

## Characteristics

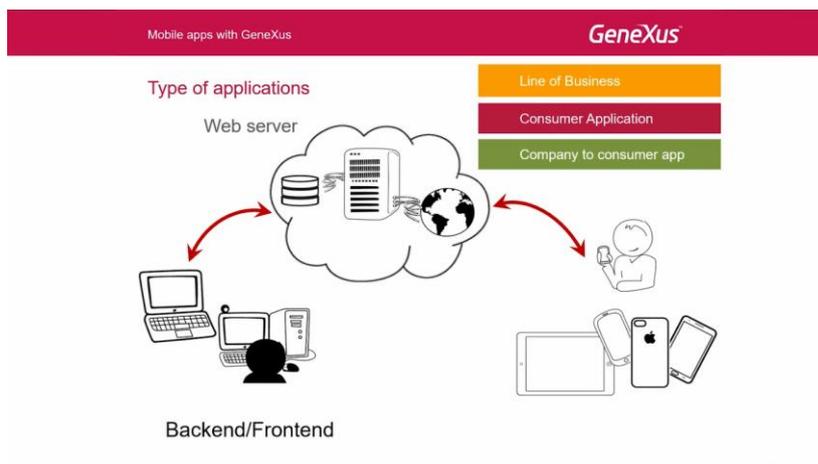
GeneXus

### Characteristics of mobile applications for Smart Devices

Vamos a ver las características principales de este tipo de aplicaciones.

Toda aplicación para Smart Devices que desarrollemos va a tener necesariamente una parte ejecutándose en un servidor web. No sólo en el caso de una aplicación **Line of Business** (es decir, una aplicación para Smart Devices que es sólo una parte de la aplicación empresarial), sino también en el de las **Consumer Applications**, es decir, aplicaciones hechas para ser subidas a los mercados de las distintas tecnologías, para poder ser descargadas luego por usuarios cualesquiera (tanto en forma gratuita como paga); y ahí tenemos ejemplos innumerables de ese tipo de aplicaciones.

Entre estas dos posibilidades se encuentran las **Company to consumer applications**, que son aplicaciones desarrolladas por la empresa para sus clientes, y para extender también su base de usuarios.

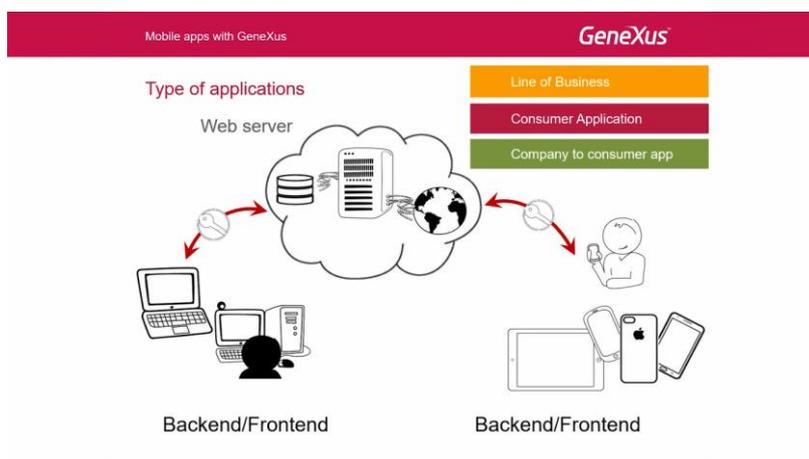


Nuestra aplicación EventDay puede considerarse un híbrido, puesto que tendrá una parte cuyo foco estará puesto en los asistentes al evento en cuestión (un frontend), y otra parte que será utilizada por los organizadores del evento, para poder modificar datos de la base de datos de forma móvil (es decir, el backend).

En definitiva, así como podemos querer publicar, crear un frontend web, para que los concurrentes al evento puedan, conectándose a internet con un navegador en cualquier computadora, acceder a la información de las conferencias y todo lo demás, también vamos a poder querer tener el frontend funcionando en un dispositivo inteligente.

Y por otro lado, así como también tenemos el backend web, podríamos querer tener el backend funcionando en el dispositivo. Un aspecto importante en el caso del backend, que no difiere funcionalmente en nada del frontend, tiene que ver con la seguridad. Evidentemente no cualquier usuario va a poder estar autorizado a realizar modificaciones en la base de datos de la aplicación, base de datos que va a estar centralizada en el servidor web.

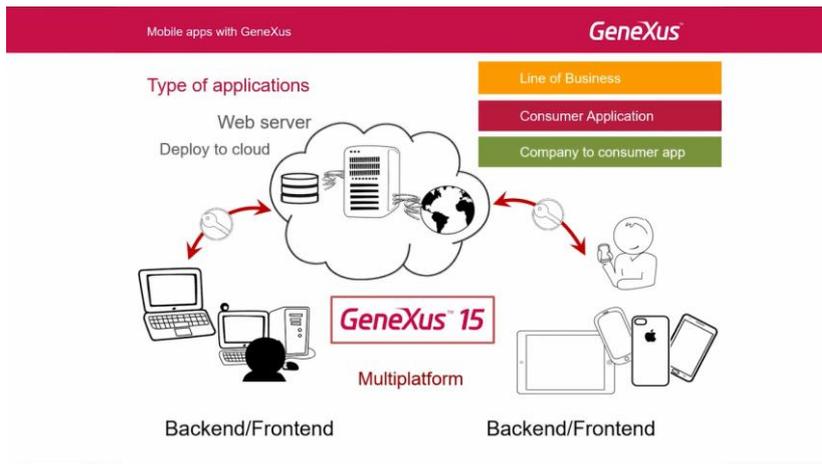
Entonces para ello vamos a tener que incorporar también en las aplicaciones para Smart Devices módulos de seguridad que puedan entonces restringir los usuarios autorizados, e incluso restringir que acciones pueden ejecutar sobre los distintos módulos de la aplicación. De este modo entonces, teniendo un backend en los dispositivos móviles los usuarios autorizados van a poder modificar la información de una conferencia un segundo antes de que esta da inicio.



Bien. Así como GeneXus nos permitía desarrollar aplicaciones en .NET o Java para web, también nos va a permitir entonces desarrollar aplicaciones para iPhones o iPads en el lenguaje Swift o para Android en el lenguaje Java, siendo otra vez multiplataforma nuestra aplicación.

Esto lo veíamos cuando recién mostrábamos las aplicaciones, veíamos que teníamos una aplicación que se comportaba de una manera que era esencialmente muy parecida, en iPad, a la aplicación para Android; pero también programábamos algunas diferencias en un caso o en el otro.

Bueno esto también lo vamos a respetar en GeneXus: vamos a desarrollar esencialmente la misma aplicación, pero vamos a poder programar o diseñar particularidades de acuerdo a la plataforma en la que se vaya a ejecutar. Pero la aplicación va a ser esencialmente la misma. Esto se va a realizar con la propiedad Deploy to cloud.

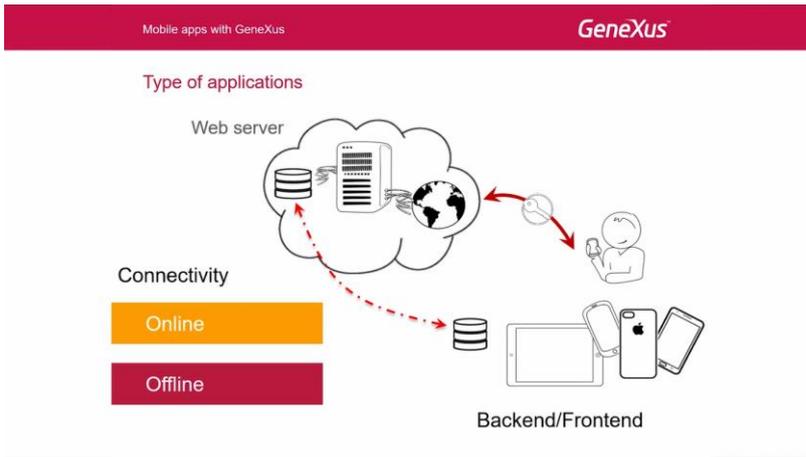


Un escenario importante para los dispositivos inteligentes es permitir que la aplicación, o parte de ella, se siga ejecutando cuando se encuentra desconectada de internet. En el caso de nuestra aplicación, vamos a querer que el usuario pueda seguir viendo toda la agenda de conferencias, y toda la información relacionada, incluso cuando pierde la conexión. Luego, cuando ésta se reestablece, automáticamente la aplicación actualizará sus datos (que estarán en una base de datos local en el dispositivo), sincronizándose con el server (Receive). Como el usuario podrá haber marcado algunas conferencias como favoritas, también enviará esa información al server al sincronizarse (es decir en el Send de la información de sincronización), actualizando la base de datos central.

Sin embargo, habrá tareas que requerirán necesariamente el acceso al servidor web, ya sea por su sensibilidad (tareas que deben ser validadas en la base de datos centralizada), como por lo rápido que cambian los datos. Estas tareas deberán ejecutarse online.

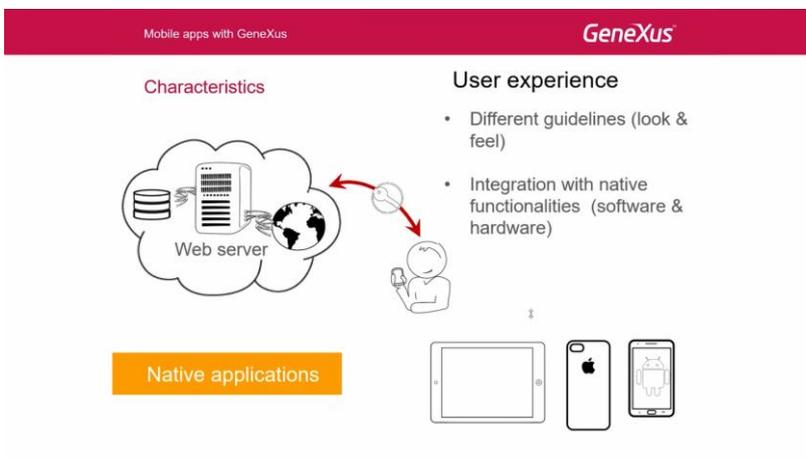
En nuestro caso, por ejemplo, el login deberá ser con conexión, y el panel que muestra los tweets es deseable que también lo sea porque los tweets son algo que cambia vertiginosamente, todo el tiempo.

Por tanto, podremos elegir qué objetos de la aplicación pueden ejecutarse offline y cuáles no. Sobre las aplicaciones offline, vamos a volver sobre el final del curso. Lo importante a tener en cuenta entonces es que las aplicaciones online no van a tener una base de datos local y van a estar siempre conectando al servidor web para obtener su información. Con esa idea es con la que nos tenemos que quedar; ahora en un rato vamos a ver la arquitectura de este tipo de aplicaciones, las online.



Como características de las aplicaciones para Smart Devices sabemos que la experiencia de los usuarios es fundamental. Cada plataforma tiene diferentes guidelines respecto al look & feel de la aplicación, es decir cómo debe verse la aplicación, cómo deben verse las acciones, y demás. Pero además, también vamos a querer que las apps en Smart Devices se integren con las funcionalidades nativas del dispositivo, tanto de software como de hardware. Por ejemplo que se integren con la cámara del dispositivo, que se integren con el programa para hacer llamadas telefónicas, con el GPS, etc.

Es decir, los usuarios van a demandar que la experiencia que tengan con las apps en su dispositivo sean uniformes con el resto de la experiencia que tienen con las demás aplicaciones nativas del dispositivo. Por lo tanto las aplicaciones para teléfonos inteligentes y tabletas necesariamente tienen que ser aplicaciones nativas.



Por ejemplo, en el caso de un iPhone, estamos viendo a la izquierda la pantalla en la app de contactos nativa del dispositivo, que como vemos tiene una action bar arriba, para mostrar las acciones que se pueden hacer sobre esta pantalla, y también tiene una opción de search. Y a la derecha por ejemplo estamos viendo la aplicación que desarrollamos con GeneXus para esta aplicación Event Day que decíamos, y vemos que lucen de manera muy parecida.

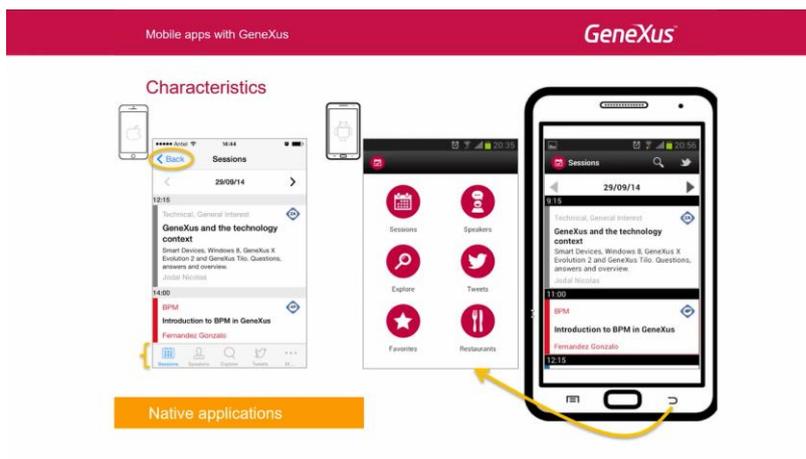
También vemos que arriba tienen entonces el nombre de la pantalla que estamos visualizando, y la barra de search; vemos que en la parte central de la pantalla es donde se muestra la

información, en un caso la lista de contactos, en otro la lista de Speakers, y vemos que tienen abajo el menú como una serie de tabs, con determinados colores: cuando está seleccionada la pantalla o el ítem del menú aparece coloreado en celeste, cuando no, aparece en gris, en estos casos, esto se trataría de un iPhone 7.



Y si nos desplazamos a otra tecnología, por ejemplo a la izquierda estamos viendo la pantalla de Sessions, de conferencias, de nuestra aplicación EventDay para un iPhone, y a la derecha estamos viendo la misma aplicación pero para un teléfono Android. También vemos que tiene diferencias, por ejemplo vemos que en el caso del iPhone, el menú está siempre visible como tabs en la parte inferior de la pantalla; en cambio, en un dispositivo Android, vamos a poder acceder a ese menú inicial, a esas opciones iniciales, con un botón, pero no lo vamos a tener entonces visible en la pantalla con forma de tabs, va a ser una pantalla independiente.

Por otro lado, tenemos en un teléfono iPhone la opción de Back dentro de la propia aplicación, y eso en un teléfono Android es un botón del propio teléfono, no está incorporado en la aplicación.



Si ahora miramos el detalle de una conferencia, vemos que en el iPhone tenemos ese ícono, que es el ícono universal de iOS para mostrar que apretándolo vamos a poder acceder a una lista de acciones para ejecutar. En este caso por ejemplo sería poder ingresar esta conferencia

en el calendario, un calendario que es propio del dispositivo, entonces presionando Schedule se abrirá la app de Calendario para agendarla.

Y por otro lado, en Android, va a depender un poco de la versión del sistema operativo con el que estemos trabajando, pero, van a aparecer esos tres puntitos por ejemplo, para mostrar allí arriba esas mismas acciones; y también eligiendo Schedule se va a abrir ahora el programa de calendario del dispositivo Android, que es diferente del de un iPhone.

Por otro lado, cómo decíamos, en un iPhone tenemos la posibilidad de hacer Back, de volver al que llamó a este panel, incorporando esa opción dentro de la propia aplicación con un botón ahí arriba. En cambio, en Android, tenemos el botón de Back.

