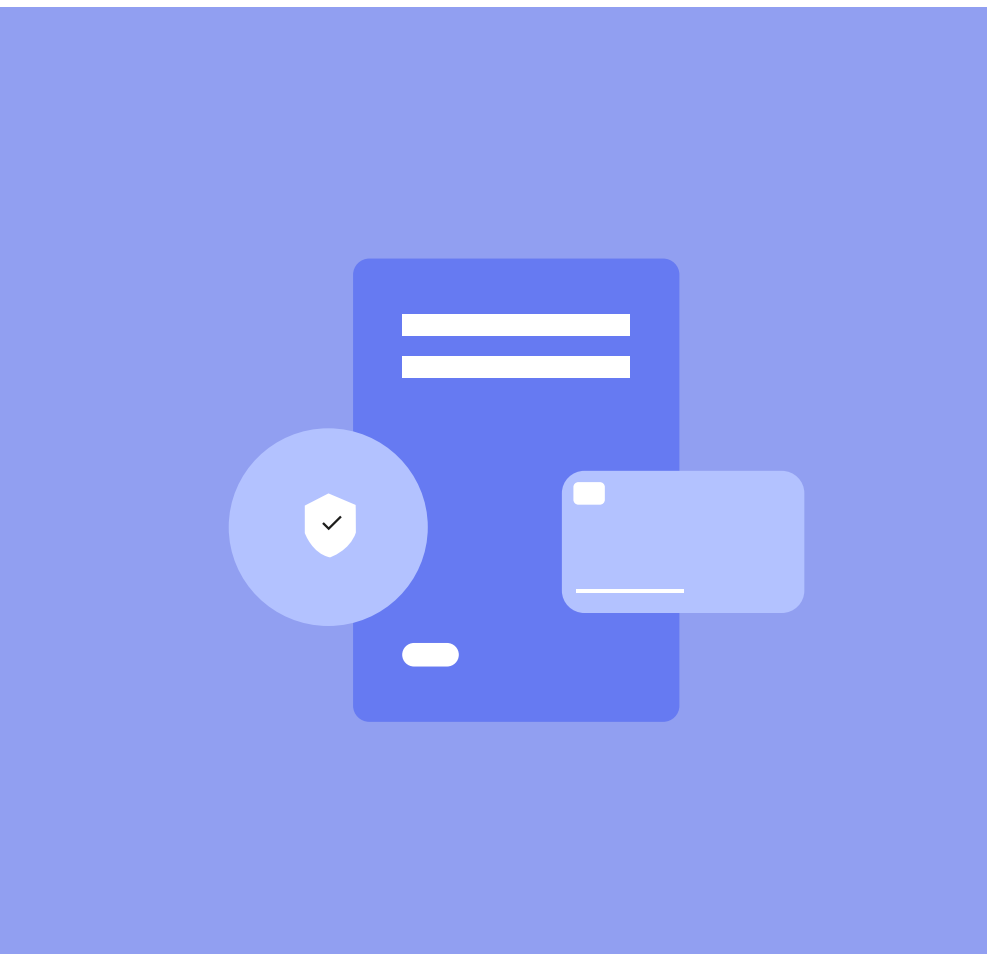
✓ No podían almacenar datos personales en la nube que iban a utilizar, ya que esta nube no se encuentra en Uruguay.

✓ Para identificar, registrar y monitorear de forma continua a una persona, debían solicitar la fecha de nacimiento y el número de cédula.

Análisis de Riesgo y de la Arquitectura

En esta tarea evaluaron la propuesta de la arquitectura desde el punto de vista de la seguridad, identificando los aspectos de interés y cómo podrían subsanarse los ataques.

"Lo que tuvimos como entrada fue un diagrama de arquitectura de alto nivel, en el cual no estaba el documento completo (porque aún no estaba armado), pero sí teníamos una idea de cómo iba a ser la arquitectura de esta solución. A partir de allí identificamos los flujos de información y definimos las fronteras del sistema, las zonas de confianza (que fueron dos), los flujos y las conexiones. Debíamos asegurarnos de que la comunicación entre esas zonas de confianza fuera segura. Para eso utilizamos la metodología **STRIDE**, logrando identificar las posibles vulnerabilidades y cómo podíamos mitigarlas", explica el también miembro del capítulo Uruguayo del Open Web **Application Security Projects (OWASP)**.



El problema: la nube pública

"Una de estas zonas es la nube pública que está fuera del territorio nacional. Para poder cumplir con los requerimientos y con las normativas del país, no podíamos almacenar allí información personal y teníamos que manejar la menor cantidad de datos posibles. Eso implicó un desafío de diseño y arquitectura. Debíamos almacenar información en la nube sin almacenar datos personales y aun así debíamos poder determinar si una persona estaba o no efectivamente habilitada para ser vacunada".

La solución

"Armamos una mejora a la arquitectura, definiendo una forma de almacenar la información de los datos personales. Para eso utilizamos algunos mecanismos de criptografía, como la función de hash.

Gracias a esta arquitectura refinada pudimos crear un documento que contenía el análisis de riesgo de cada uno de los componentes y el modelado de amenazas, cómo podían ejecutarse y las motivaciones que podrían tener los atacantes en cada caso".

Definición de Requerimientos de Seguridad

Los requerimientos de seguridad surgieron de las tareas anteriores y se plasmaron en un documento que se compartió con todo el equipo.

Testeo de Seguridad

"Uno de los controles más efectivos que implementamos fue la validación de tipo de datos en las APIs. Cada una de las entradas de datos de las APIs fueron variadas con respecto a ciertas expresiones regulares. Realizando esas validaciones, miramos de forma muy eficiente muchos ataques que se podrían dar en otros contextos.

Otra opción que implementamos fue la definición de Casos de Abuso. Básicamente hicimos hacking de la aplicación en papel, armando cómo sería el ataque que tendría que hacer un adversario para poder dañar el sistema. Con esa información comenzamos a realizar tests de seguridad cada vez que se liberaba algún componente. Y aquí es donde los casos de abuso llegaron a su uso. Tomamos aquellos casos que habíamos visto que no podían suceder y probamos que efectivamente no sucedían en el sistema.

A partir de esa información pudimos probar que los riesgos que más nos importaban no podían ser materializados de forma razonable. Como resultado, pudimos salir con la primera versión del sistema en el tiempo planificado, con un nivel de seguridad conocido, con riesgos mitigados y riesgos asumidos".

Para Canedo, es utópico creer que un sistema puede estar libre de riesgo. Sin embargo, este proceso les permitió:

✓ Conocer el nivel de seguridad de la aplicación.

✓ Conocer las fortalezas y debilidades del sistema.

✓ Determinar los controles que tenían, y cómo proceder en caso de un ataque.

Video

Para conocer más detalles de este tema, te invitamos a ver la charla Seguridad de las Aplicaciones en Software de Misión Crítica: Un enfoque proactivo y rentable, dirigida por Canedo en la pasada edición del GeneXus Live.

Conceptualizando una API para la app CoronavirusUy

En Uruguay, cuando inició la pandemia en marzo del 2020, se planificó, a través del Plan Nacional Coronavirus, la posibilidad de ofrecer distintos métodos de contacto entre la ciudadanía y los prestadores de salud, para coordinar test COVID-19. De esta manera surgieron la aplicación CoronavirusUy, y los chats a través del sitio del Ministerio de Salud Pública, Facebook Messenger y WhatsApp.

En 2021, cuando comenzó el proceso de agenda digital para vacunarse contra el COVID-19, se definió usar los mismos canales de comunicación. La solución se convirtió en una Aplicación de Misión Crítica, por la cantidad de personas que iban a intentar agendarse en un mismo momento.

"Debíamos pensar una arquitectura completamente distinta. Necesitábamos un mecanismo, una arquitectura asíncrona donde todas estas personas, accediendo desde distintas aplicaciones y distintas interfaces, pudieran agendarse y que eso no saturara los sistemas que ya existían a nivel de Ministerio de Salud Pública. Lo que hicimos fue conceptualizar una API, separar lo que es el proceso de agenda del proceso de chequear si ya la persona estaba agendada y en qué estado se encontraba", explica Eugenio García, Gerente de Producto de GeneXus For SAP Systems.

¿El objetivo? Permitir que todas las personas pudieran agendarse sin saturar el sistema. "Para eso se hizo una capa de mediación, donde esa información iba, a su vez, siendo almacenada en la infraestructura de Amazon SQS. Después, a través de GeneXus, se construyó una capa de Business Logic que iba leyendo la información y pasando paulatinamente al sistema de agenda electrónica. De esta forma, cada vez que se iba confirmando, se almacenaba el estado de cada una de esas agendas en una estructura de datos DynamoB, más pensada para estos sistemas, donde se necesita gran escalabilidad también a nivel de lectura".

Para saber más sobre las APIs en el mundo GeneXus, no dejes de ver la charla Innovando en la Economía API con GeneXus, ofrecida por Eugenio García en el marco del GeneXus LIVE. En la presentación explica, además, cómo trabajar con GeneXus en distintos escenarios de integración: por un lado, trayendo información de un API de un tercero y por el otro, exponiendo con GeneXus el API que otros puedan usar.

GeneXus™

MONTEVIDEO - URUGUAY	Av. Italia 6201- Edif. Los Pinos, P1	(598) 2601 2082
CIUDAD DE MÉXICO - MÉXICO	Hegel N° 221, Piso 2, Polanco V Secc.	(52) 55 5255 4733
MIAMI - USA	8950 SW 74th Ct, Suite 1406	(1) 201 603 2022
SÃO PAULO - BRASIL	Rua Samuel Morse 120 Conj. 141	(55) 11 4858 0300
TOKYO - JAPAN	2-27-3, Nishi-Gotanda	(81) 3 6303 9381
	Shinagawa-ku, Tokyo, 141-0031	(81) 3 6303 9980