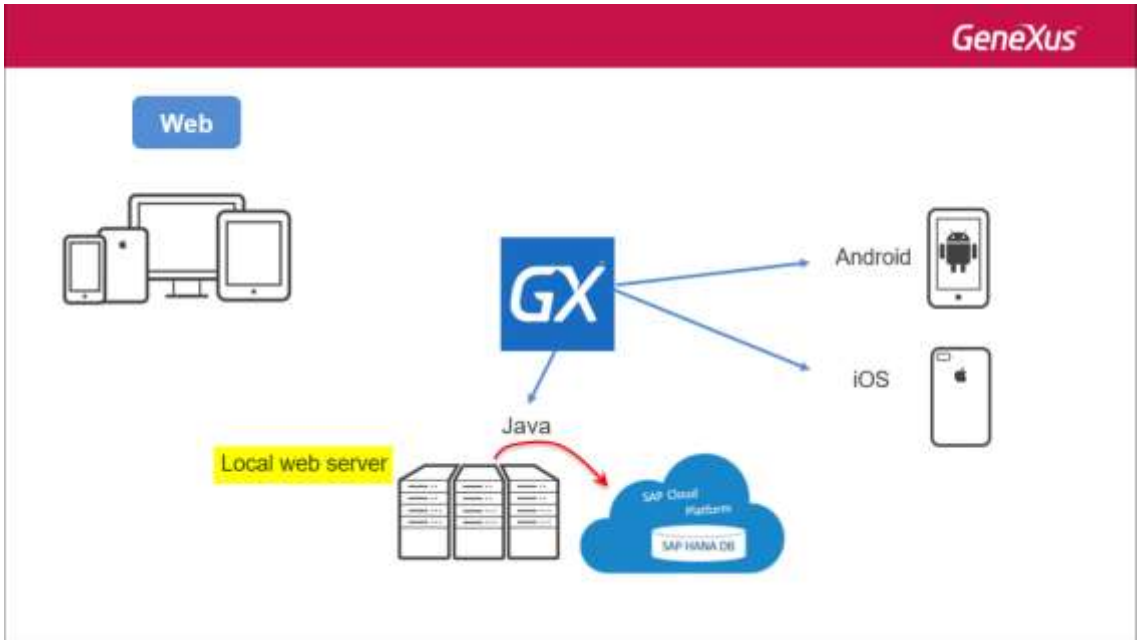
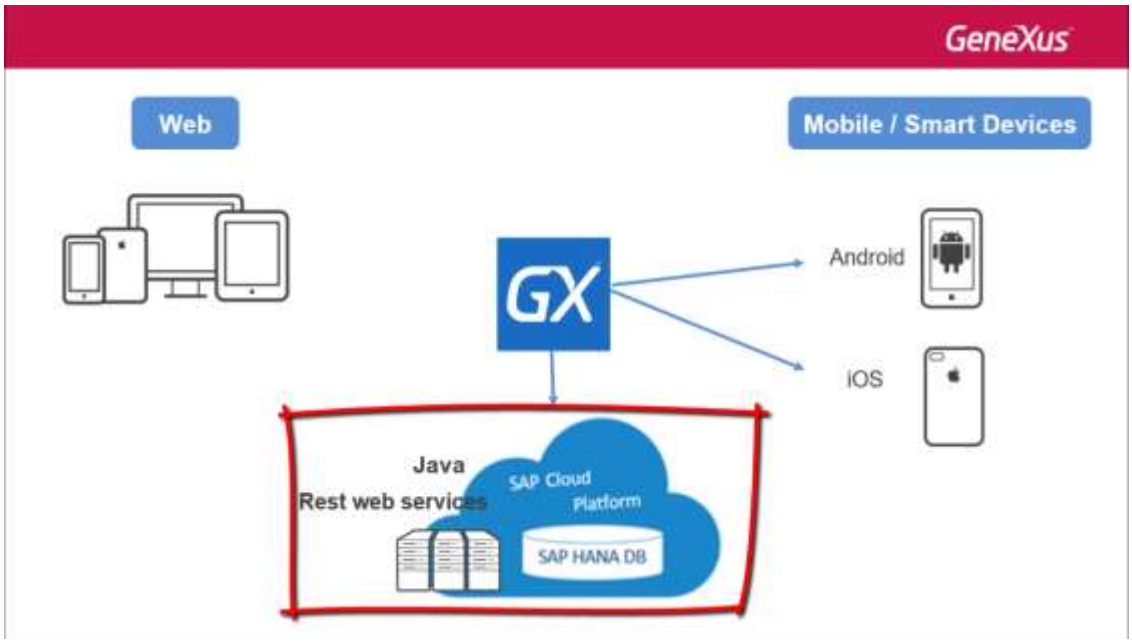
Ambiente de Test en SAP Cloud Platform and Puesta en Producción

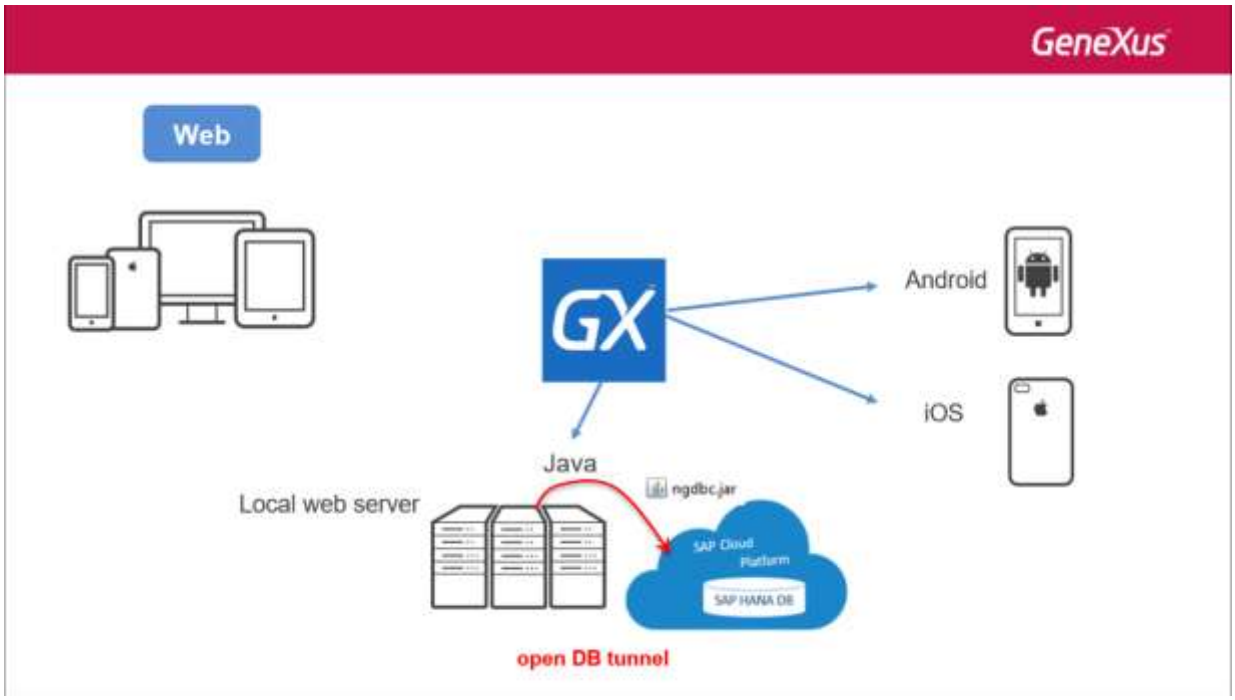
Hasta ahora estábamos prototipando en un servidor web local.



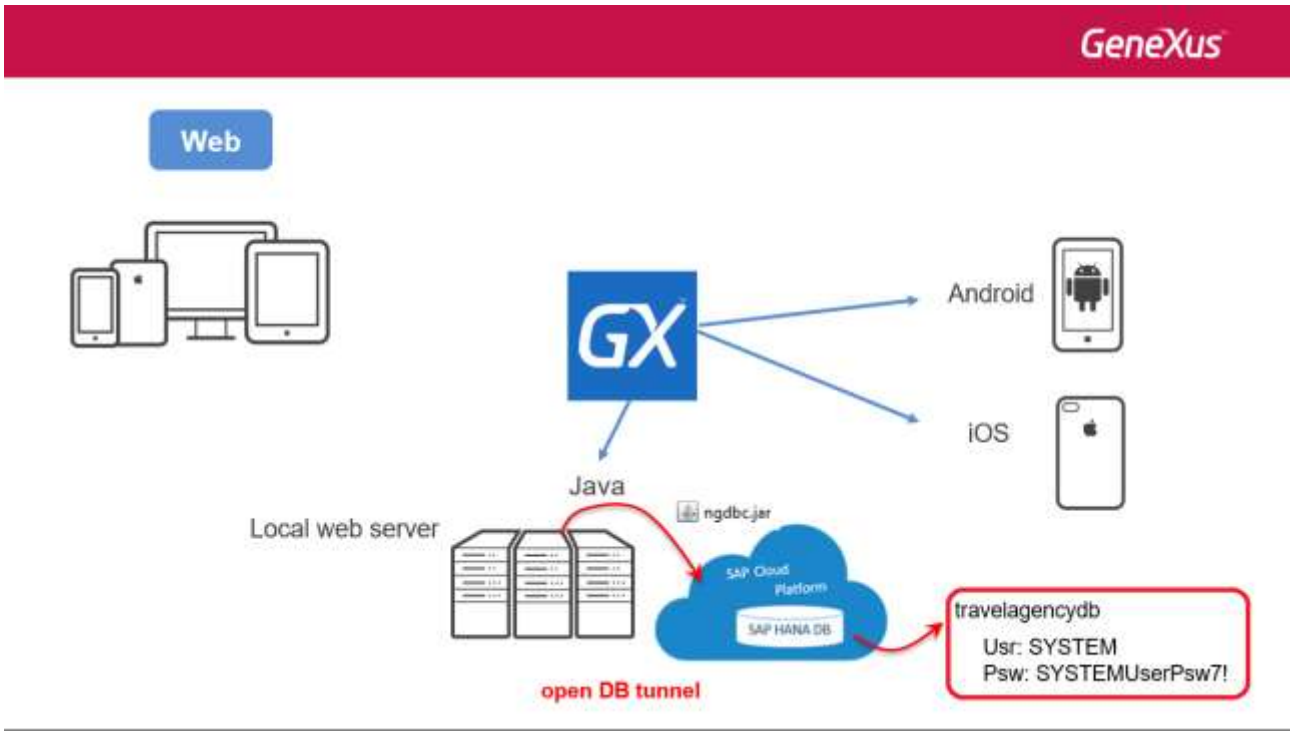
Supongamos que queremos armar un ambiente de test en SAP Cloud Platform utilizando la misma base de datos. ¿Cómo lo logramos?



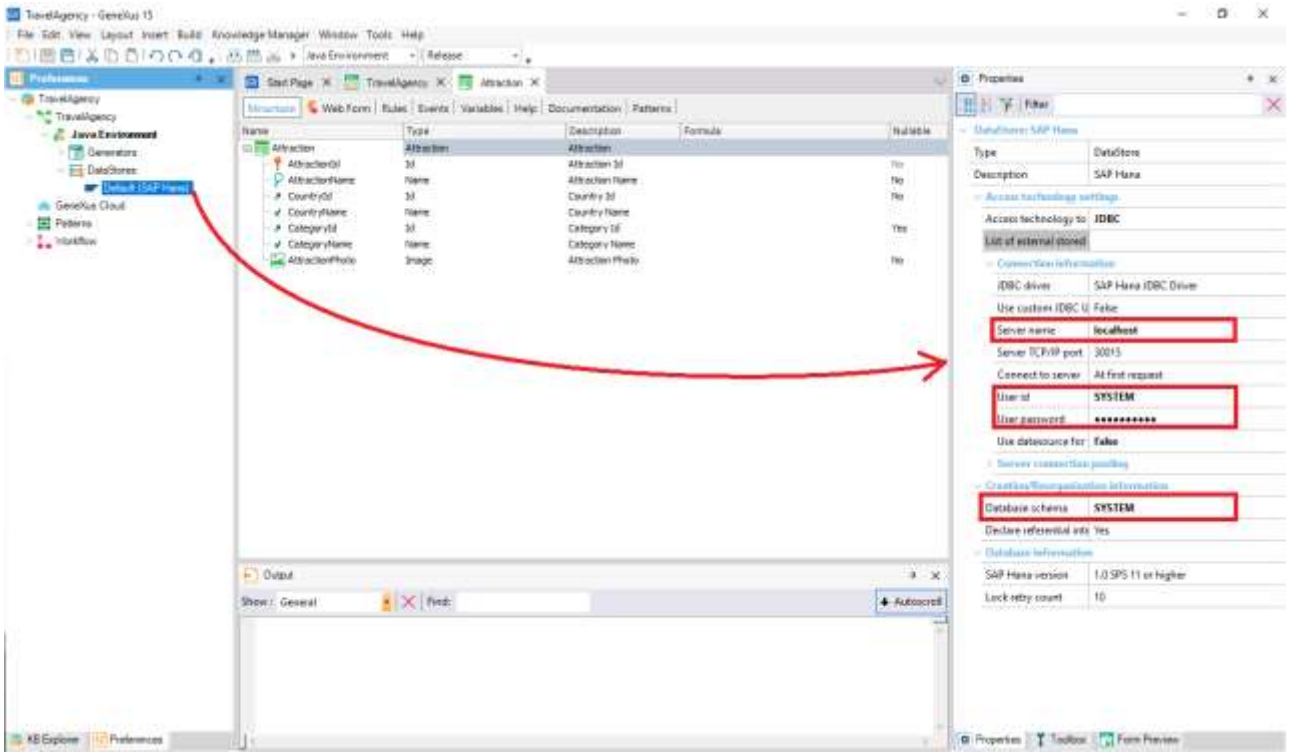
En nuestro environment, para poder conectarnos desde el Tomcat local a la base de datos Hana en SAP Cloud Platform utilizábamos el driver provisto, teniendo que abrir un túnel a la base de datos.



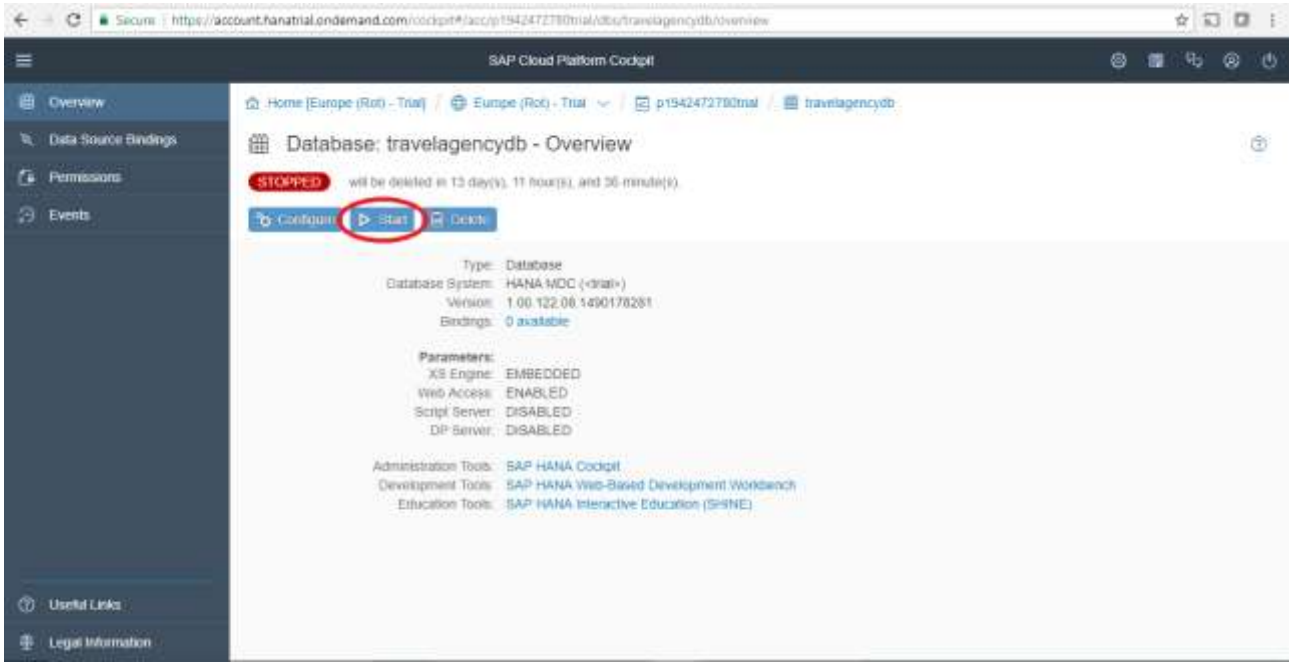
Base de datos que en nuestro caso habíamos creado en la versión trial provista por SAP Cloud Platform, utilizando el usuario SYSTEM, al que le asignábamos una password al momento de crear la base de datos.

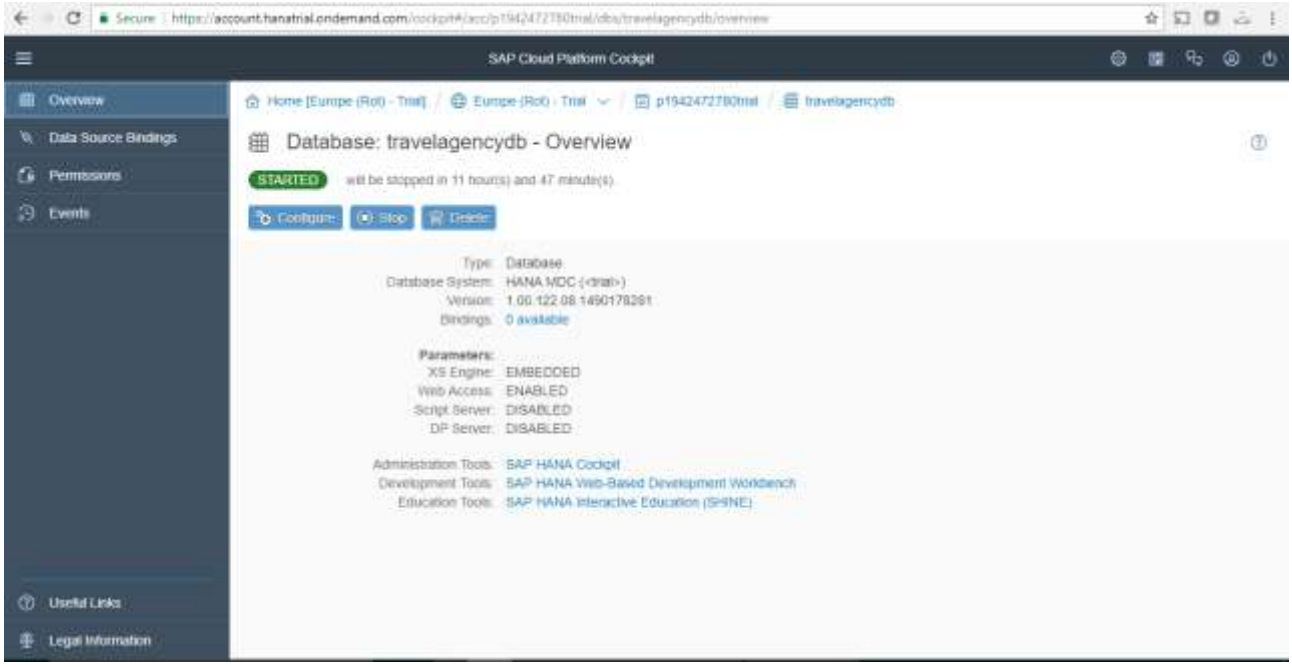


Luego en las propiedades del Data Store ingresábamos estos valores:

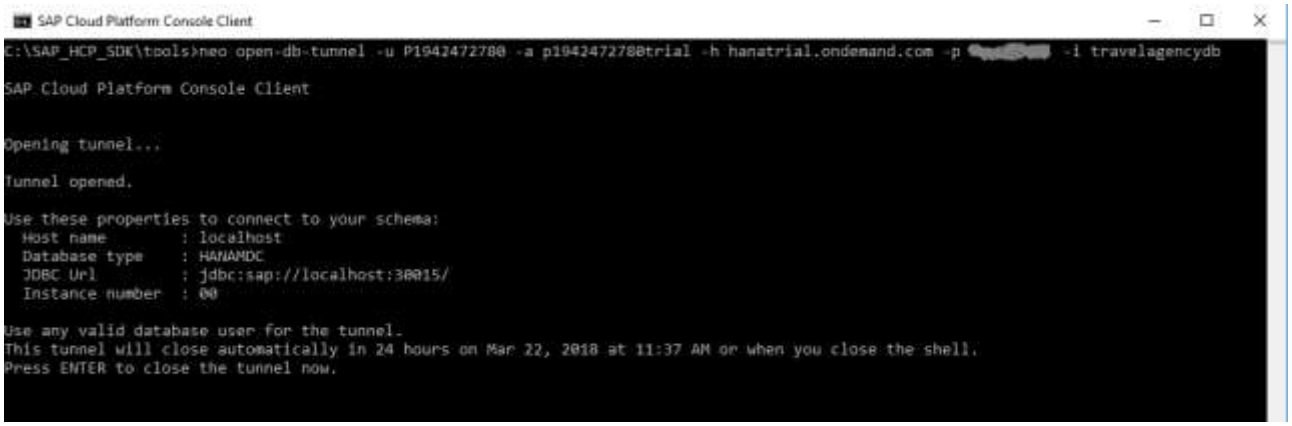


Antes de abrir el túnel necesitábamos tener levantada la base de datos:

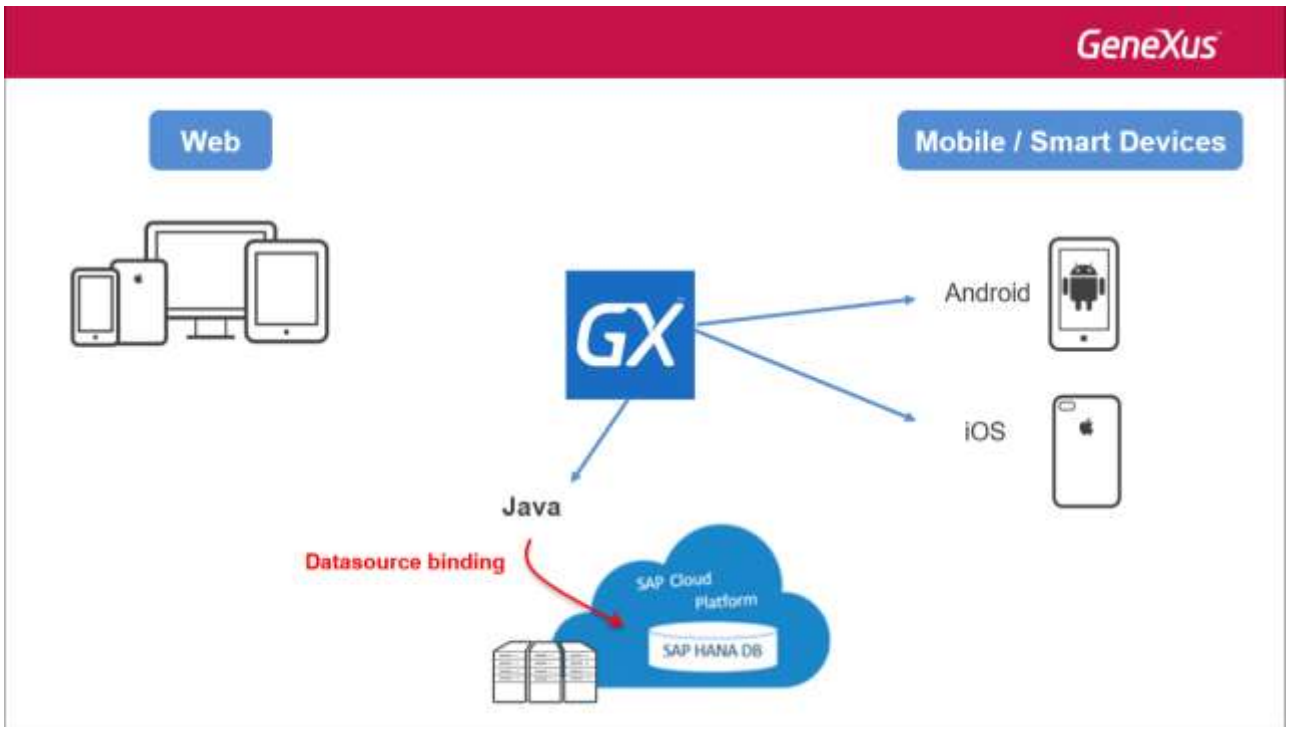




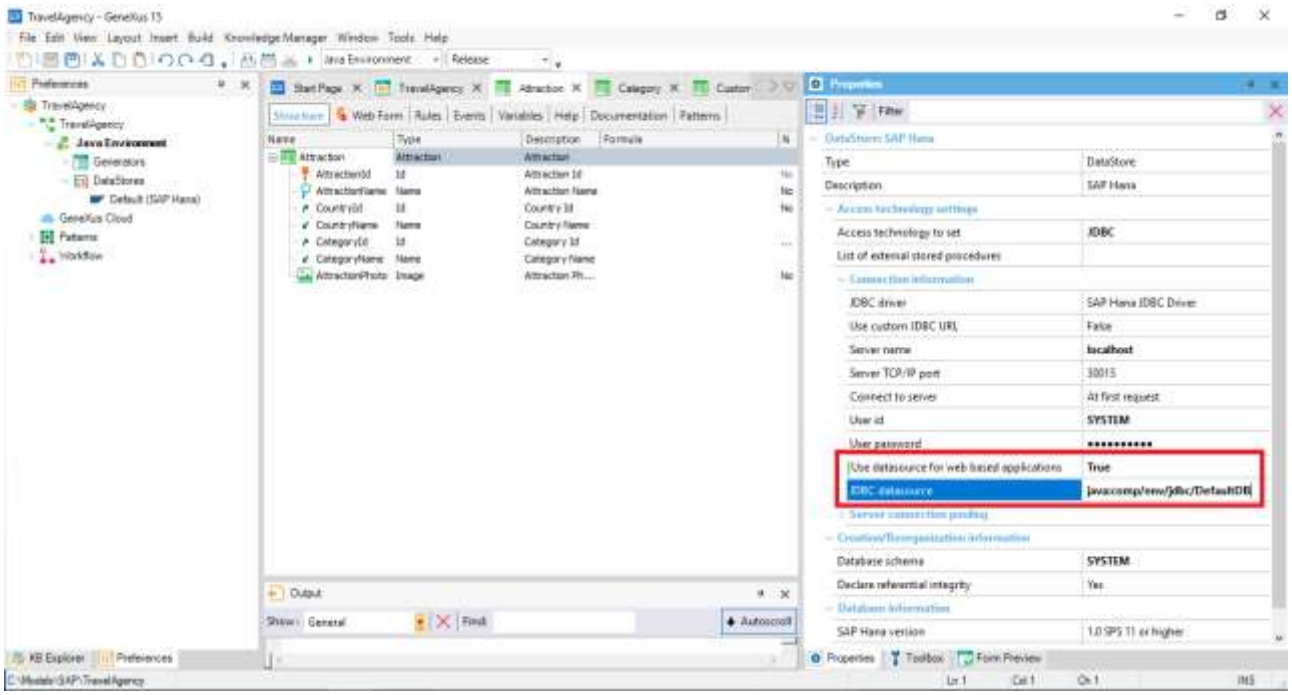
Para abrir el túnel necesitábamos los valores de ID de usuario en el SAP Cloud Platform, de la cuenta, password de ese usuario web, host (en nuestro caso hanatrial.ondemand.com) y nombre (id) de la base de datos (en nuestro caso, travelagencydb).



Ahora subiremos al servidor web de la SAP Cloud Platform los programas construidos por GeneXus para esta aplicación, que accederán a la base de datos en esa misma plataforma, por lo que ya no será necesario abrir ningún túnel, y el acceso a la base de datos se hará a través del mecanismo de JNDI, para lo cual se hará un Datasource Binding entre la aplicación y la base de datos utilizada.

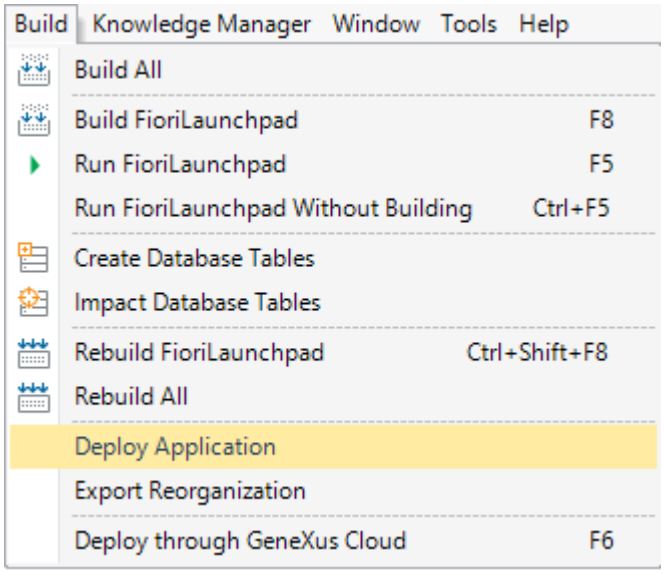


Por ello, pasamos a True el valor de la propiedad “Use data source for web based applications”.
Al hacerlo se habilita esta nueva propiedad, “JDBC data source”, donde debemos especificar “java:comp/env/jdbc/DefaultDB”, que es la forma en que SAP Cloud Platform encuentra la base de datos que se asociará a la aplicación.

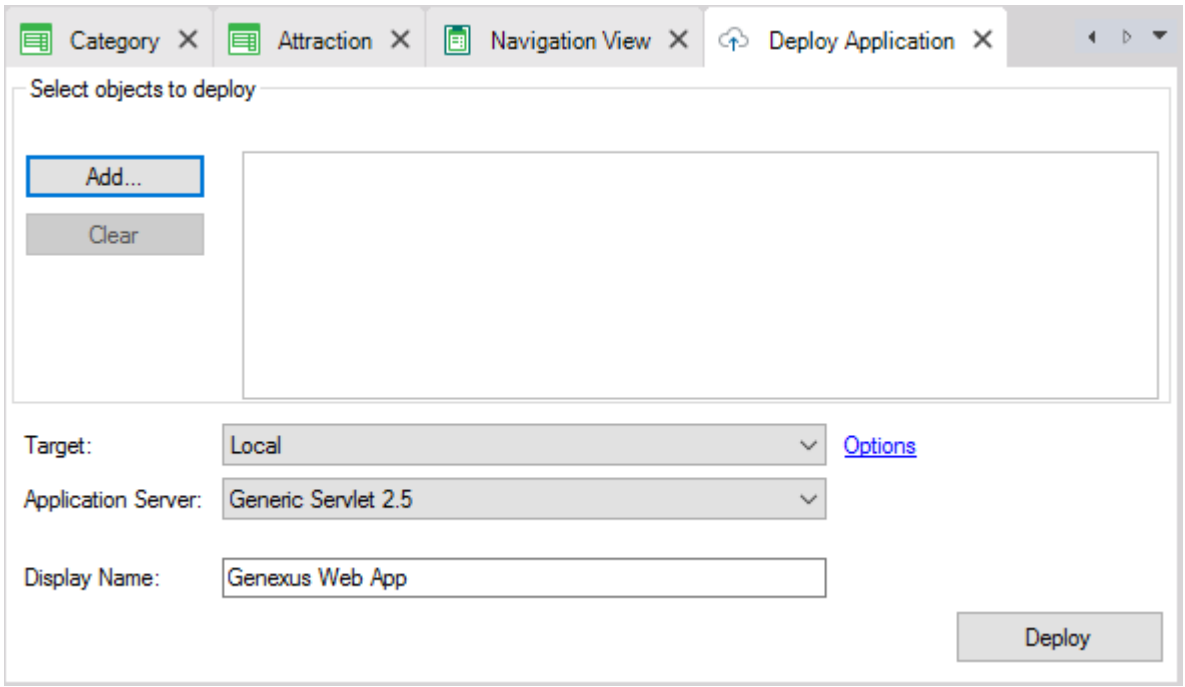


Nos basta ahora con hacer un **Build All**, para que los programas se construyan teniendo en cuenta ese datasource.

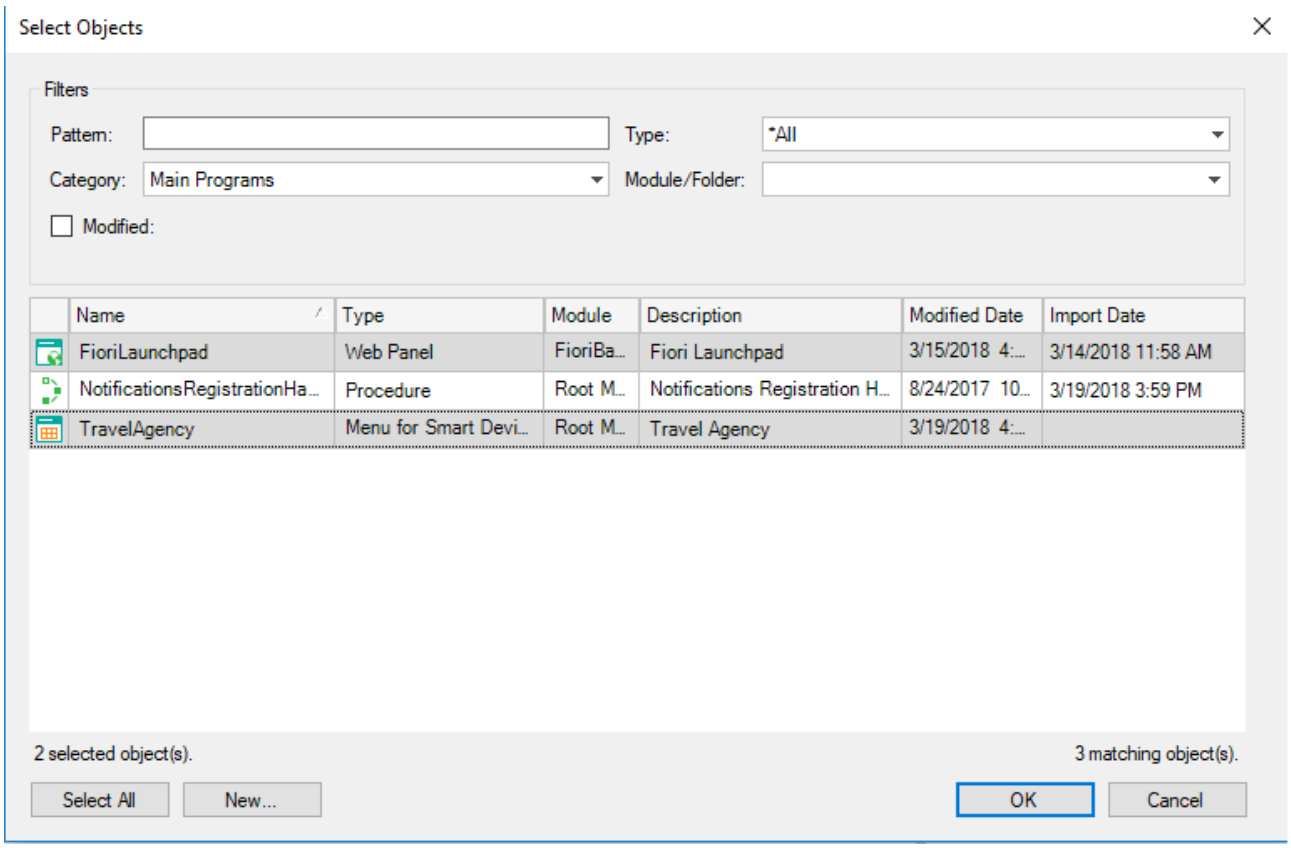
Luego, vamos al menú Build y elegimos Deploy Application:



Primero que nada seleccionamos los objetos main que queremos poner en producción:

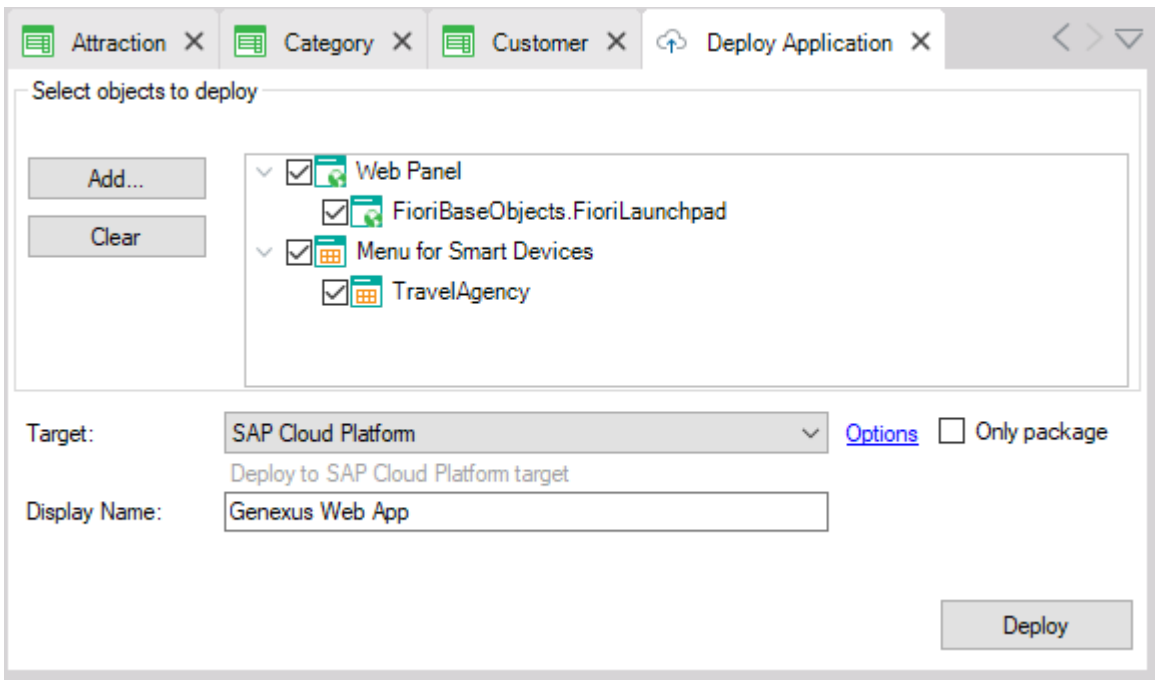


En nuestro caso será el launchpad para la aplicación web, y el análogo pero para Smart Devices:

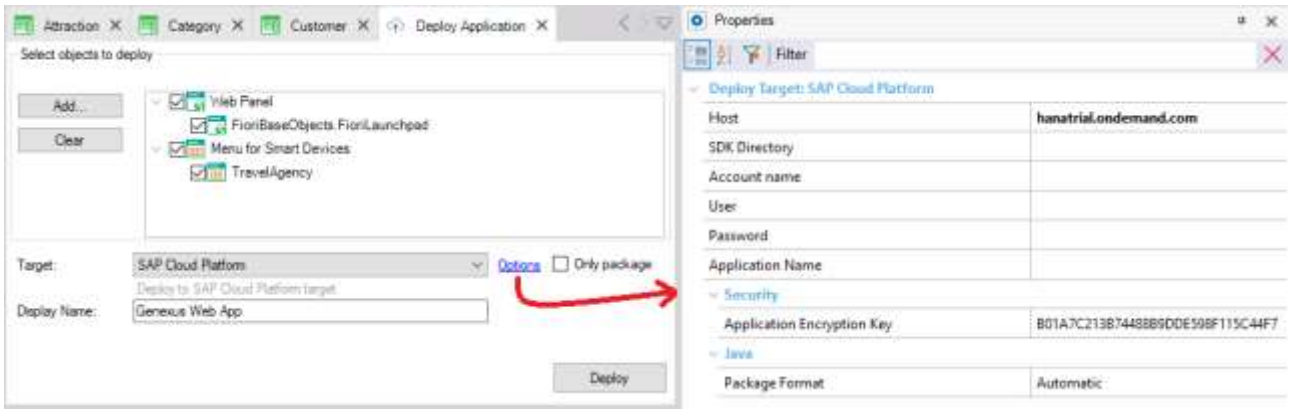


Este último es incluido porque necesitamos poner en producción también los servicios Rest que la aplicación móvil utilizará para recuperar y modificar información de la base de datos.

Ahora en Target elegimos SAP Cloud Platform…

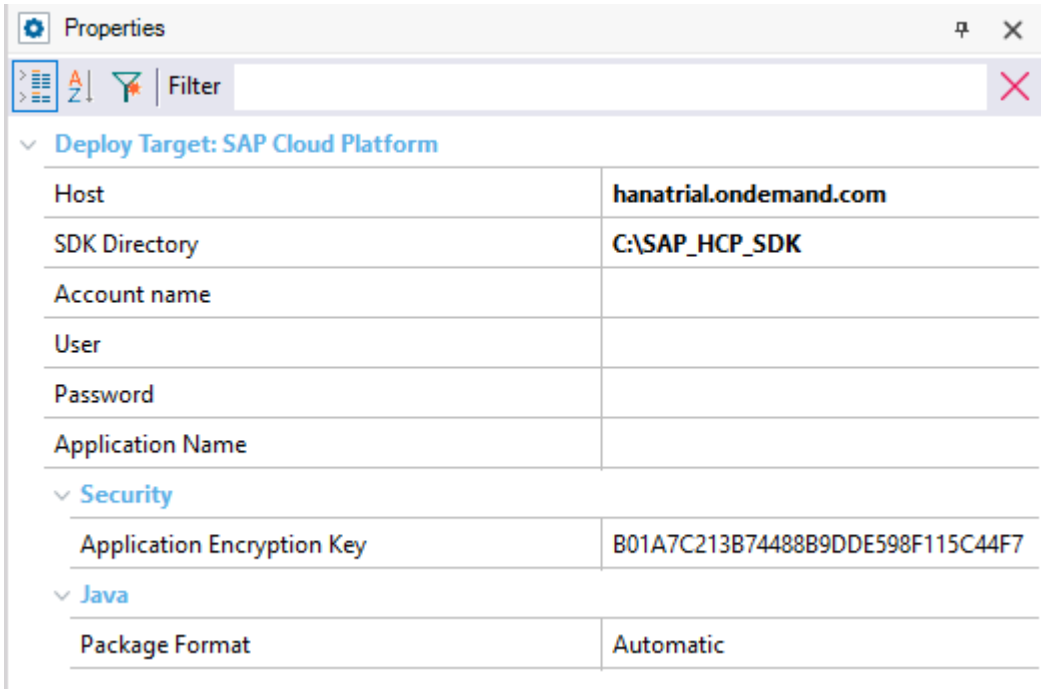


Y debemos configurar los valores de estas propiedades:

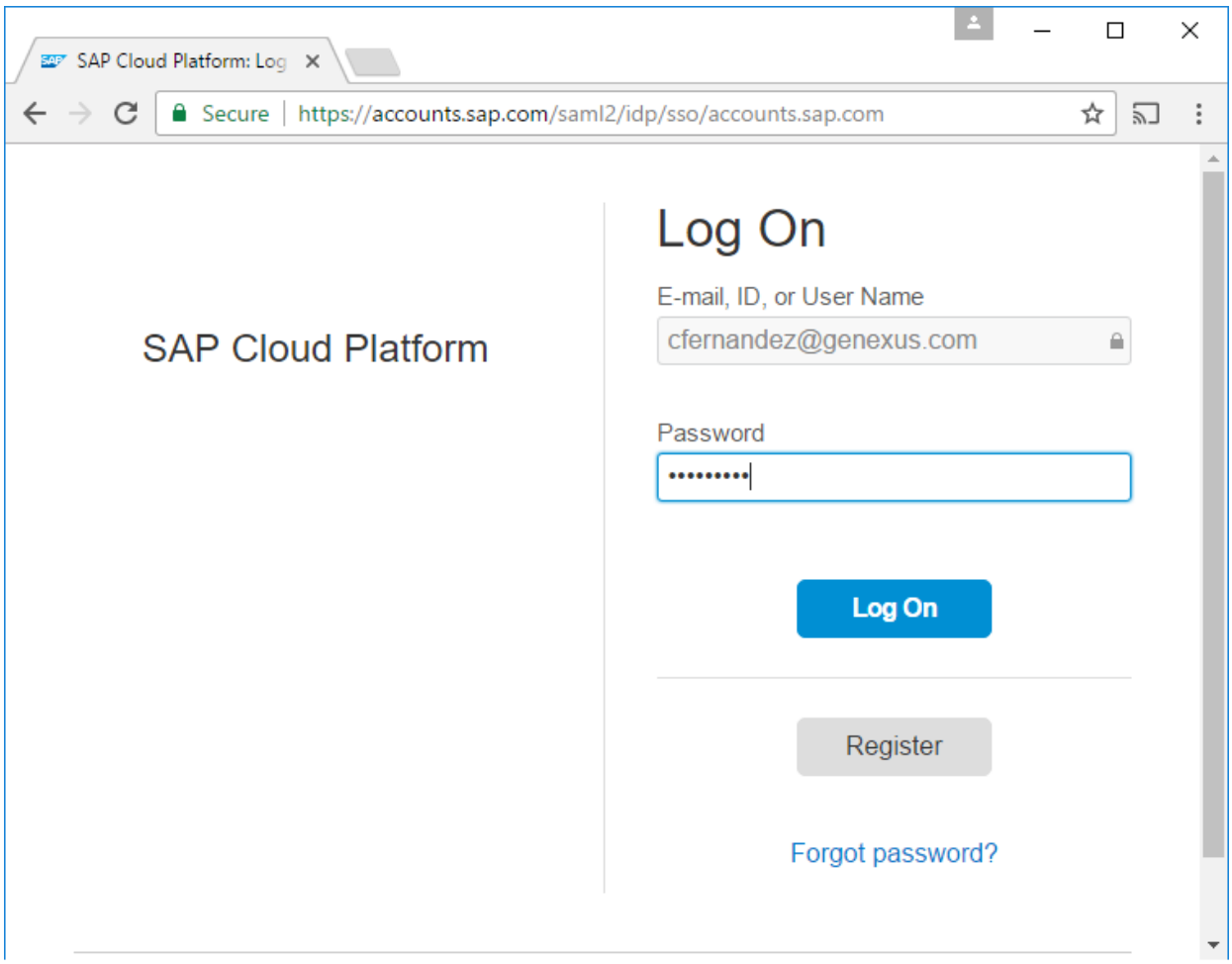


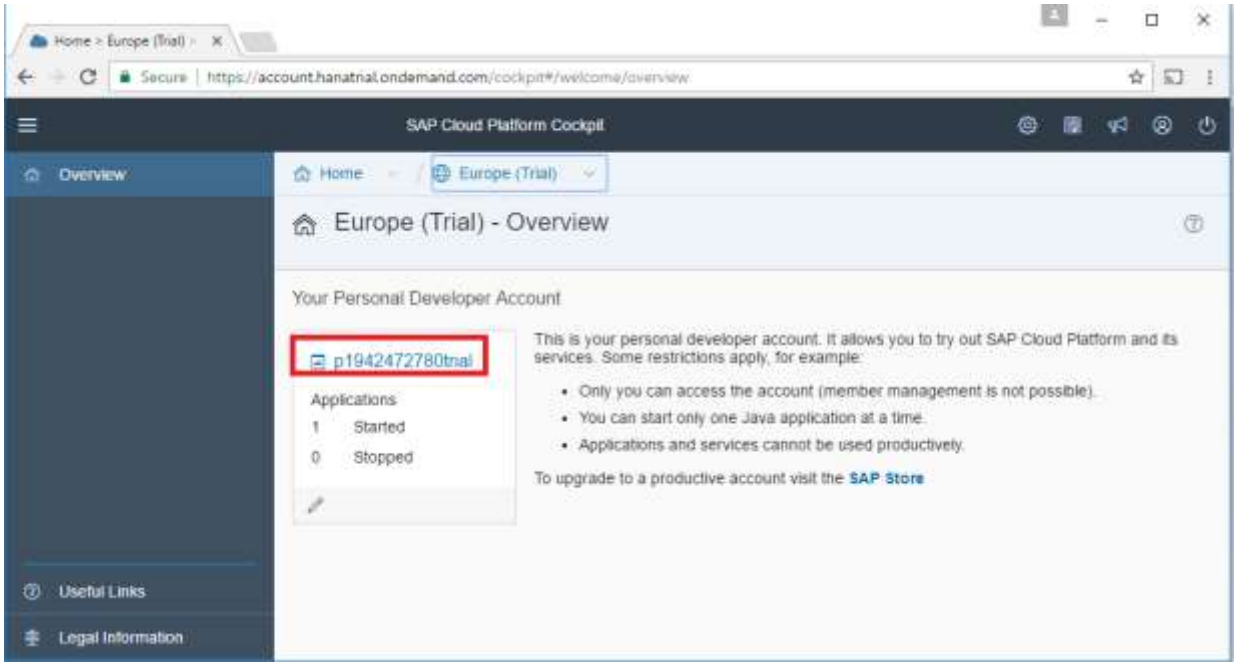
En Host debemos especificar el servidor contratado en SAP Cloud Platform. Como estamos usando una versión trial, especificaremos hanatrial.ondemand.com.

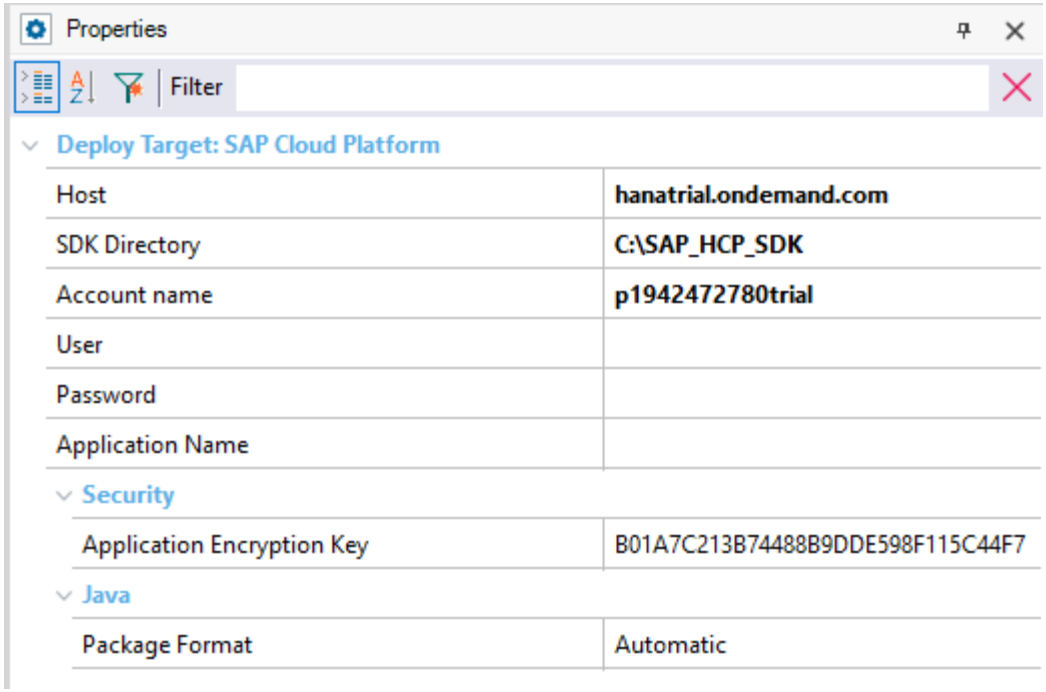
Luego, debemos especificar dónde se encuentra el directorio del SDK de SAP Cloud Platform. En nuestro caso es:



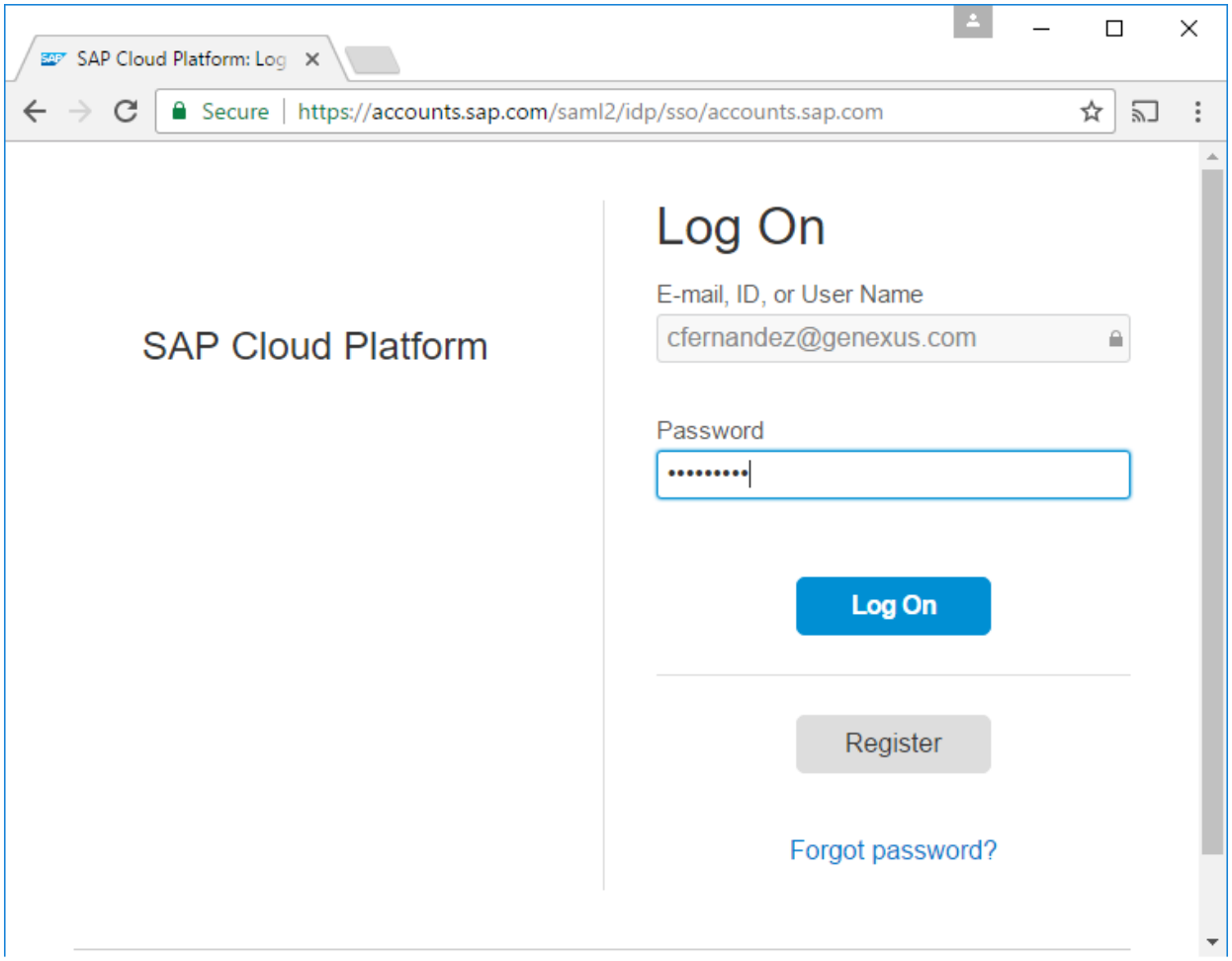
“Account name” es el nombre de nuestra cuenta en SAP Cloud Platform Cockpit. Es la misma que habíamos utilizado para abrir el túnel:

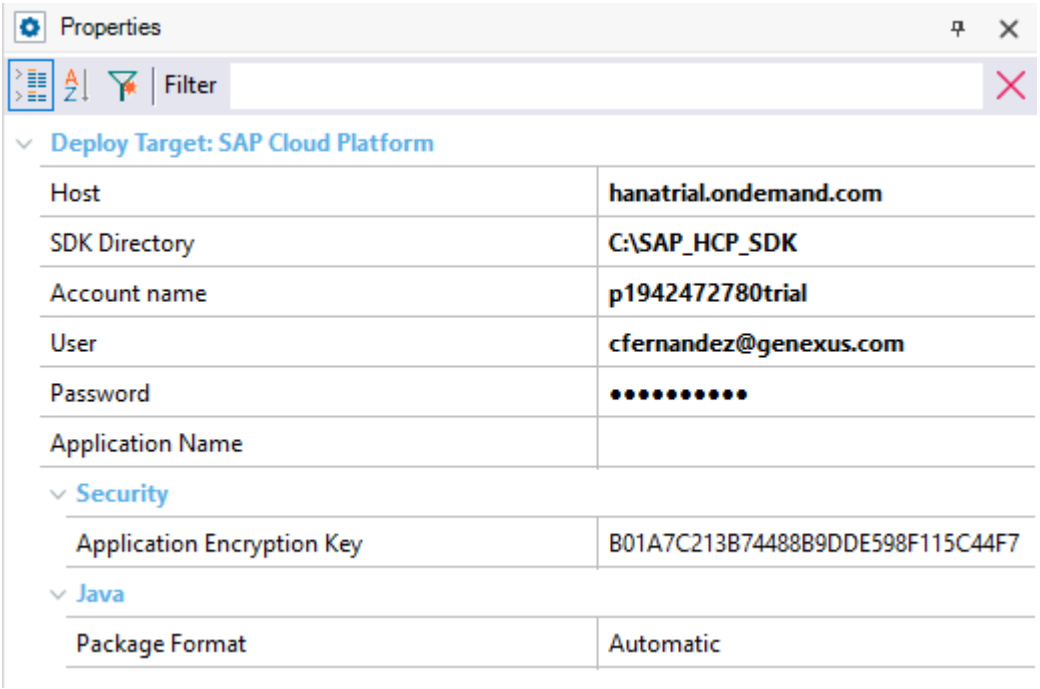

SAP Cloud Platform: Log
Secure | https://accounts.sap.com/saml2/idp/sso/accounts.sap.com
SAP Cloud Platform
Log On
E-mail, ID, or User Name
cfernandez@genexus.com
Password
Log On
Register
Forgot password?


SAP Cloud Platform Cockpit
Overview
Home
Europe (Trial)
Europe (Trial) - Overview
Your Personal Developer Account
p1942472780trial
Applications
1 Started
0 Stopped
This is your personal developer account. It allows you to try out SAP Cloud Platform and its services. Some restrictions apply, for example:
Only you can access the account (member management is not possible).
You can start only one Java application at a time.
Applications and services cannot be used productively.
To upgrade to a productive account visit the SAP Store
Useful Links
Legal Information

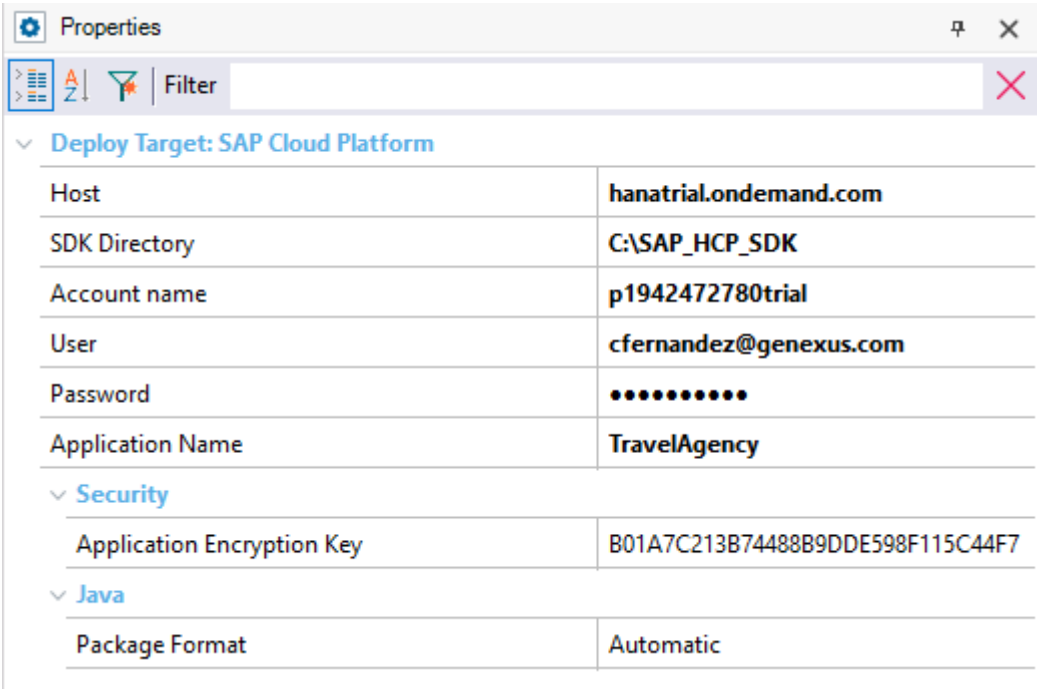


User y Password son aquellos con los que nos registramos a SAP Cloud Platform:

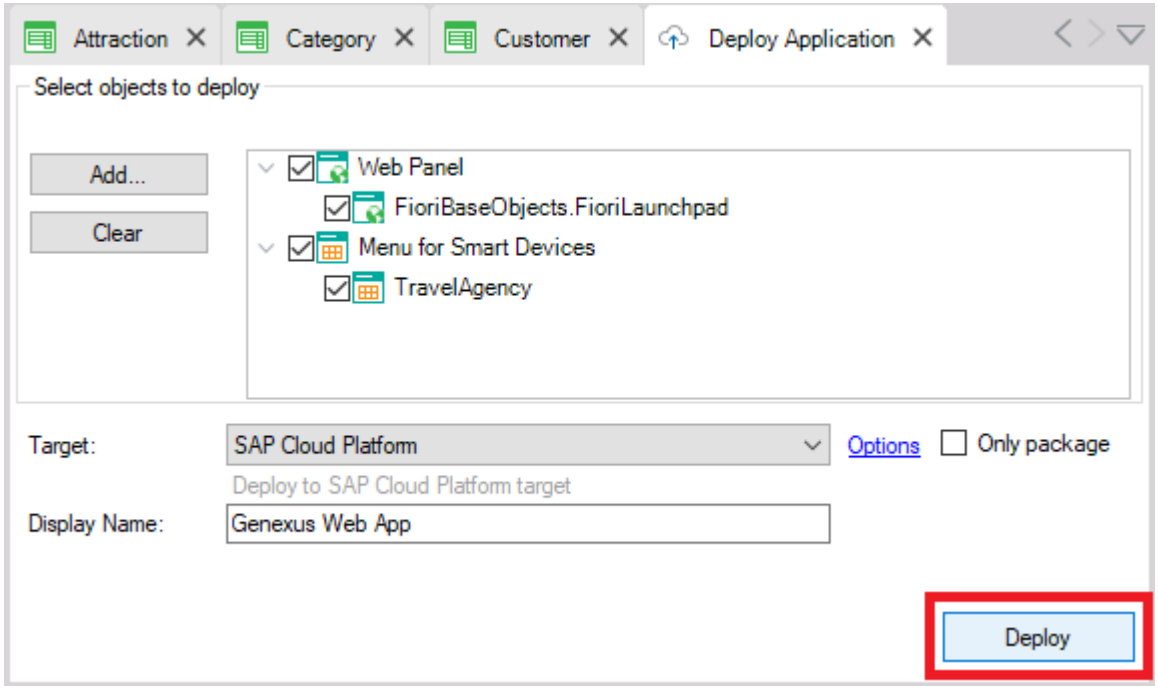




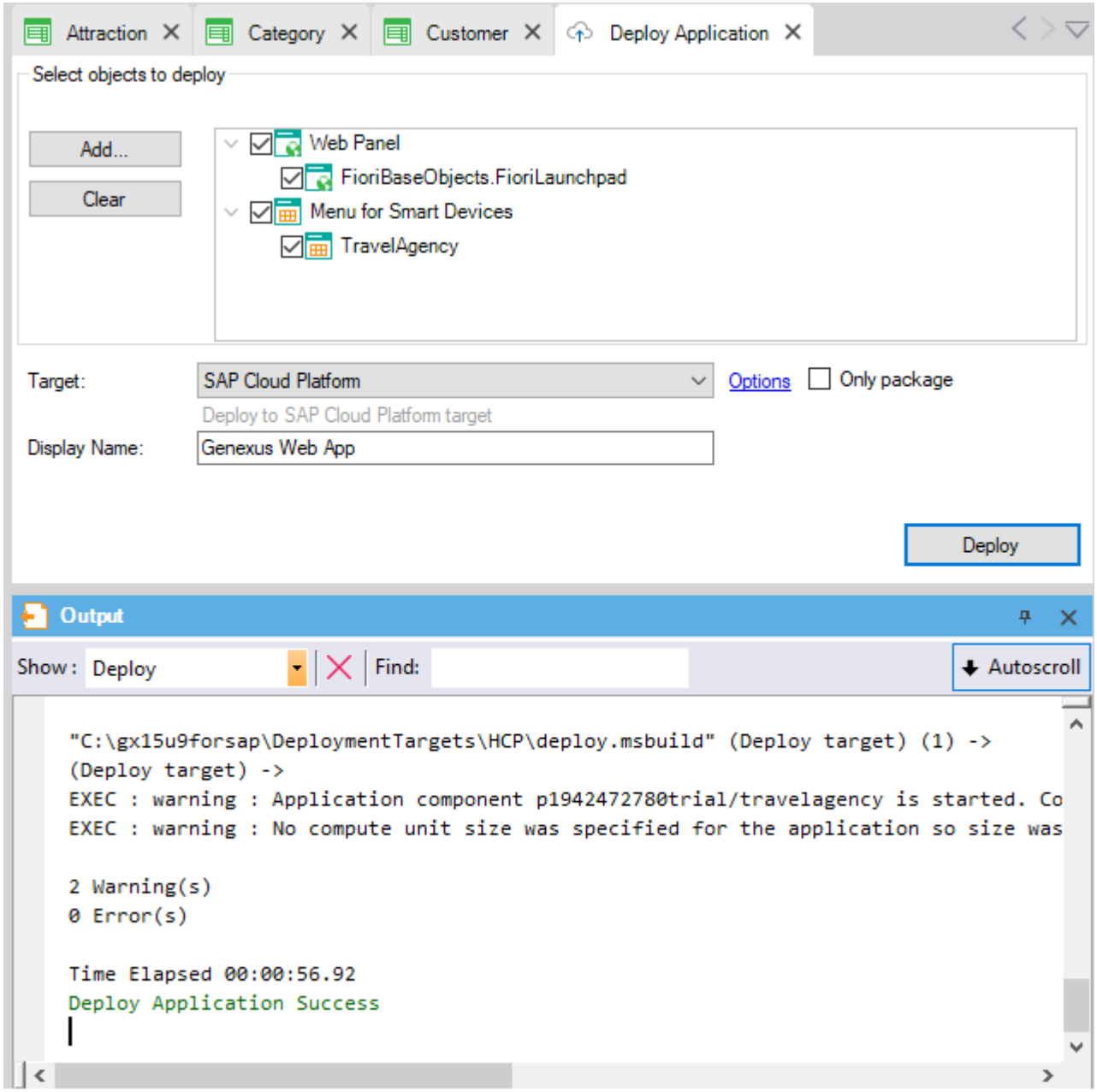
Y por último Application Name será el nombre utilizado para desplegar la app en SAP Cloud Platform.



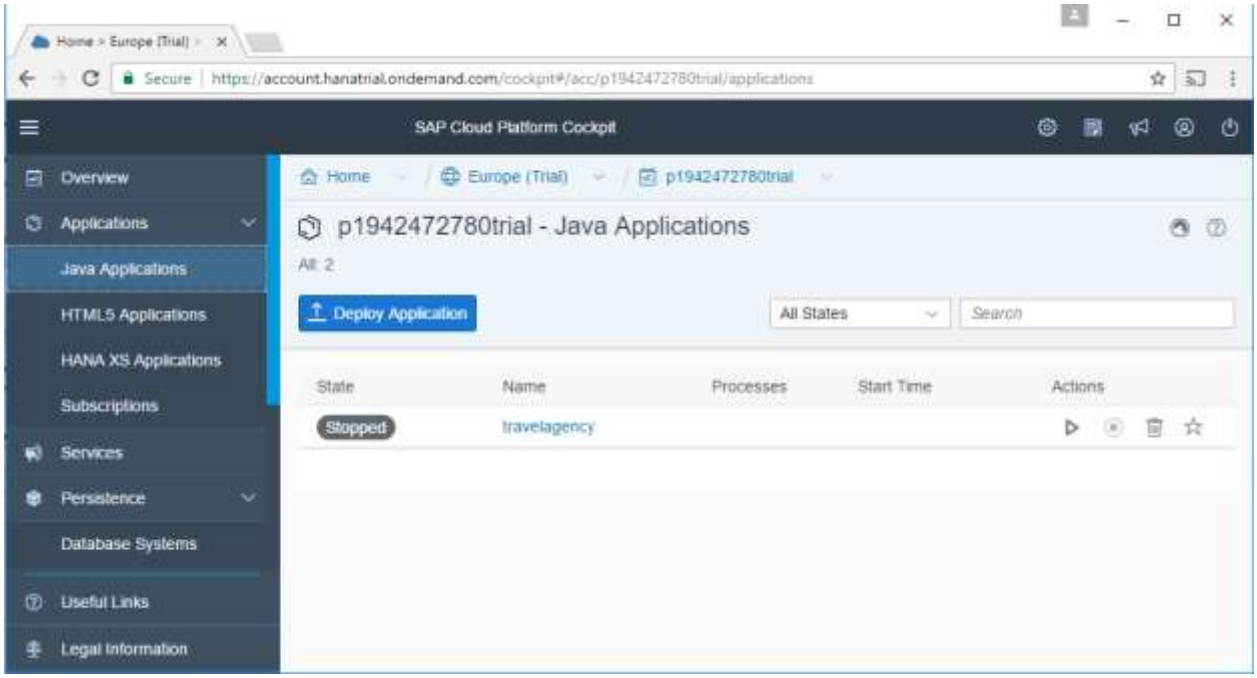
Ahora presionemos el botón Deploy, para que se construya el paquete WAR y se lo suba y se haga el deploy en SAP Cloud Platform.



Esperamos en la ventana de output ver el resultado exitoso del proceso:

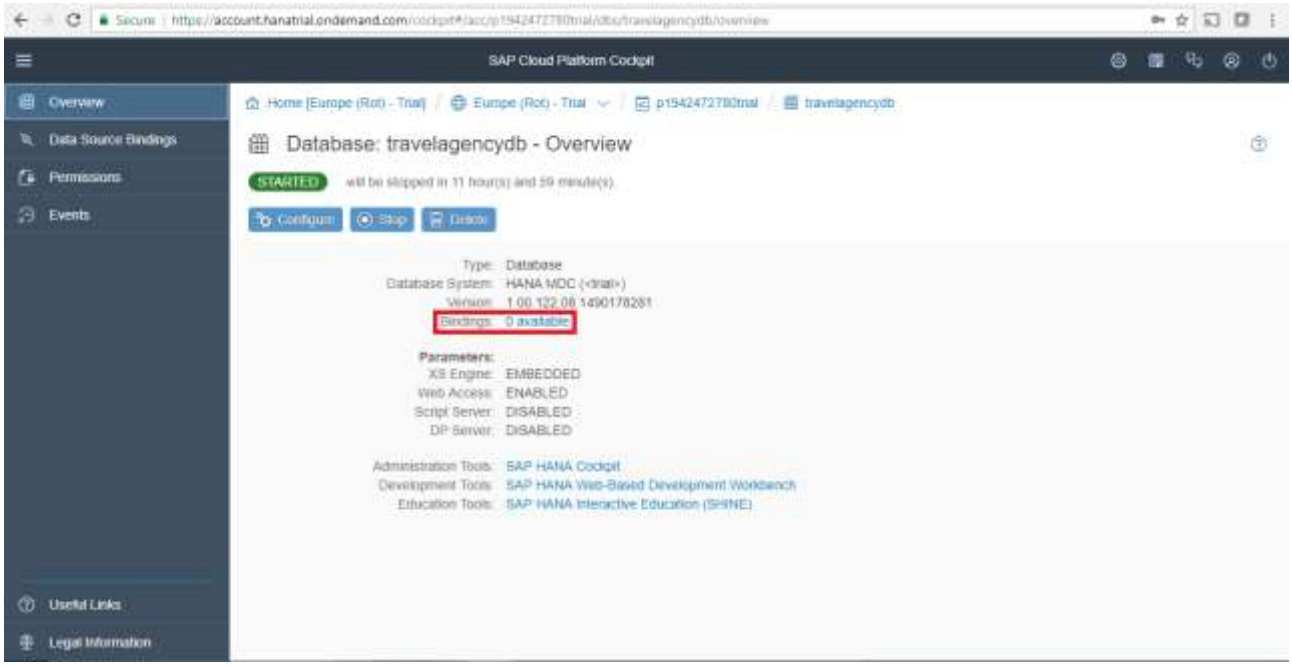


Volvemos a nuestra cuenta en SAP Cloud Platform Cockpit, y entre las Java Applications encontramos la que acabamos de subir, que está detenida:

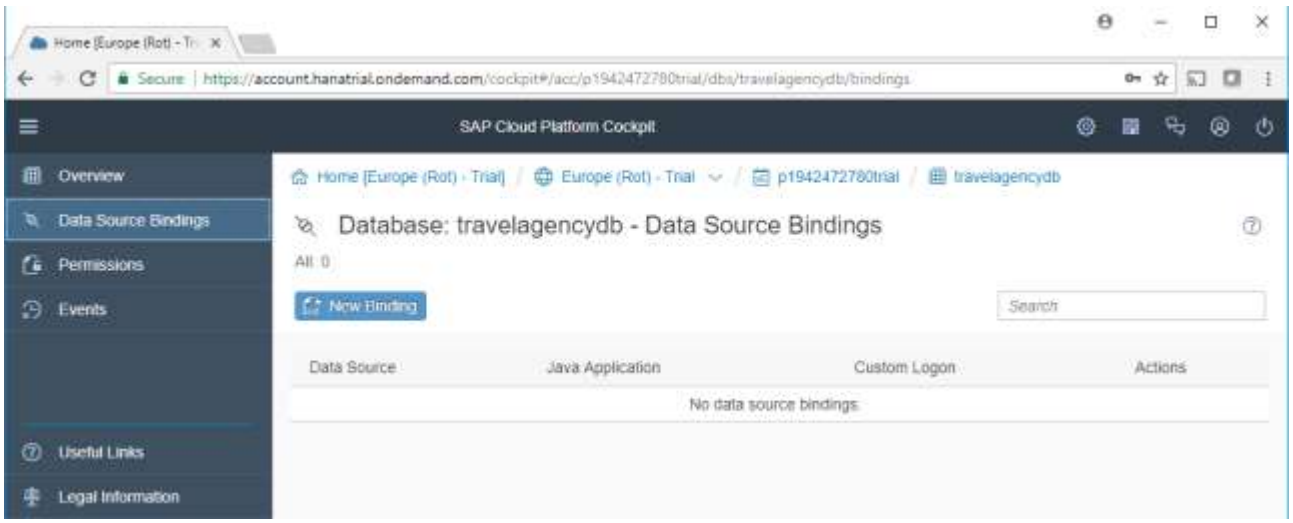


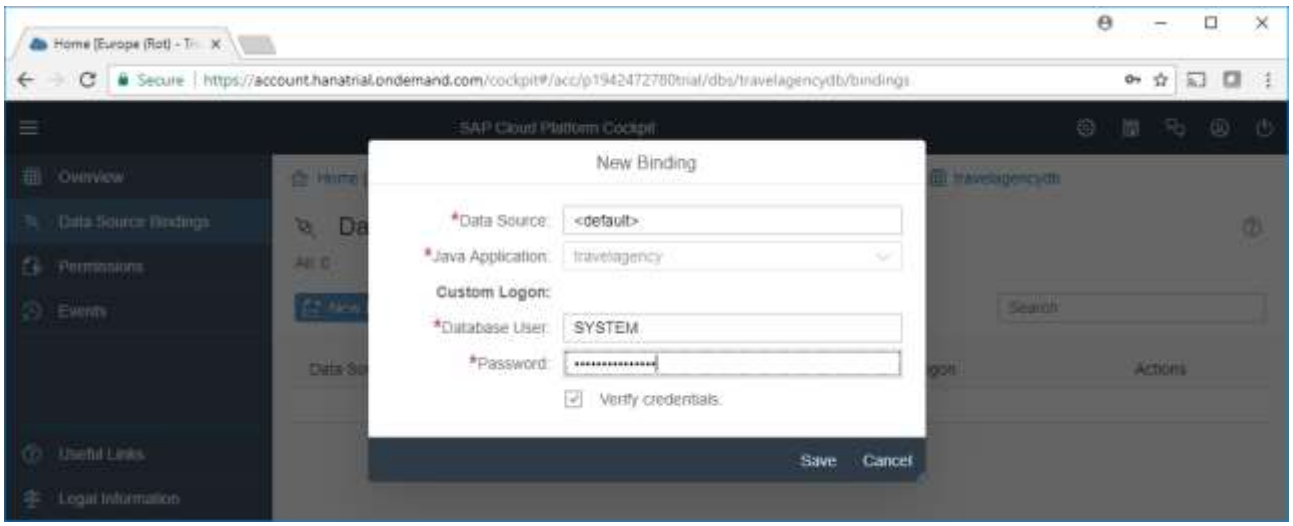
Antes de iniciarla, debemos establecer un “binding” entre esta aplicación y la base de datos, para que queden vinculadas. (Recordemos que ya no se necesitará túnel).

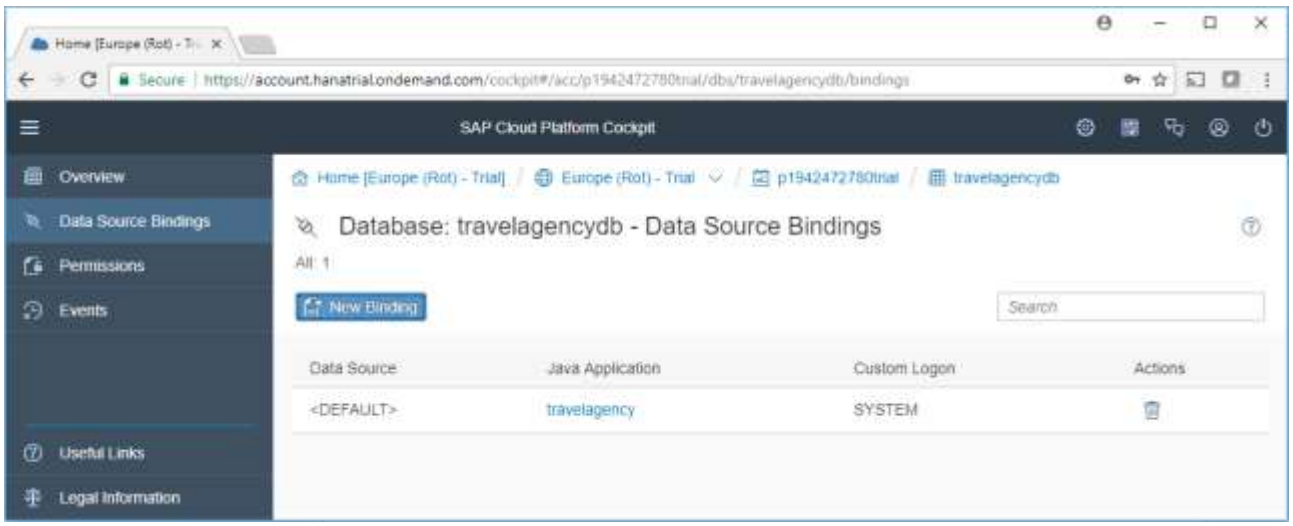
Vamos a Databases & Schemas, elegimos nuestra base de datos, que como podemos ver tiene 0 bindings…



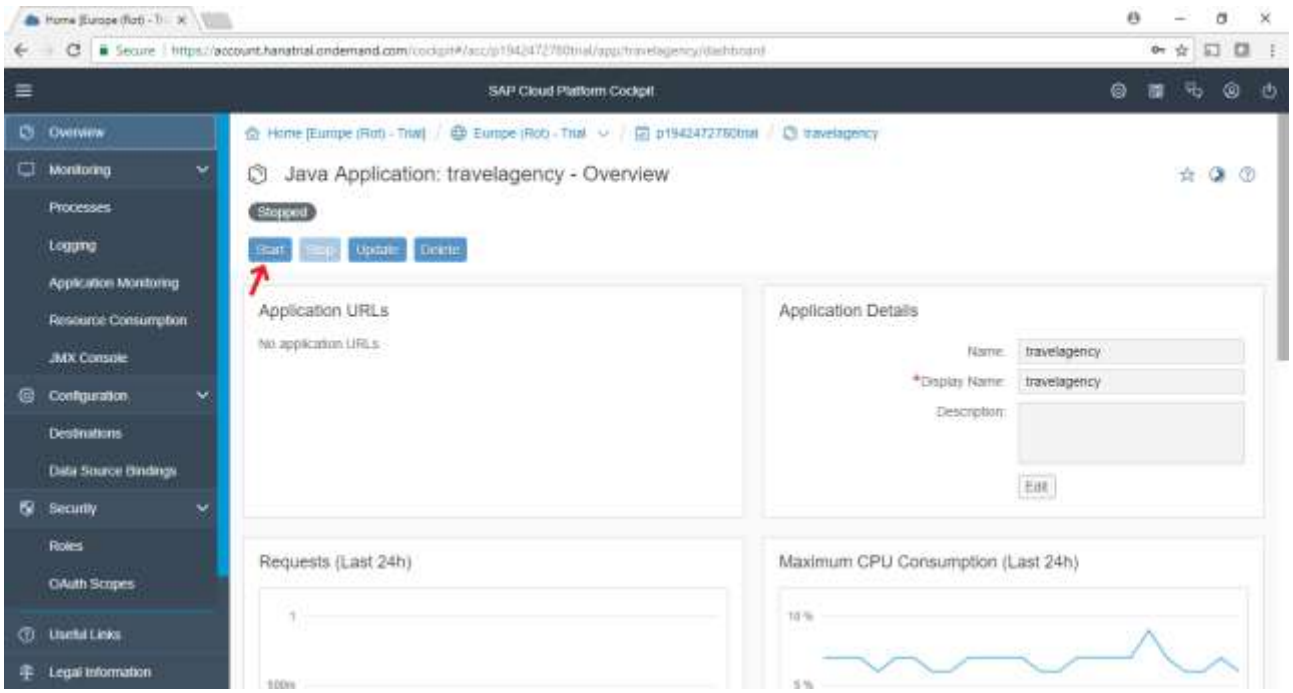
Y en Data Source Bindings agregamos uno para la app que acabamos de deployar:



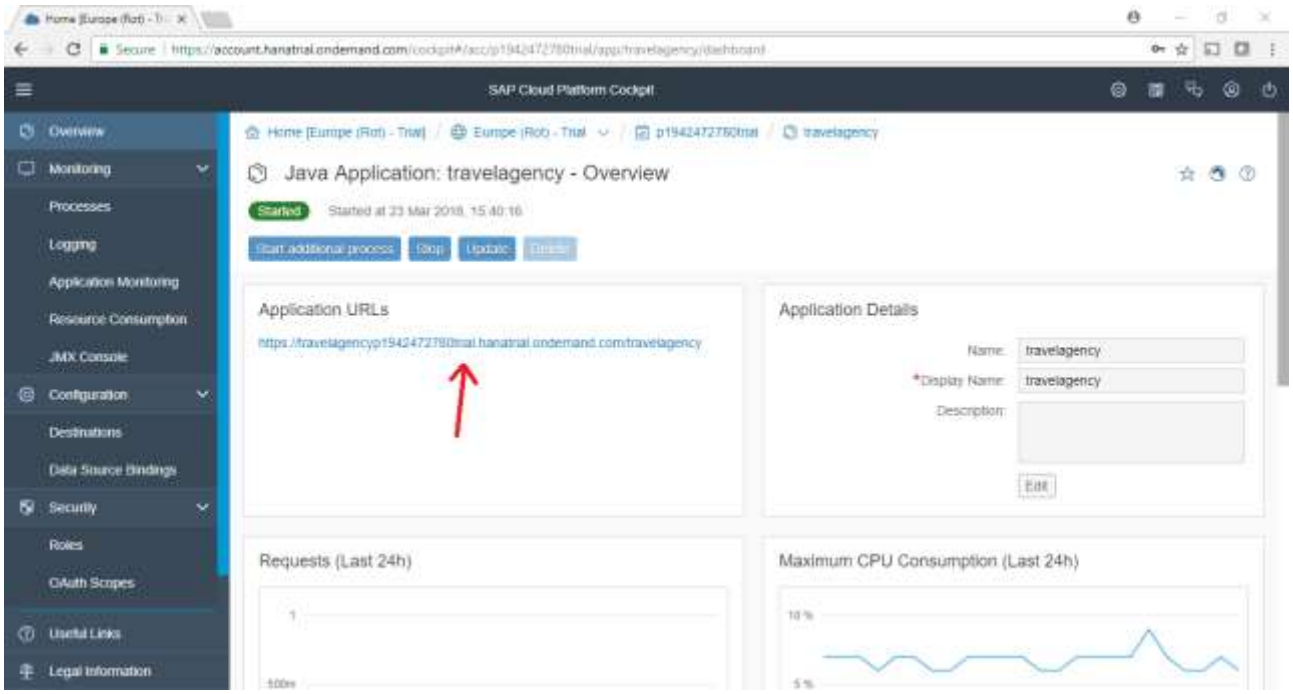




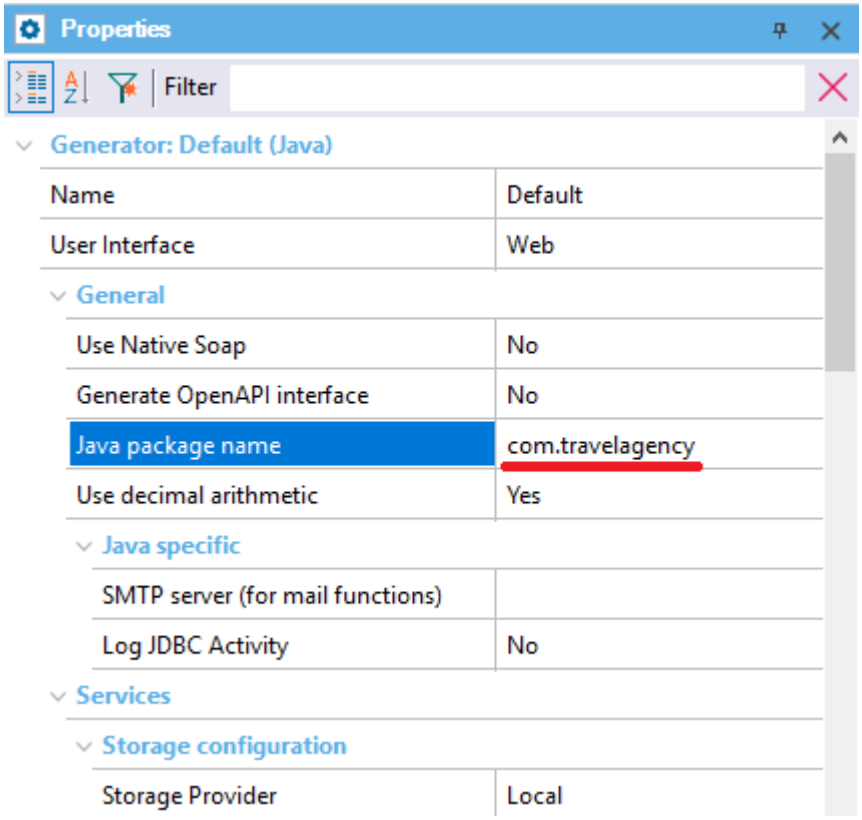
Volvemos a las Java Applications, seleccionamos la aplicación y presionamos Start:



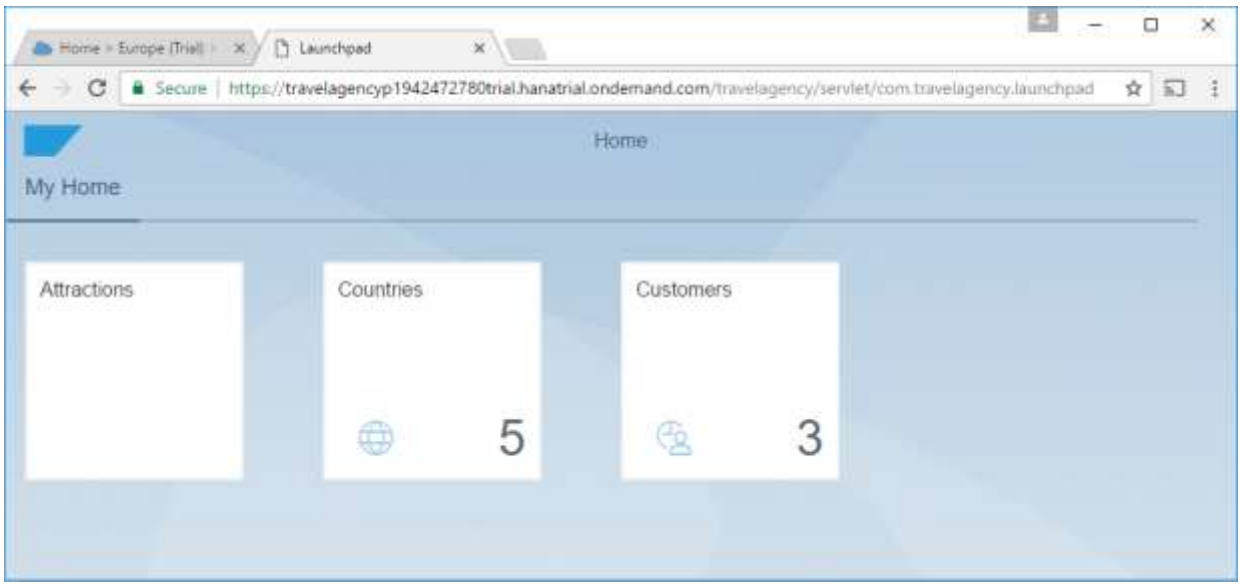
Cuando termina, aparece la URL base de la aplicación:



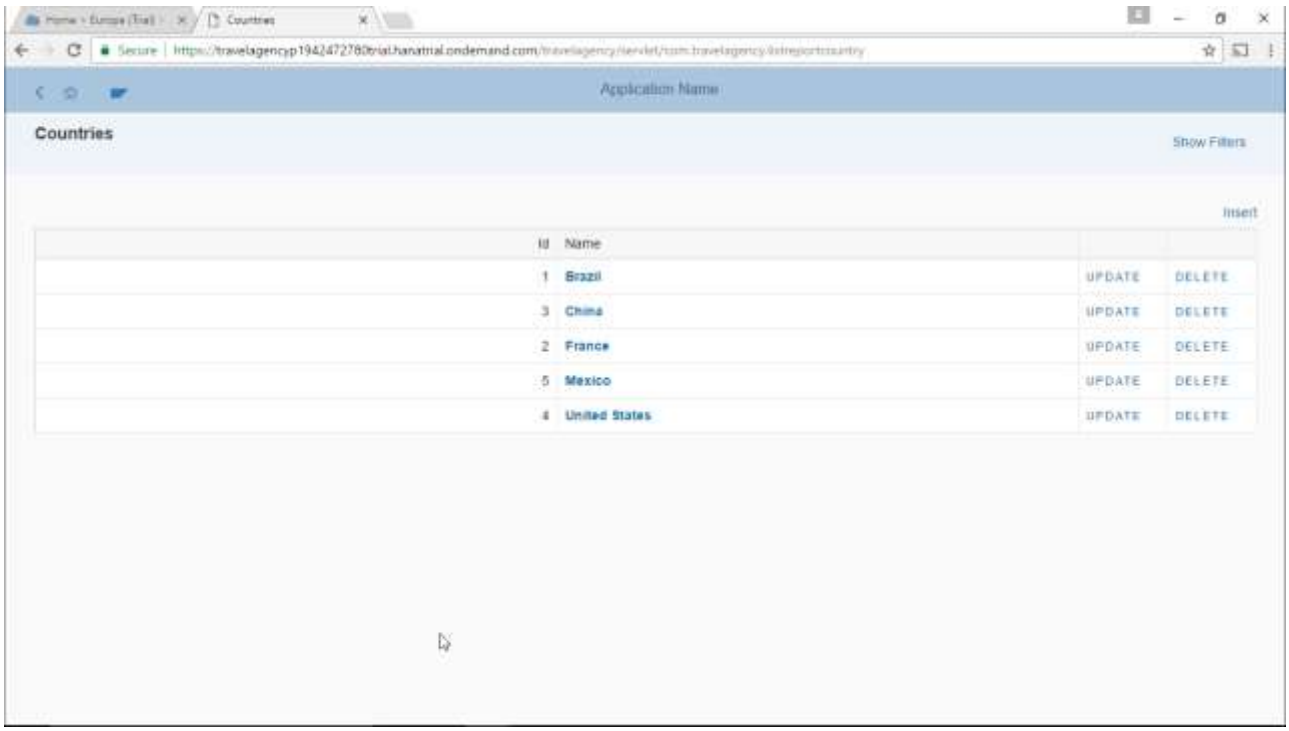
Tenemos que agregarle: barra servlet, barra… el java package name que encontramos entre las propiedades del generador Java:



punto... el nombre del módulo en el que se encuentra el objeto main, FioriBaseObjects, punto el nombre del objeto main, que en nuestro caso es “Fiorilaunchpad”:

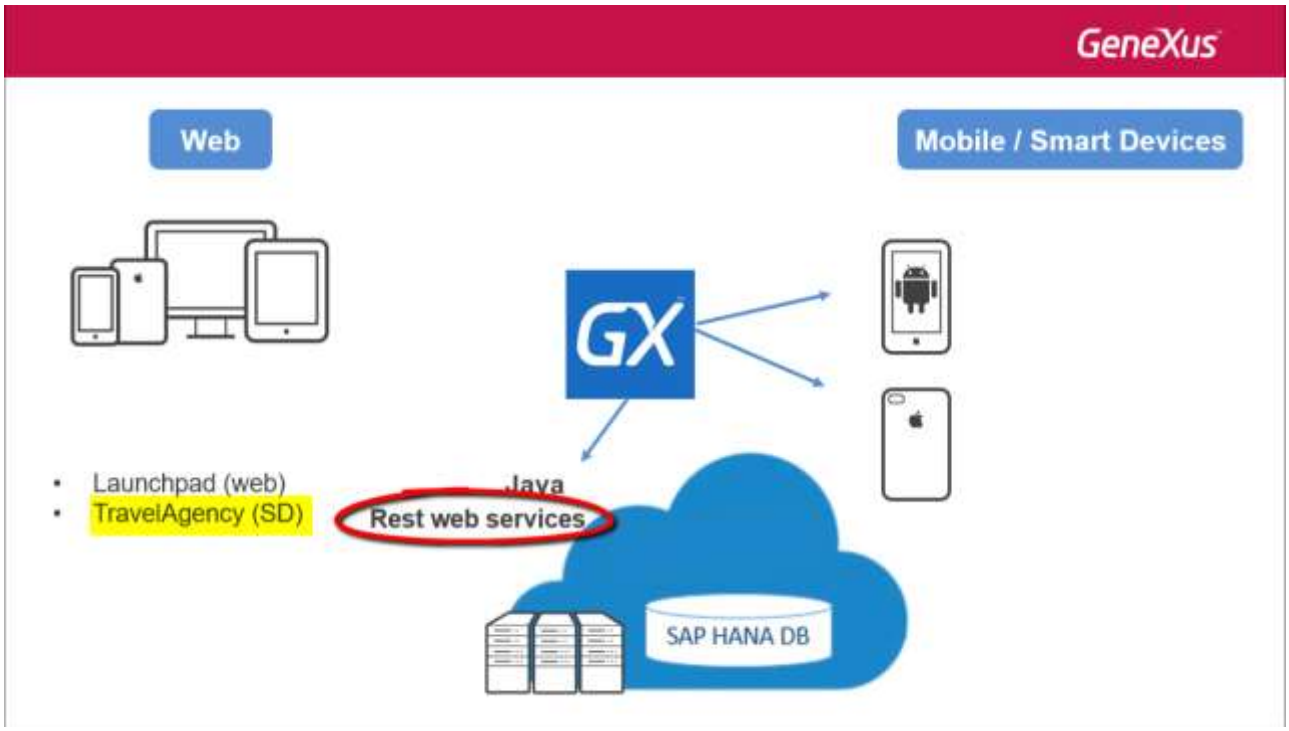


Ya tenemos la aplicación ejecutando en SAP Cloud Platform, accediendo a la misma base de datos con la que veníamos trabajando.

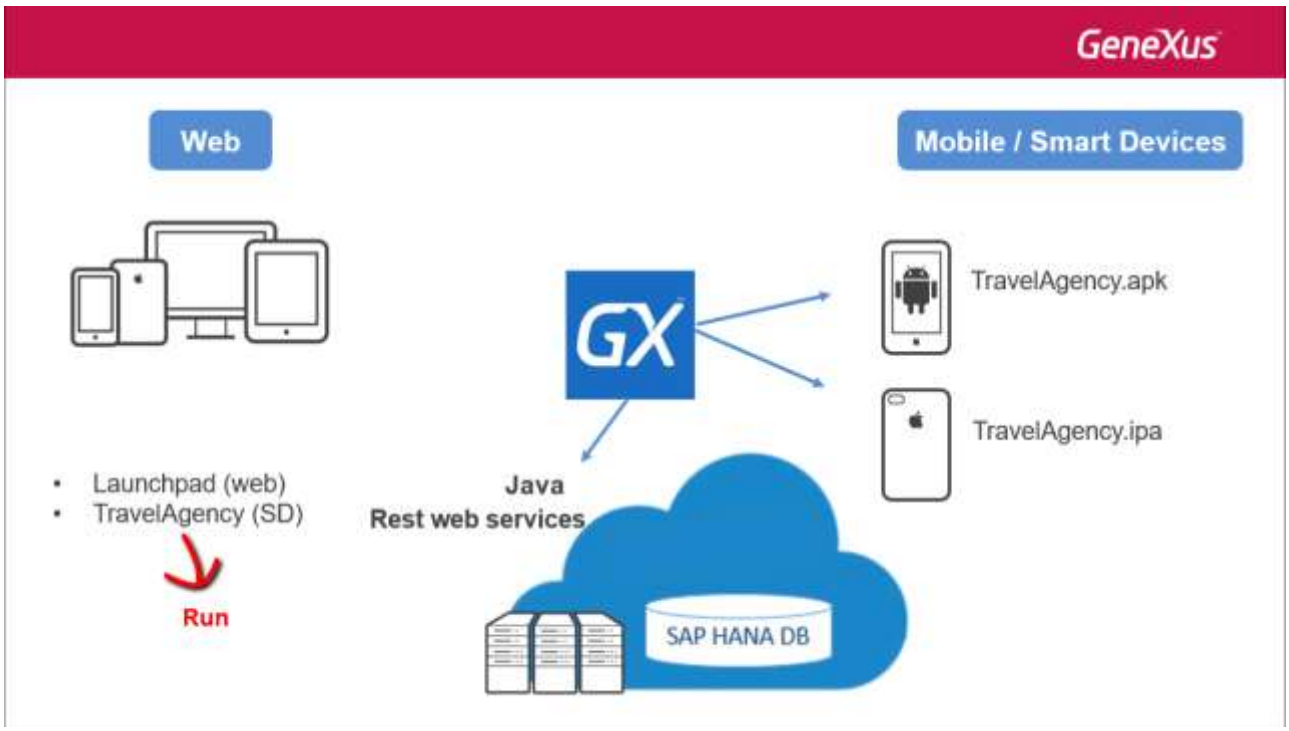


¿Qué pasa con la aplicación móvil?

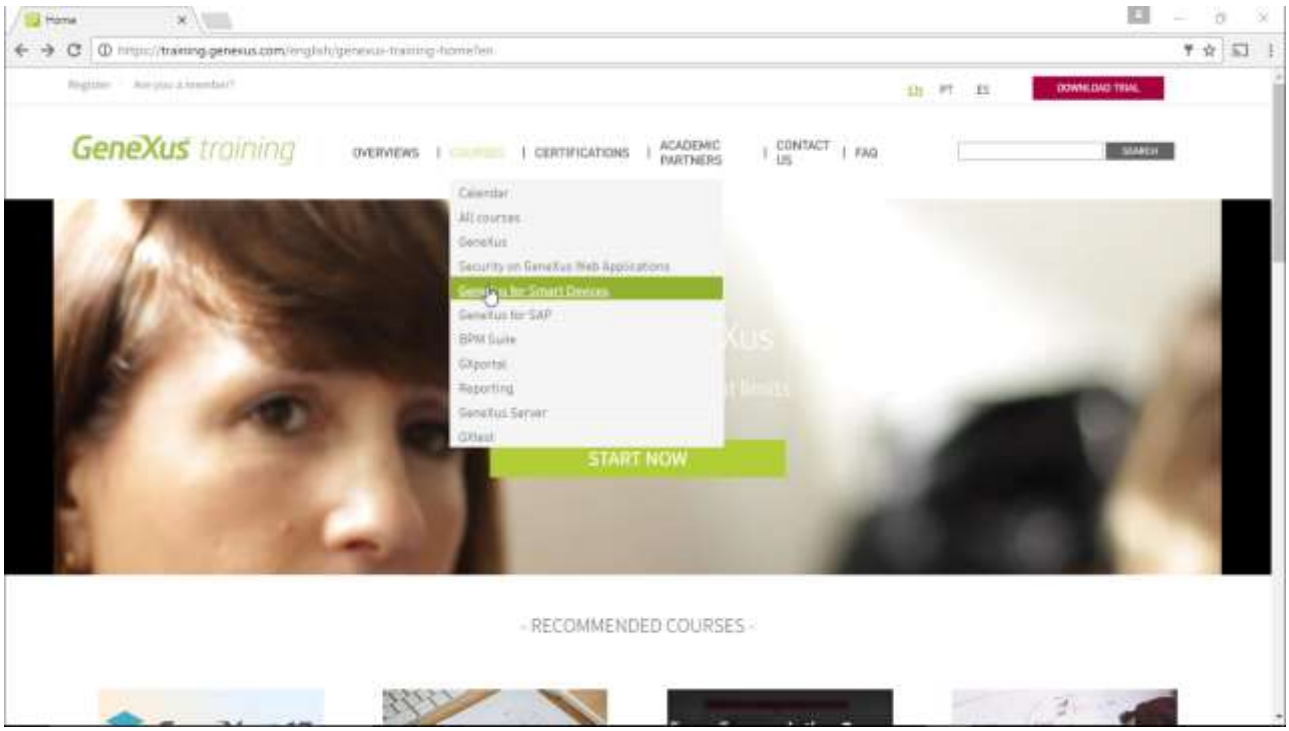
En el Deploy habíamos incluido al objeto Menu for Smart Devices. Pero esto lo hicimos únicamente para que, además de la aplicación web que utilizará el usuario, también se instalaran los servicios Rest que la aplicación móvil deberá utilizar toda vez que pida datos a la base de datos o actualice.



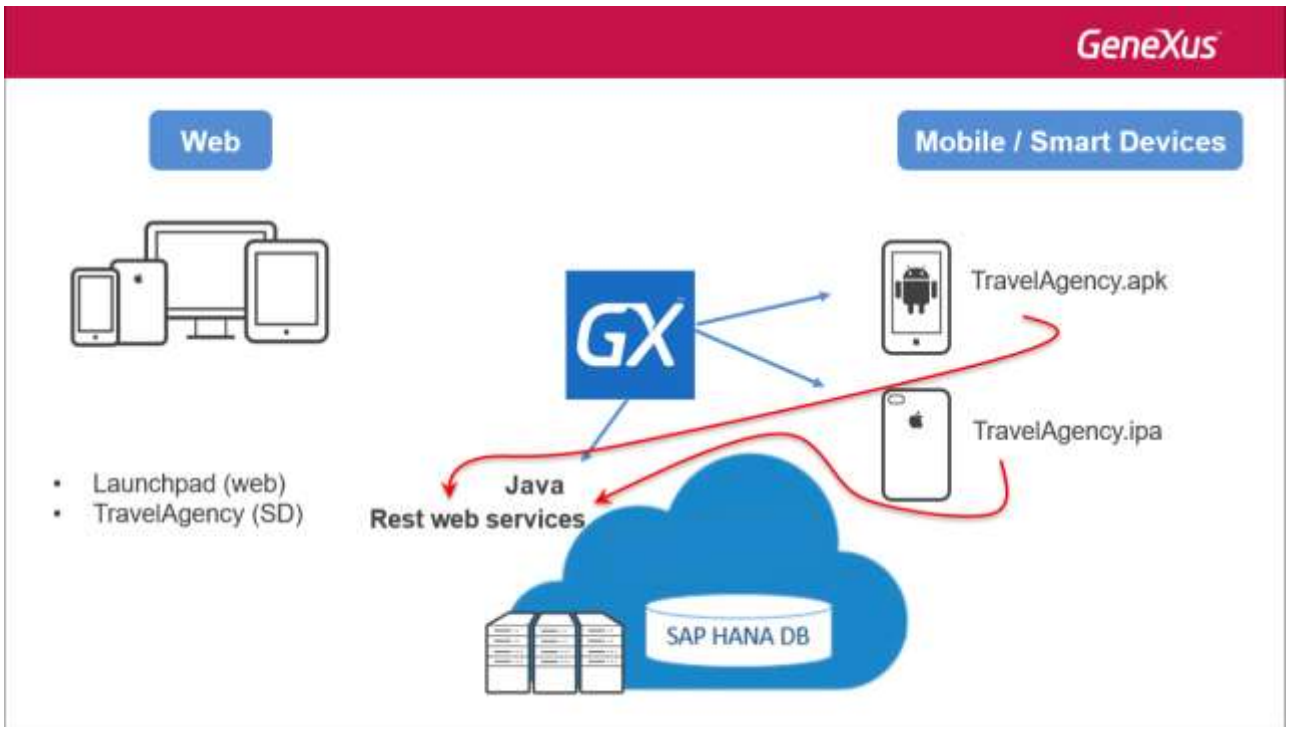
El compilado de ese menu que se obtiene automáticamente toda vez que hacemos Run (en Android un archivo de extensión apk, en iOS uno de extensión ipa)…



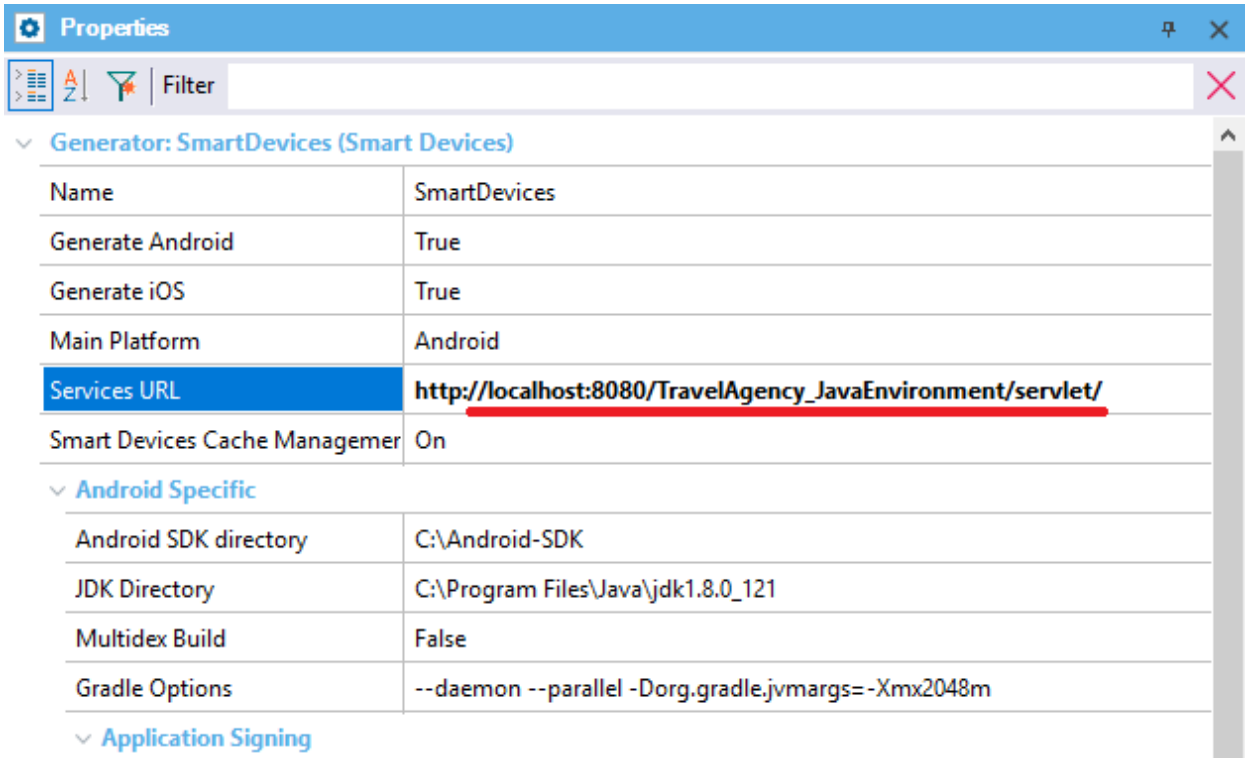
… luego deberá seguir su proceso para ser subido al store correspondiente (debe ser aprobado por Google o Apple), o, si no se desea disponibilizar a todo público, para ser distribuido en los dispositivos móviles de la compañía, por ejemplo. Para consultar sobre los distintos pasos a seguir para el deploy y la publicación en los stores, acceda a nuestro curso específico para Smart Devices.



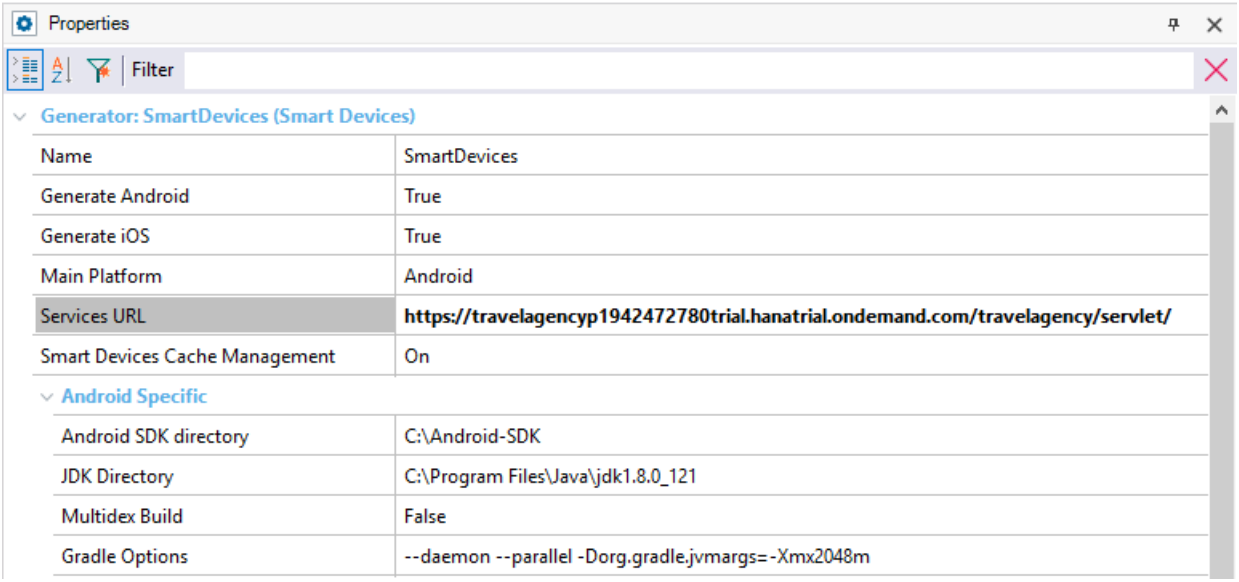
independientemente de ello, nos está faltando hacer que los programas de ese compilado que se instalará en los dispositivos apunten a los servicios rest en el servidor de SAP Cloud Platform.



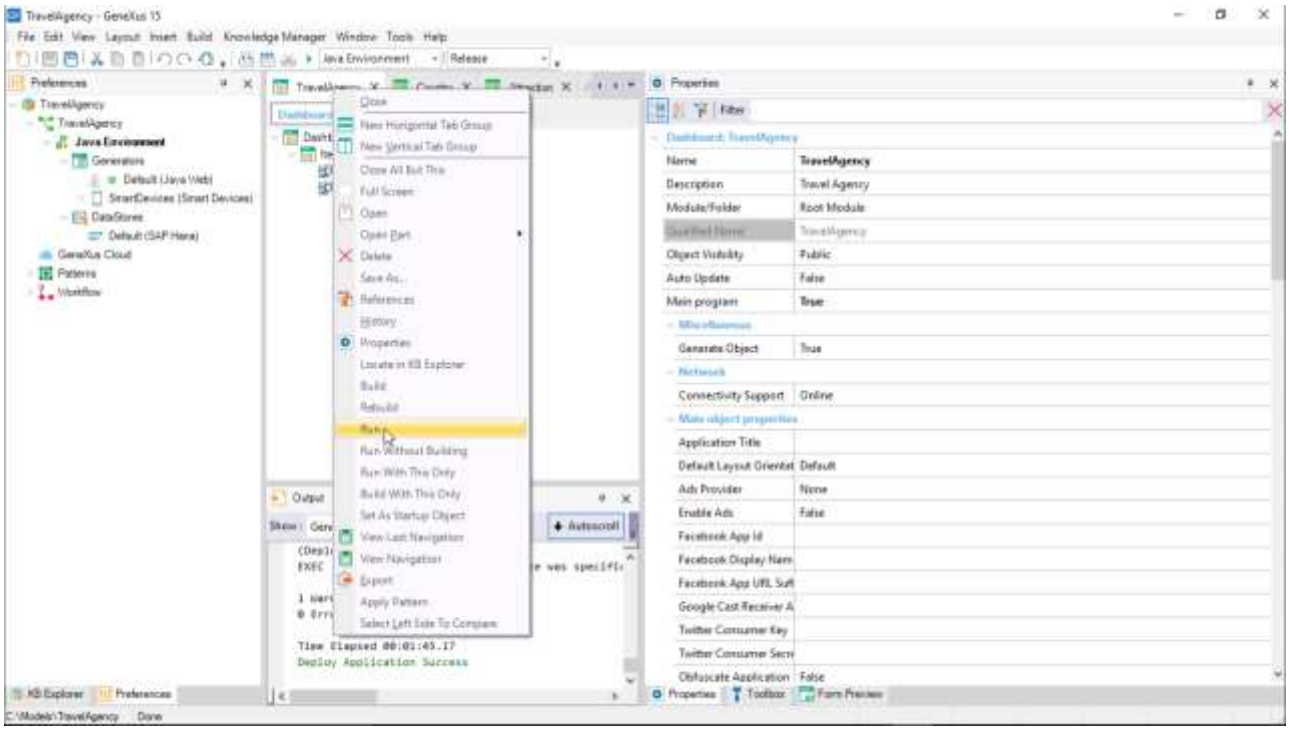
En nuestro modelo de prototipo, como el server era el tomcat local, la url de los servicios era:



Tenemos que modificarla para que sea la URL del web server en SAP Cloud Platform (url que obtuvimos al hacer el deploy):



Y entonces volver a compilar el objeto main para que vuelva a generar los programas, ahora apuntando a los servicios Rest en el servidor de SAP Cloud Platform.
(Haciendo Build si sólo queremos compilar, o Run si además queremos ejecutar en el emulador o en el dispositivo conectado a la computadora).

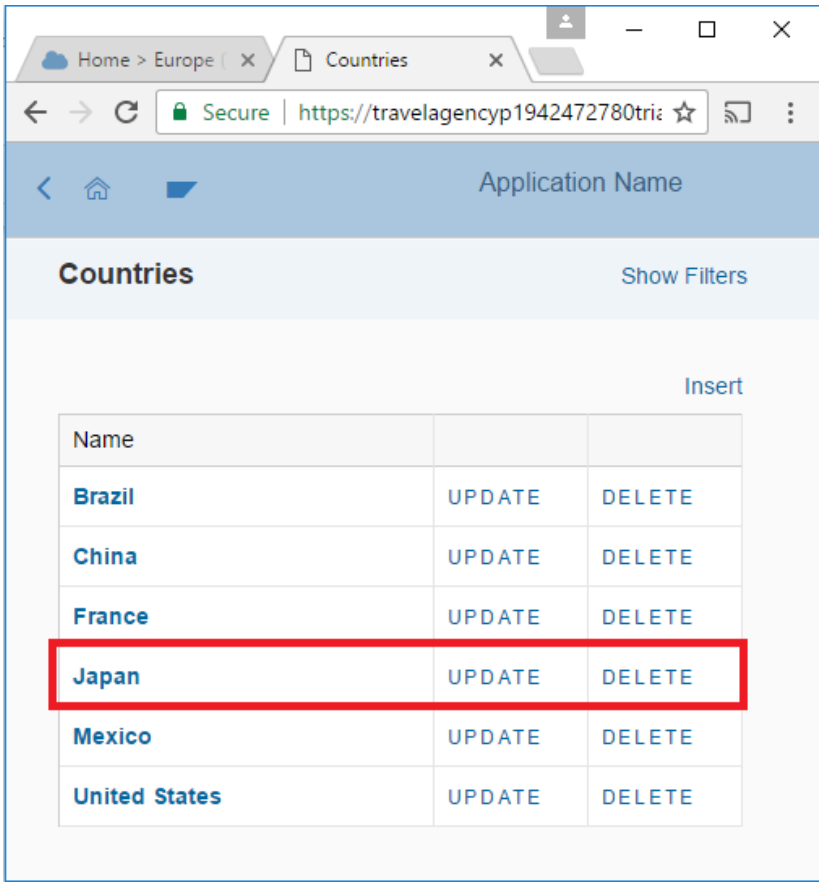


Observemos la lista de países.

Ahora insertemos un nuevo país a través de la aplicación en SAP Cloud Platform: Japón.

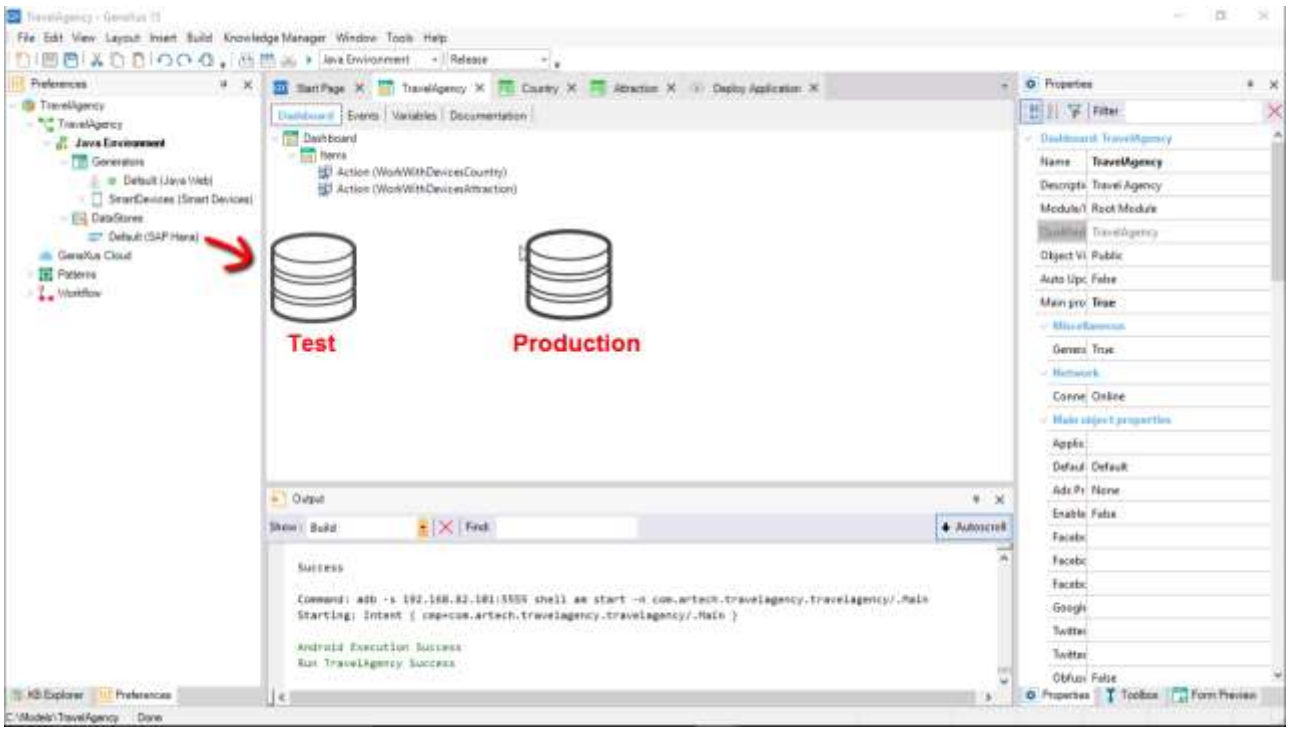
Y veamos la lista de países:

Y en la aplicación web:



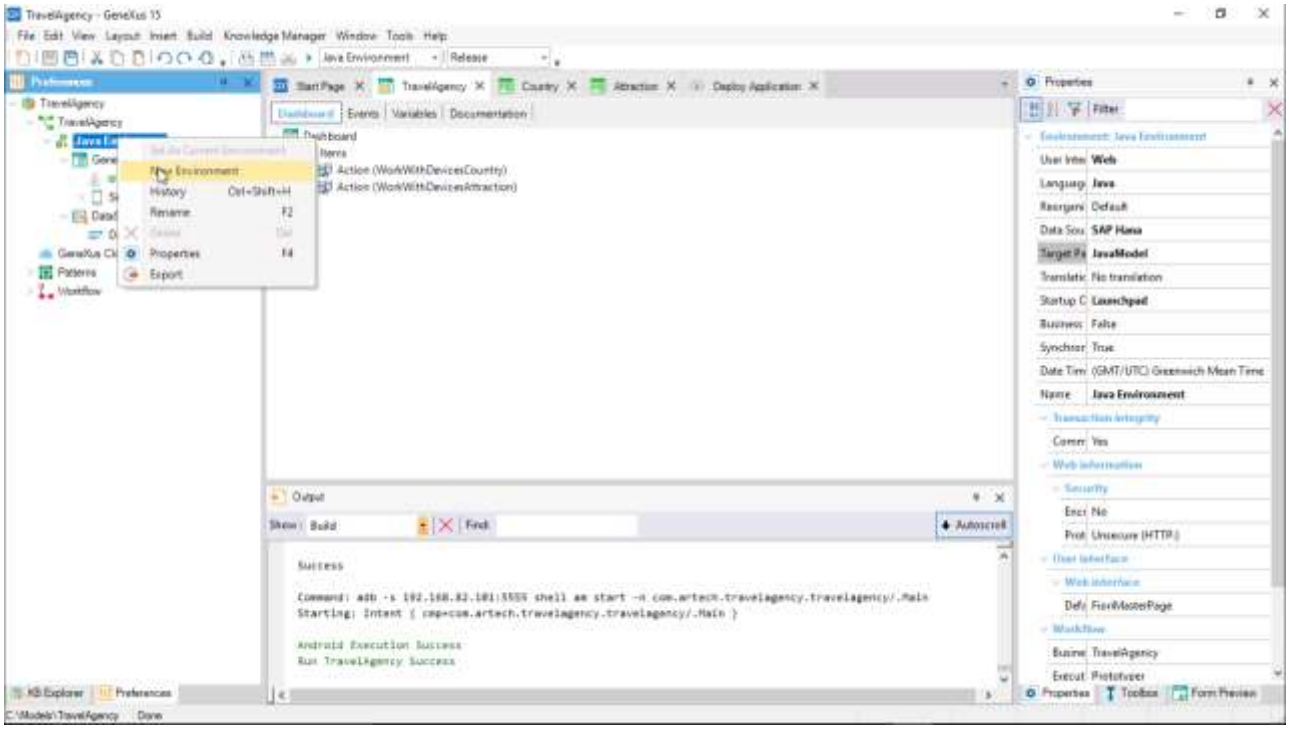
La pregunta que ahora se abre: ¿qué pasa cuando queremos poner en producción?

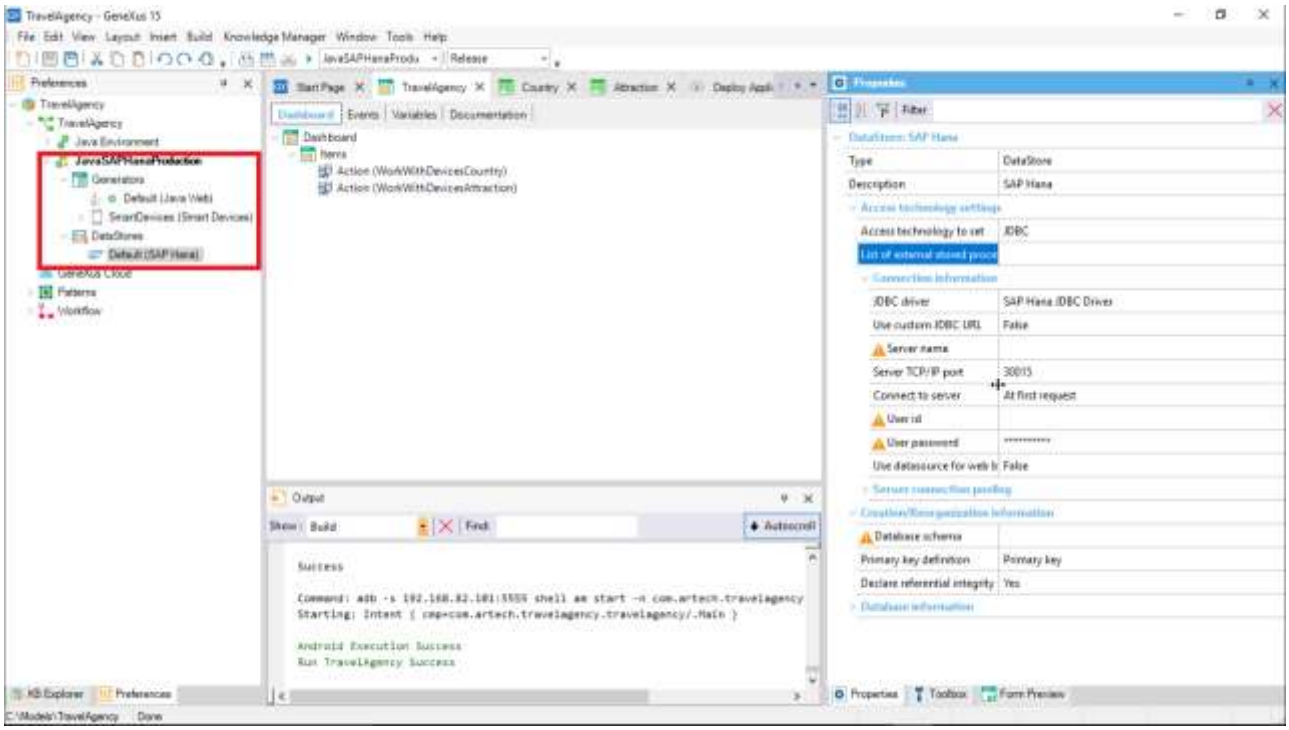
La base de datos de producción evidentemente no será la misma que la de test.



Entre otras cosas porque una vez puesta en producción la aplicación, seguiremos desarrollándola, tanto para realizar cambios en lo ya implementado como para incorporar nuevas funcionalidades. Y esa etapa de prototipación no debe realizarse en el ambiente de producción.

Por lo tanto, en lugar de modificar el environment que ya teníamos (en el que veníamos prototipando), a la hora de poner en producción sugerimos crear un nuevo environment:



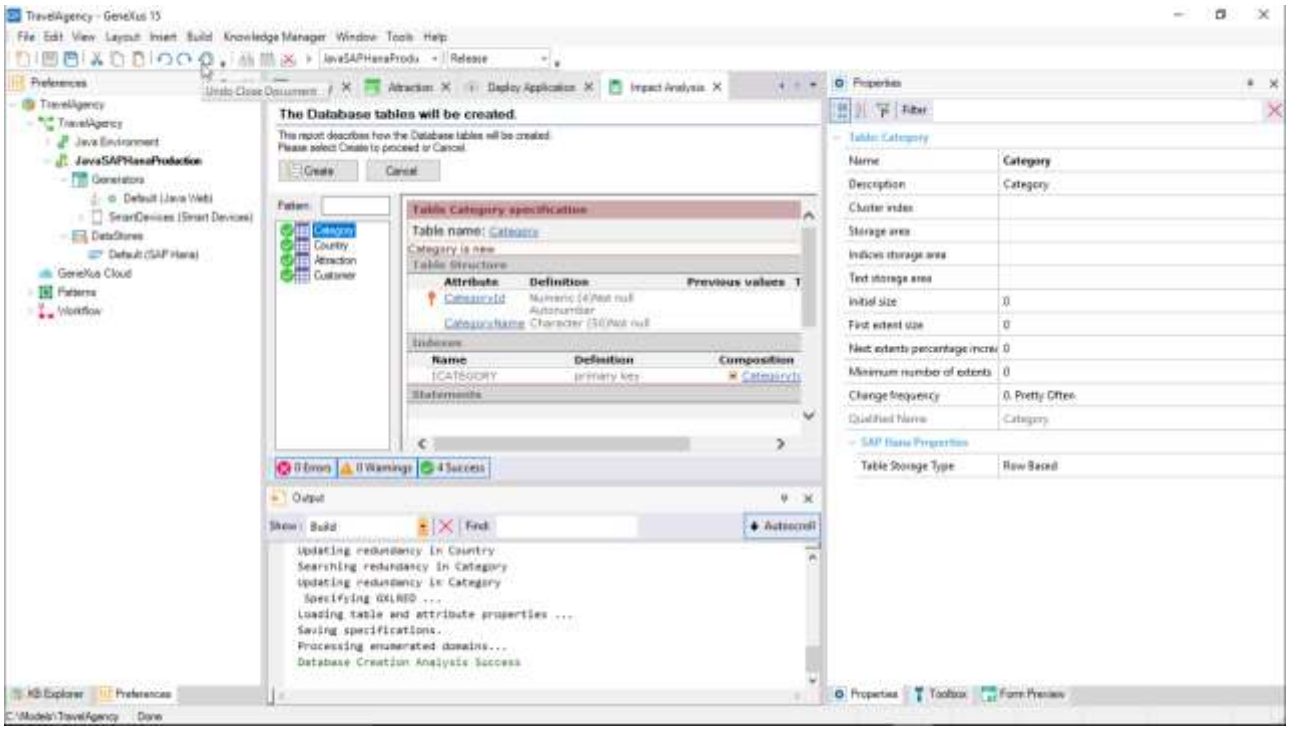


…crear la base de datos de producción en SAP Cloud Platform tal como lo hicimos la primera vez (cuando quisimos ejecutar en el environment creado por defecto con la KB). Como estamos trabajando con la trial para esta demo, que solo admite una base de datos por usuario, deberemos utilizar otro usuario, haciendo todo de manera idéntica a como ya lo hicimos. Es decir, crear la base de datos:

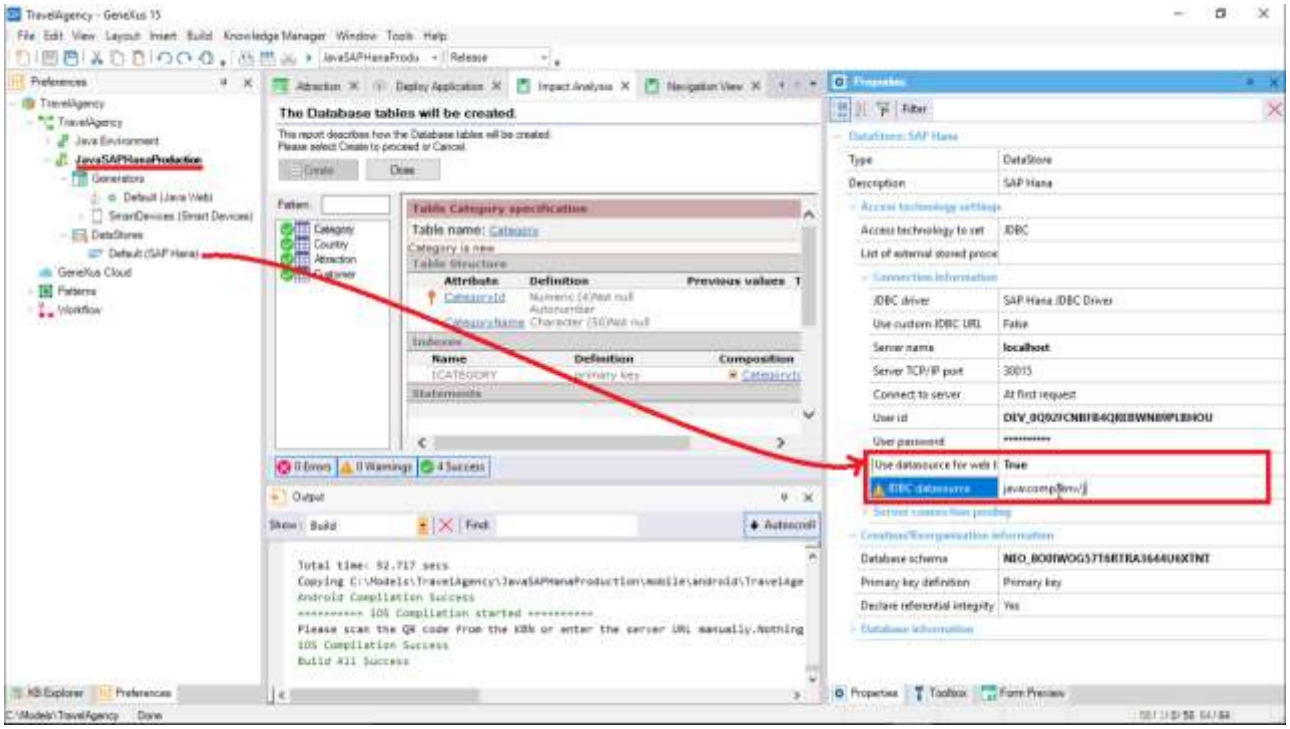
Y abrir el túnel con los datos nuevos… todo análogo a lo que hicimos aquella vez...

Y luego de todo eso, hacer un **Build All**:

…donde pedirá reorganizar la base de datos, para crear todas las tablas, que en este nueva base de datos aún no existen.



Una vez completado este paso, usted hará todo lo que vimos antes (cambiar la propiedad “Use data source for web based applications”, etcétera).



Con esto vimos lo fácil que es poner en producción una aplicación en SAP Cloud Platform.