

# デジタルアプリケーションの現状

## アーキテクチャと分類

*GeneXus*<sup>TM</sup>

## 概要



現在、ソフトウェアが世界中に普及し、スマートウォッチからスマートフォン、テレビ、クラウド サーバー ファーム、自動運転車まで、多様なプラットフォームでデジタルアプリケーションが実行されています。

コンピューターシステムの急増や、急速な進化を目のあたりにして、私たちはいったん立ち止まり、最近のアプリケーションとそのアーキテクチャを理解する必要があります。

これから、それらの分類と、ビルドや通信の方法について見ていきます。

## プラットフォームとアプリケーションのパーツ



最初のポイントとして、アプリケーションは多様なプラットフォームで実行されるだけでなく、ほとんどの場合、アプリケーションの一部が1つのプラットフォームで実行され、残りの部分が別のプラットフォームで実行されます。

このことが多くの課題をもたらします。たとえば、さまざまなオペレーティングシステムに関する知識、そのハードウェアおよびソフトウェアの要件に関する知識、また、1つのアプリケーションのパーツ間や異なるアプリケーション間での通信に対する理解が必要となります。

ここでは Web アプリケーションとモバイルアプリケーションに対象を絞って説明していきます。

## デジタルアプリケーションのパーツ

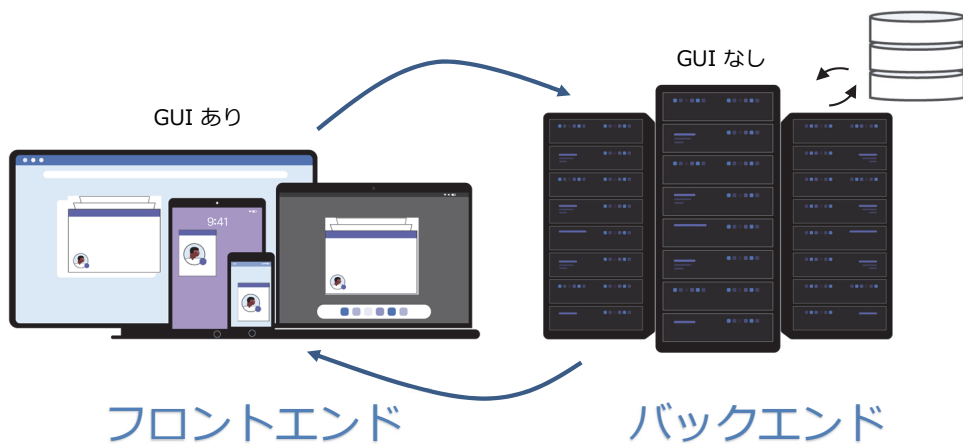


Web アプリケーションとネイティブ モバイル アプリケーションのいずれにも、ユーザーが操作するプラットフォーム（「クライアント」デバイス）で実行されるパーツと、アプリケーションのサービス専用のコンピューター（「サーバー」）で実行されるパーツがあります。

Web システムの場合、クライアントはインターネットブラウザであり、デスクトップコンピューター、ラップトップ、タブレット、スマートフォンなどで実行されます。サーバーは Web アプリケーションがホストされている場所であり、通常は、多くのユーザーから同時に届くリクエストを処理できる大規模なコンピューターです。

ネイティブアプリケーションの場合も、モバイルデバイス（クライアント）でローカルに実行されるソフトウェアと、サーバーで実行される別のソフトウェアがあります。デジタルアプリケーションのうち、クライアントデバイスで実行されるパーツは「フロントエンド」と呼ばれ、サーバーで実行されるパーツは「バックエンド」と呼ばれます。すべてのアプリケーションが、連携するこの 2 つのパーツから構成されます。

## デジタルアプリケーションのパーツ (続き)



アプリケーションのバックエンドは、クライアントアプリケーションにサービスを提供するプログラムで構成されています。たとえば、クライアントからリクエストされたデータを返すためにデータベースに接続します。

バックエンドは、ユーザーが操作できる画面を持たず、クライアントデバイス側のアプリケーション経由で指示を間接的に受け取ります。その後、必要なサービスを実行し、クライアントデバイス、厳密にはフロントエンドに応答を返します。

これに対して、フロントエンドはユーザーが使用するソフトウェアであるため、ユーザーがアプリケーションを操作できる画面があり、データの問い合わせ、作成、変更、削除などが可能です。こうした操作はバックエンドへのリクエストに変換され、その結果として、データがデータベースに保存されたり、問い合わせの結果が返されたり、データが操作されたりします。

たとえば、Web アプリケーションの場合、サーバーがリクエストされた操作を実行し、Web ページを HTML 形式で組み立ててクライアントに返します。その後、ブラウザがコードを解釈し、ユーザー向けに画面を表示します。

モバイルデバイス向けのネイティブアプリケーションの場合、画面はデバイス独自のコントロールを使用して表示され、画面に表示される情報はサーバー上のアプリケーションによって生成されます。

## Web のフロントエンドとネイティブのフロントエンド



Web ブラウザーで実行されるフロントエンドアプリケーションは、デスクトップ PC かモバイルデバイスかにかかわらず、ネイティブで実行されるフロントエンドアプリケーションとはデザインや動作が異なります。

Web ブラウザーで実行されるとき、アプリケーションは一般にレスポンシブ (レスポンシブ Web デザイン) になります。つまり画面のサイズに合わせて表示が自動的に調整されます。

## Web のフロントエンドとネイティブのフロントエンド (続き)



ネイティブアプリケーション、つまりデバイスのネイティブ言語でプログラミングされたアプリケーションは Web ブラウザーを使用せず、プラットフォーム固有のコンポーネントを使用して画面がデザインされます。

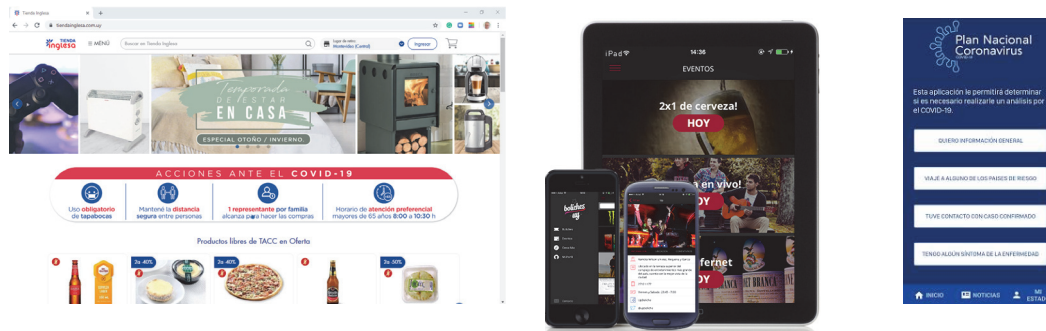
Web アプリケーションに対するネイティブアプリケーションの最大のメリットは、実行と画面表示の速度、モバイルデバイスの GPS、カメラ、マイクなどネイティブのハードウェアリソースと、カレンダー、メモ帳、リマインダーなどのソフトウェアリソースへのアクセスです。

最近では、Progressive Web Applications (PWA) という新しいタイプのアプリケーションが登場しました。

PWA は Web アプリケーションの進化系であり、ネイティブアプリケーションに近いユーザーエクスペリエンスを提供します。デバイスのリソースへのアクセスなど、これまでネイティブアプリケーションでのみ可能であった処理を実行でき、クライアントデバイスへのインストールやオフラインでの操作が可能です。

GeneXus では、この 3 種類すべてのアプリケーションをほとんど意識せずに開発できます。複雑に思えるかもしれませんが、想像以上にシンプルです。

## デジタルアプリケーションのフロントエンドの種類



顧客向けアプリケーション

## フロントエンド

デジタルアプリケーションのパーツに注目したあとは、対象とするユーザーによって分類する方法を説明します。

消費者や顧客など、デジタルに精通していない人々を含むユーザーを対象とするアプリケーションを顧客向けアプリケーションと呼びます。

このタイプのアプリケーションは、使いやすさ、双方向性、慎重なデザインを特徴とし、ユーザーが目にするものに好感を持ち、その操作のエクスペリエンスを楽しむことを目指しています。情報が明確かつ迅速に表示され、各機能へのアクセスが直観的で、操作を容易に理解する必要があります。

パフォーマンスとユーザーエクスペリエンスを可能な限り高めることで、ユーザーのロイヤルティと、アプリケーションをまた使いたいと思われるかどうかが決まるため、この領域には労力を惜しみません。Web アプリケーションとモバイルアプリケーションのいずれでも、顧客に最高水準のエクスペリエンスを提供しようとします。組織の収益や存在意義は顧客あってのものです。





## アプリケーションのパーツのプログラミング言語とツール



市場には、アプリケーションのフロントエンドとバックエンドを開発するためのさまざまなツールが展開されています。特に広く使用されているものをスライドに示します。

左側はフロントエンド用です。右側がバックエンド用です。さらにデータベースも多数あります。

ここではそれぞれの詳しい説明は省きます。これらについてマスターする必要はないからです。GeneXus を使用すれば、希望するプラットフォームに関する知識がなくてもデジタルソリューションを実装できます。GeneXus という単一のツール/言語/プラットフォームを理解していれば十分です。

次は、GeneXus について説明します。