

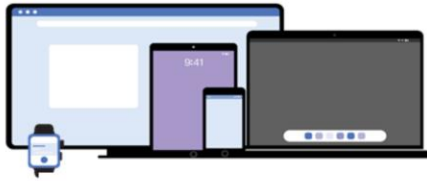
Course introduction

The apps that will be developed

GeneXus™

In order to study the various topics of the course, we will be developing an application for a Travel Agency. The idea is to abide by a development as real as possible, as an excuse to learn about the platform and its language.

Application for a Travel Agency

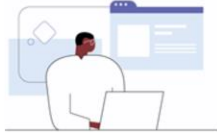


FRONT-END



BACK-END

Back-office &
Customer-facing
(WEB)



Back-office &
Customer-facing
(MOBILE)



Services & Database
Management

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. Another part to include is one meant for end users, known as customer-facing, which enables queries on tours available and on the main tourist attractions in each city.

Both apps (back-office and customer-facing) are developed for a web environment and some functions in both parts will also be available for access from mobile devices.

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



FRONT-END



BACK-END

BACK-OFFICE (WEB)



CUSTOMER-FACING (WEB)



CUSTOMER-FACING (MOBILE)



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

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; and for the customer-facing part of the app, you will be using the Angular framework to generate the web part and Android to generate the mobile part.

Despite the mention to these generators for describing what GeneXus does automatically, in fact, you need not know anything about these languages to do this course.

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.

Just to inform you: the new GeneXus M –the GeneXus multi-platform IDE capable of executing on Linux or on a Mac- will be available shortly, though it will not be applied in this course.

Prototyping the app



LOCAL



GENEXUS CLOUD

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, from mobile devices.

GeneXus™

training.genexus.com
wiki.genexus.com
training.genexus.com/certifications