

# 顧客向けに重点を置いた Web 画面

概要

*GeneXus*<sup>™</sup>

シンクライアント  
+  
サーバー上の  
ビジネスロジック

バックオフィス重視の  
Web アプリケーション

スマート  
クライアント  
+  
サーバー上の  
サービス

UX 重視の  
Web アプリケーション

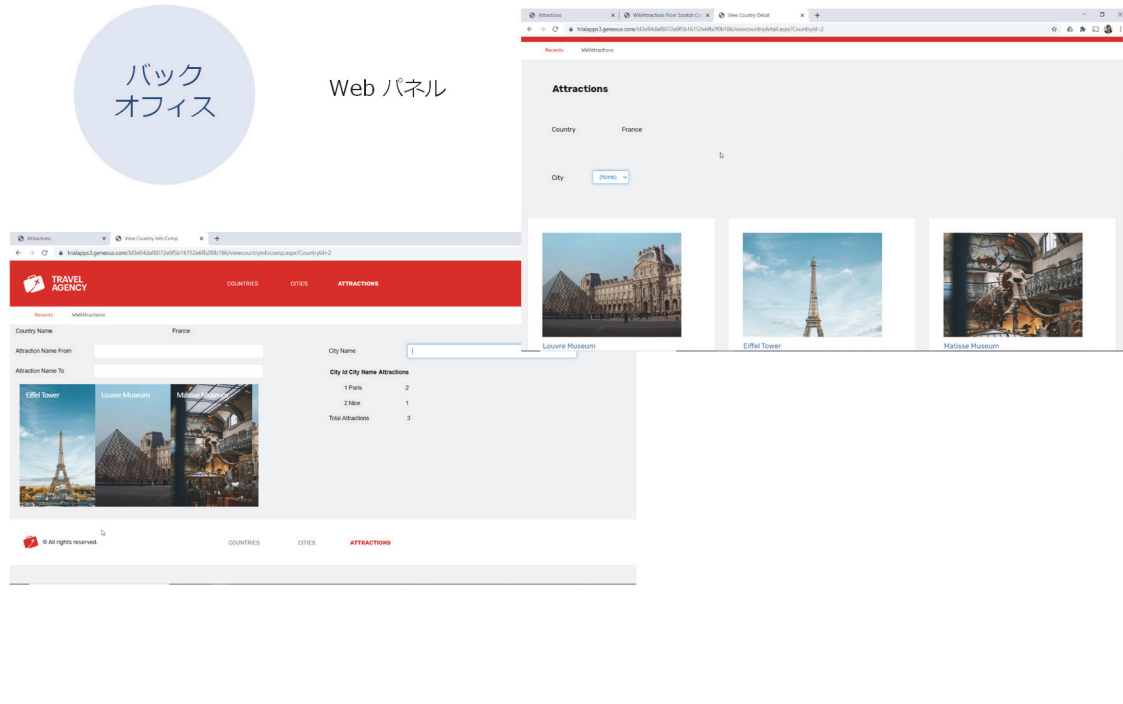
ネイティブ モバイル  
アプリケーション

これまで、アプリケーションのアーキテクチャ条件について、特にプログラミング方法について説明しました。

この章では、スマートクライアントで実行される Web アプリケーション、つまり、可能な限り優れたユーザーエクスペリエンスの提供が可能な顧客向けアプリケーションに焦点を当てます。

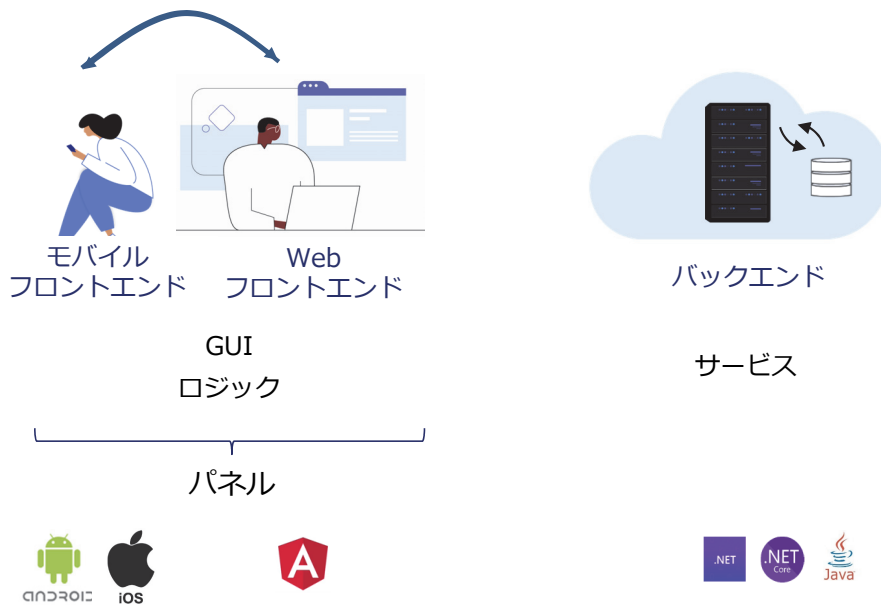
バック  
オフィス

Web パネル



これまで、旅行代理店のバックオフィス Web アプリケーションの開発に Web Panel オブジェクトを使用する方法を見てきました。

ここではユーザーエクスペリエンスを重視したプログラミングを目的としているので、アプリケーションのクライアントサイドのロジックに集中できる、これまでに扱ってこなかった新しい種類のオブジェクトを扱うことになります。



ユーザーが (Web アプリケーションとネイティブアプリケーションの両方から) 操作するすべての画面に、クライアントに表示される視覚的な部分とそれに関連するロジックが含まれています。ここで取り上げるアプリケーション、つまり可能な限り優れたユーザーエクスペリエンスを備えた顧客向けアプリケーションでは、このロジックはクライアント側で実行され、サーバーは、データの取得と管理に関連する機能をクライアントに提供する役割を担います。

Panel オブジェクトと呼ばれる GeneXus オブジェクトを使用すると、クライアント側で実行されるコード、およびバックエンドサービスの呼び出しを通じてサーバー側で実行されるコードを追加できます。このオブジェクトにより、Web アプリケーションやネイティブ モバイル アプリケーションなど、顧客向けアプリケーションの開発が可能になります。

これらの複雑なアプリケーションをクライアント側 (フロントエンド) に配置するためには、Web クライアントの生成には Angular を使用し、ネイティブ アプリケーション クライアントの生成には Android または iOS を使用します。GeneXus では、サーバー部分 (バックエンド) は .NET、.NET Core、または Java で生成されます。

Panel オブジェクトをそのロジックから順番に学習し、その後で画面のデザインに焦点を当てます。このオブジェクトの興味深い点は、プログラミング方法が両方のプラットフォームで有効であることから、Web アプリケーション用にデザインした画面をネイティブアプリケーションで使用したり、その逆を行ったりできることです。

ここでは、特に Web 画面の開発に Panel オブジェクトを利用するケースの簡単な紹介にとどめます。Angular を使用した Web アプリケーションの開発やネイティブ モバイル アプリケーションの開発など、ほかのコースでは、このオブジェクトの使用方法をより詳しく学習できます。



動画	<a href="https://www.genexus.com/community-and-support-jp/training?ja">https://www.genexus.com/community-and-support-jp/training?ja</a>
ドキュメント	<a href="http://wiki.genexus.jp/">http://wiki.genexus.jp/</a>
認定資格	<a href="https://training.genexus.com/certifications">training.genexus.com/certifications</a>