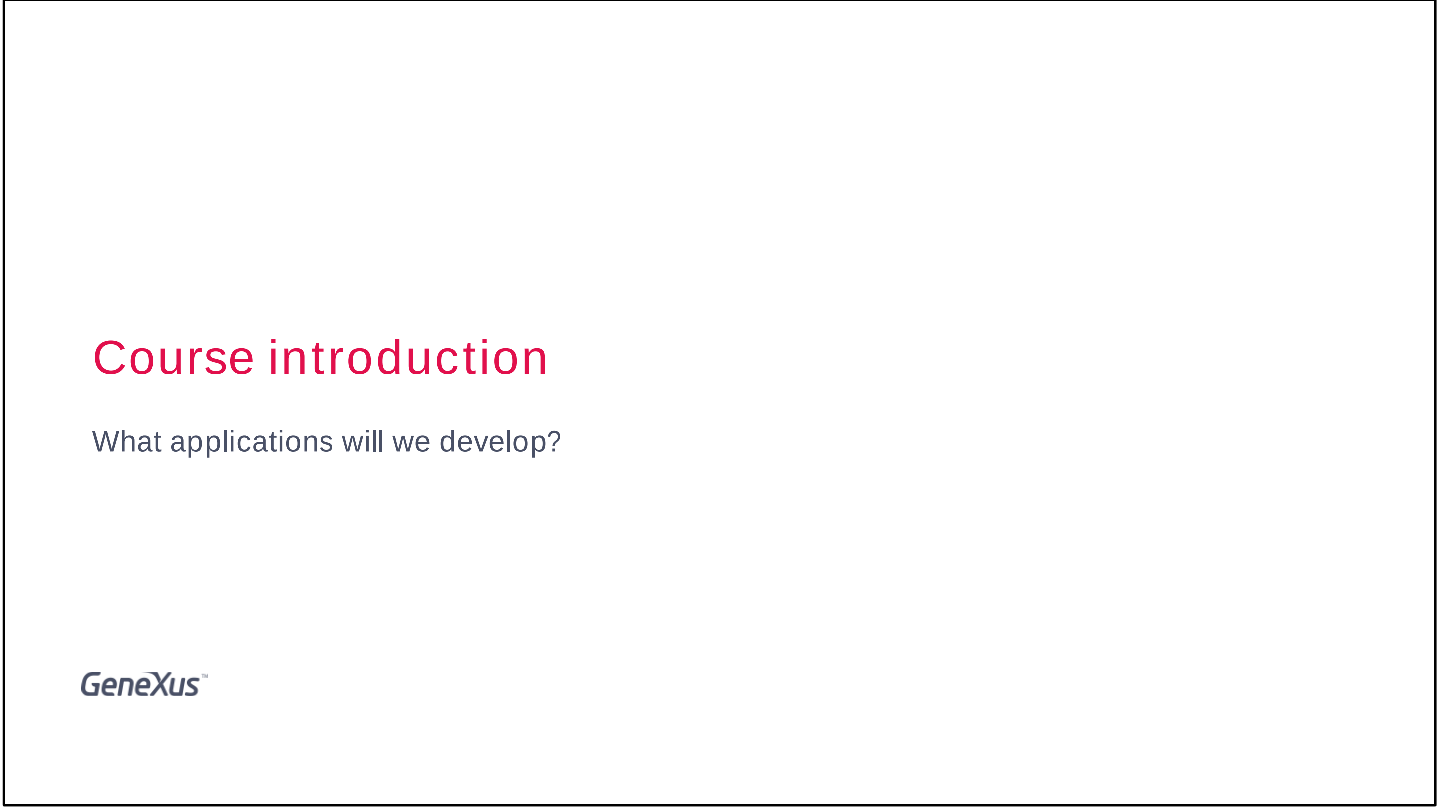
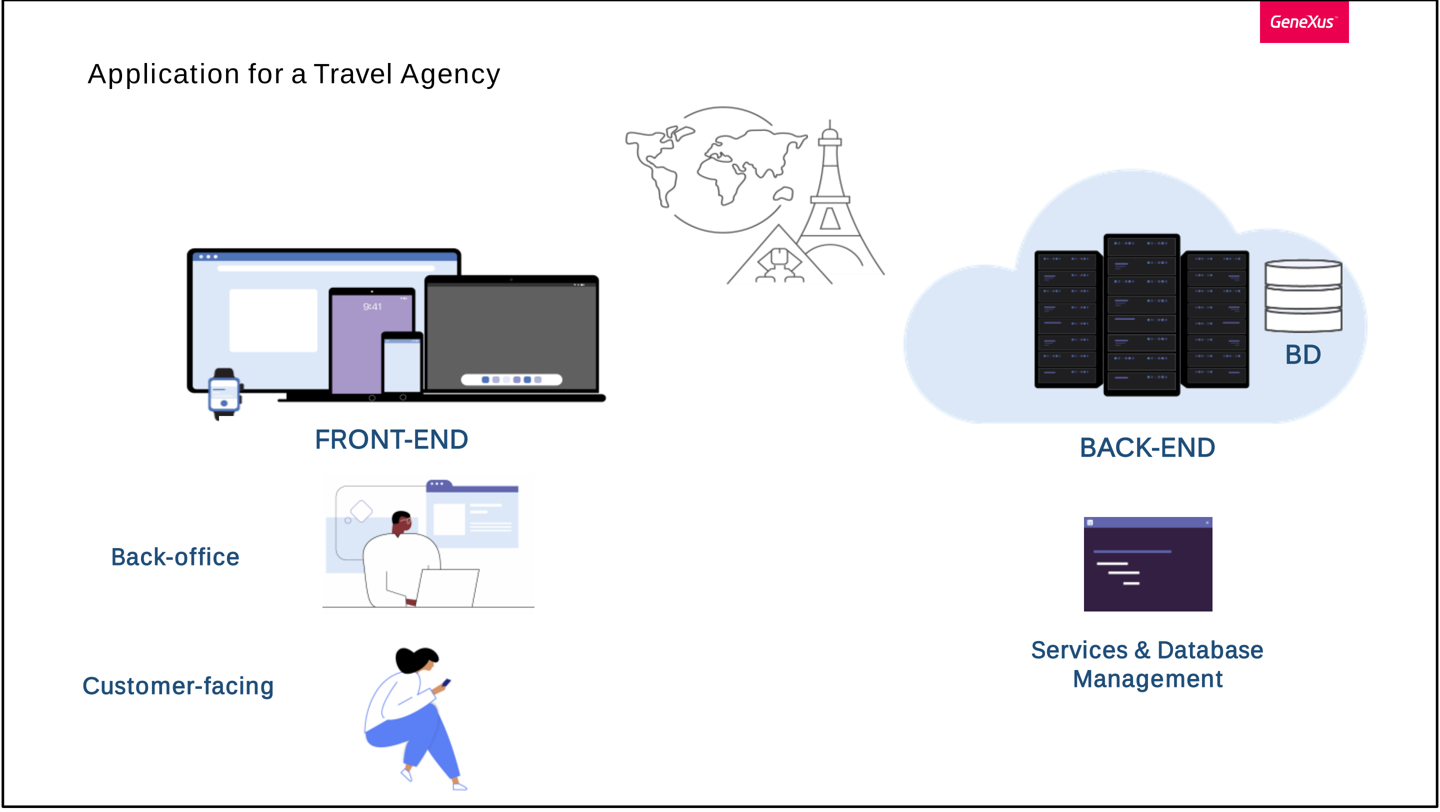


Course introduction

What applications will we develop?

In order to study the various topics of the course, we will be developing an application for a Travel Agency.



Application for a Travel Agency

FRONT-END

BACK-END

BD

Back-office

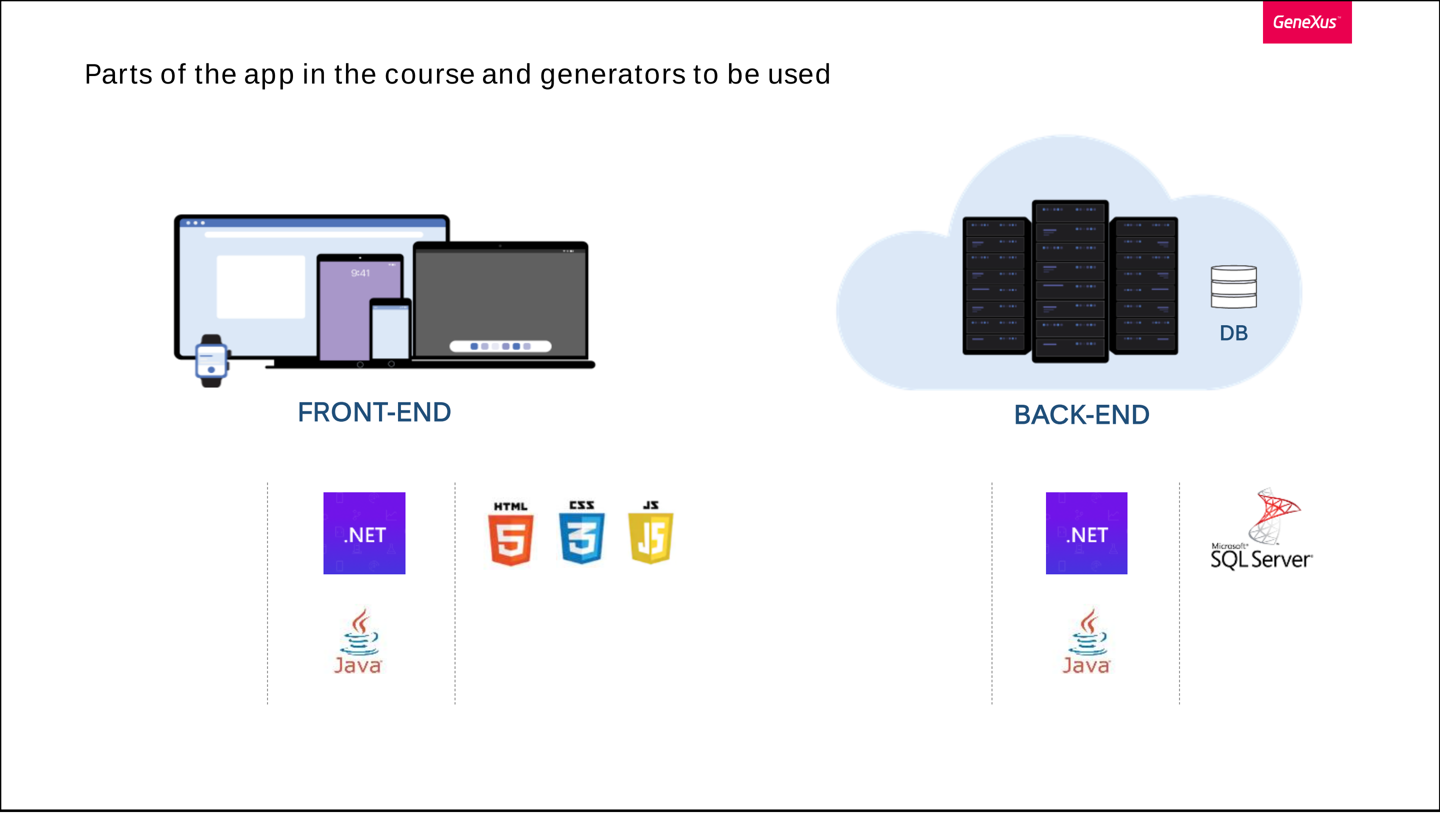
Customer-facing

Services & Database Management

The idea is to abide by a development as real as possible, as an excuse to learn about the platform and its language.

Let′s create the front-end with which users will interact by means of a web navigator or a mobile device, in addition to the back-end, which will include the programs on the server and the creation and maintenance of structures in the database. As you will see, GeneXus provides you with all this in one, making this itemization irrelevant. GeneXus generates everything necessary for the app to function without the need for you to be concerned about each part of the project in a separate manner.

You will develop a part of the app to be used only by the staff at the travel agency, so they may enter the data about countries, cities and tourist attractions offered, as well as information on flights, customers and so on. It is the one known as Back-Office. But we will keep in mind, even if we don't implement it, the part meant for end users, known as customer-facing, which enables queries on tours available and on the main tourist attractions in each city.



Parts of the app in the course and generators to be used

FRONT-END

BACK-END

DB

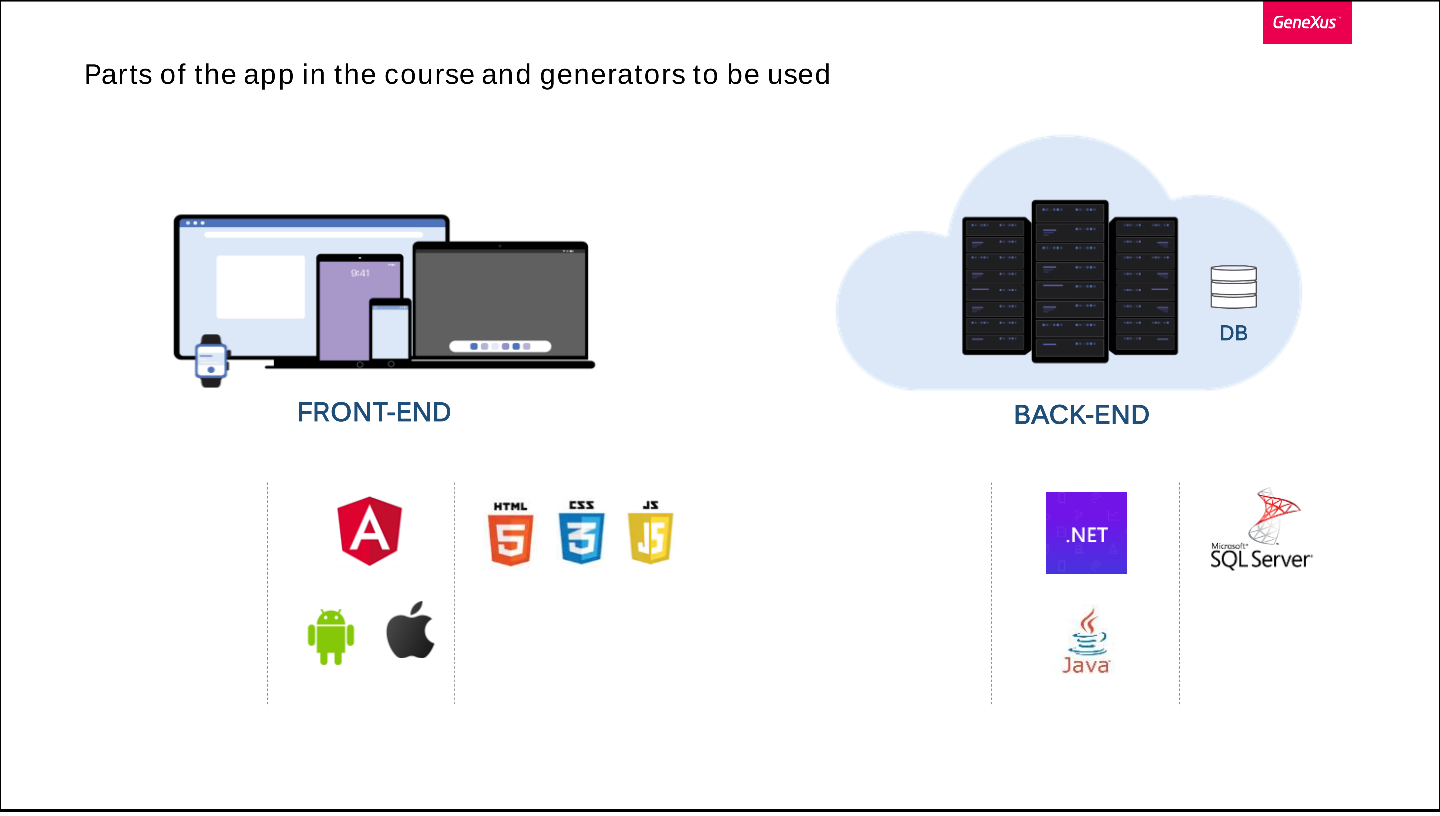
As mentioned before, GeneXus is multi-platform, meaning that you may generate your app in different languages or programming frameworks, for varied platforms and for different databases.

However, in this course:

For the case of the app's back-end for the travel agency, you will be generating in .NET language, using the SQL Server database.

For the front-end, you will use .Net to generate the back-office pages that include HTML, CSS and JavaScript.

Using Java would be exactly the same.



Parts of the app in the course and generators to be used

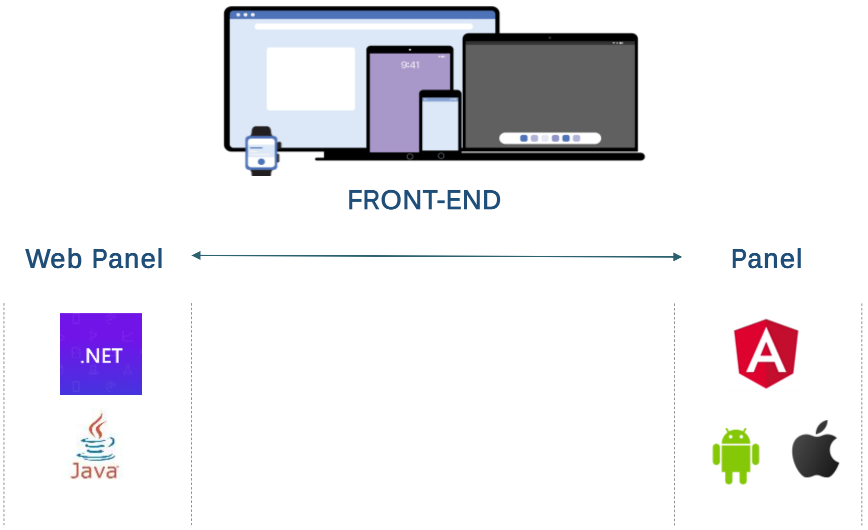
FRONT-END

BACK-END

DB

But we could also think about using Angular for the front end, or even implement a native –not web– front end for mobile devices. While this is possible, it requires using different objects for UI screens.

Differences between Front-end with Web Panels and Panels



FRONT-END

Web Panel

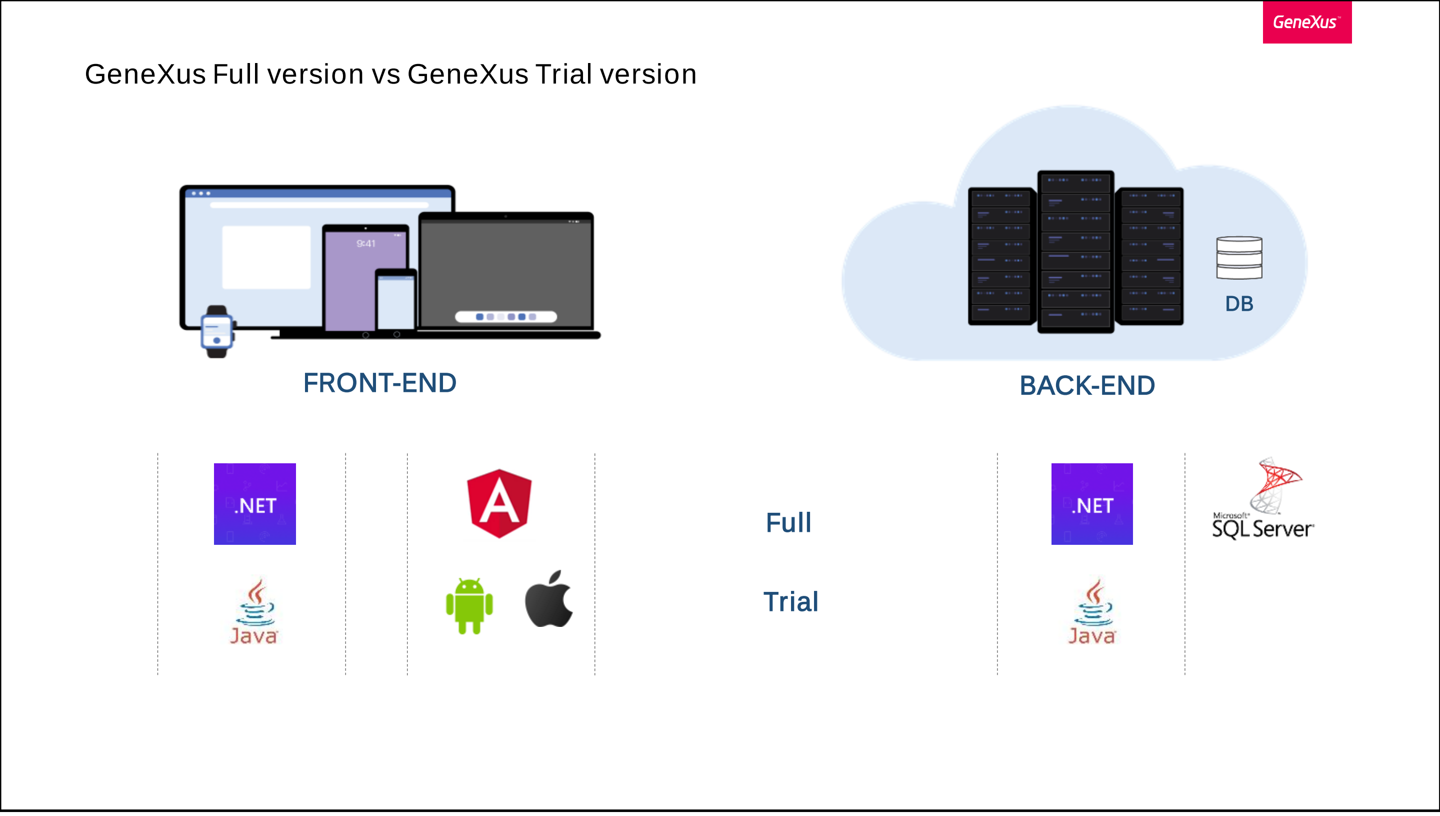
Panel

With no intention of going into this now, and just to get an idea: if we want to implement a web front-end screen for .Net or Java, we will use the GeneXus object known as Web Panel; meanwhile, if we want to implement that same web screen for Angular, or if we want to implement it natively (in Android or Apple), we will resort to the object simply known as Panel. These objects have exactly the same function but have slightly different programming logics in GeneXus.

Although the objective of this course is not to learn how to develop web applications with GeneXus for .Net or Java, but to learn the basics of application development with GeneXus –which means understanding the logic of this low-code platform, regardless of whether the final application will be web for .Net/Java or web Angular, or if it will be native mobile– since there are some differences we had to choose one of the two flavors, and we chose the web one, not Angular. We will implement in .Net, but what you will see here is exactly the same for Java. Exactly the same.

The only distinction we care about is the one we mentioned: if the front-end will be web (.Net or Java), or if it will be web Angular or native (Android, Apple). That's because we will use Web Panels or Panels, respectively. As for the rest, since GeneXus will automatically deploy our application in the language and platform we ask for, we won't really need to know practically anything about that language and platform. We will only need to know GeneXus.

In view of all the above, please note: if you are going to take this course because you are interested in developing Angular or native applications, we recommend that you keep the attached document open. In it, for each video of the course we indicate its relationship with Angular or with the native mobile application.



GeneXus Full version vs GeneXus Trial version

FRONT-END

BACK-END

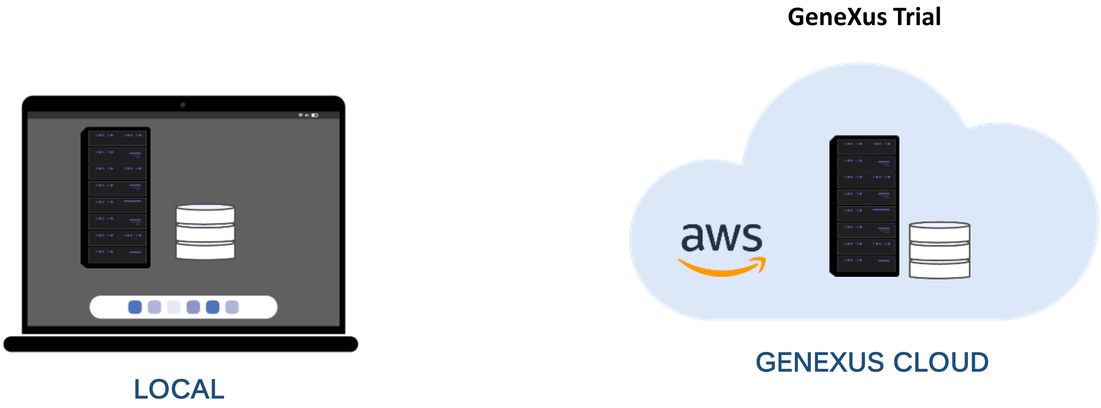
DB

Full

Trial

GeneXus includes a Full version and a Trial version that does not include all functions and allows for a restricted number of GeneXus objects to be included in your project, but it is totally free of charge. In order to enable everyone to do this course, we will be using the Trial version at all times, except in the case of explanations of some very specific issues.

Prototyping the app



For prototyping purposes, you may do it locally (with the server and the database located in the development machine) or in the cloud (with the server and the database located in the GeneXus cloud). This will allow you to skip the step of installing the software for the web server and the database in your computer.

And specifically, you may only prototype in the cloud with the trial version (GeneXus has the Amazon cloud available), but this will make the installation a lot easier, in addition to allowing you access to the data entered in the app.

GeneXus™

training.genexus.com
wiki.genexus.com