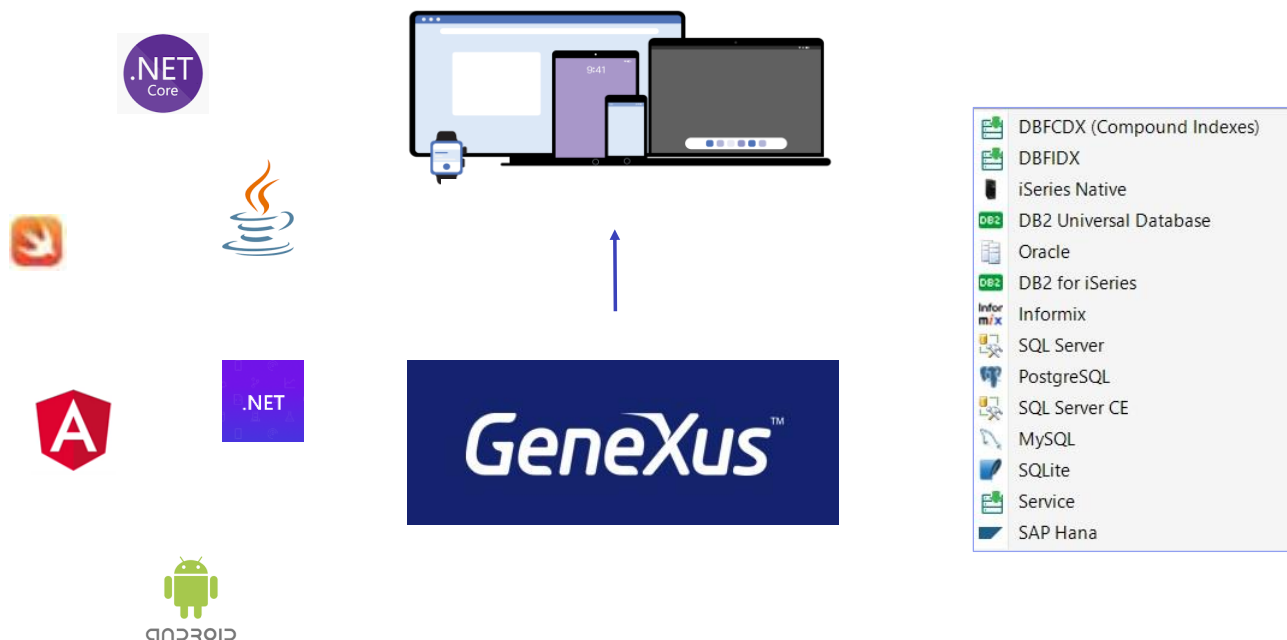


Introdução aos Environments

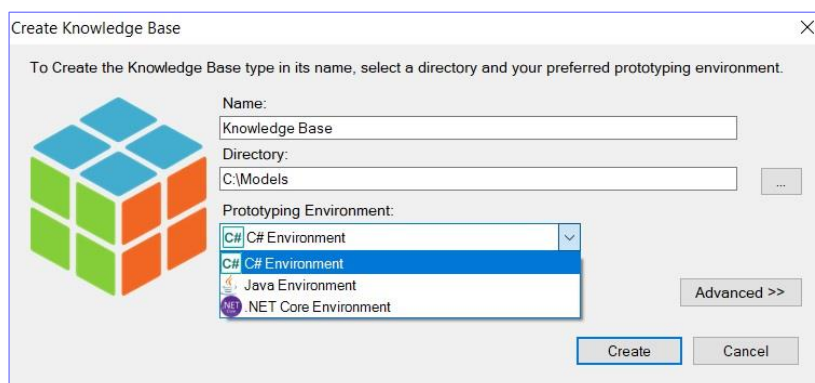
GeneXus™



Como já sabemos, um dos grandes benefícios de GeneXus é que permite gerar aplicações para diferentes plataformas, com interface web ou para dispositivos móveis, gerando código em diferentes linguagens de programação ou armazenando os dados da aplicação em diferentes bases de dados.

Todas estas informações são definidas em um Environment, um ambiente de execução.

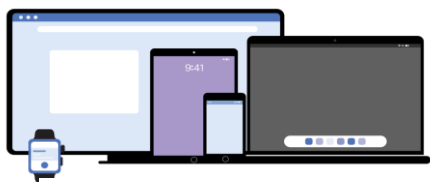
GeneXus Full version - Environments



Lembremos que GeneXus conta com uma versão full e uma versão de teste que tem certas limitações, entre as quais se encontram as linguagens de programação e bases de dados disponíveis.

Bem. No momento de criação da base de conhecimento, é criado um ambiente padrão. Se temos uma versão full de GeneXus, então a caixa de diálogo solicita ao usuário que selecione a linguagem de programação que será utilizada para o Back end da aplicação. O restante das informações deverá ser configurado por meio de suas propriedades.

Front end e Back end



FRONT-END

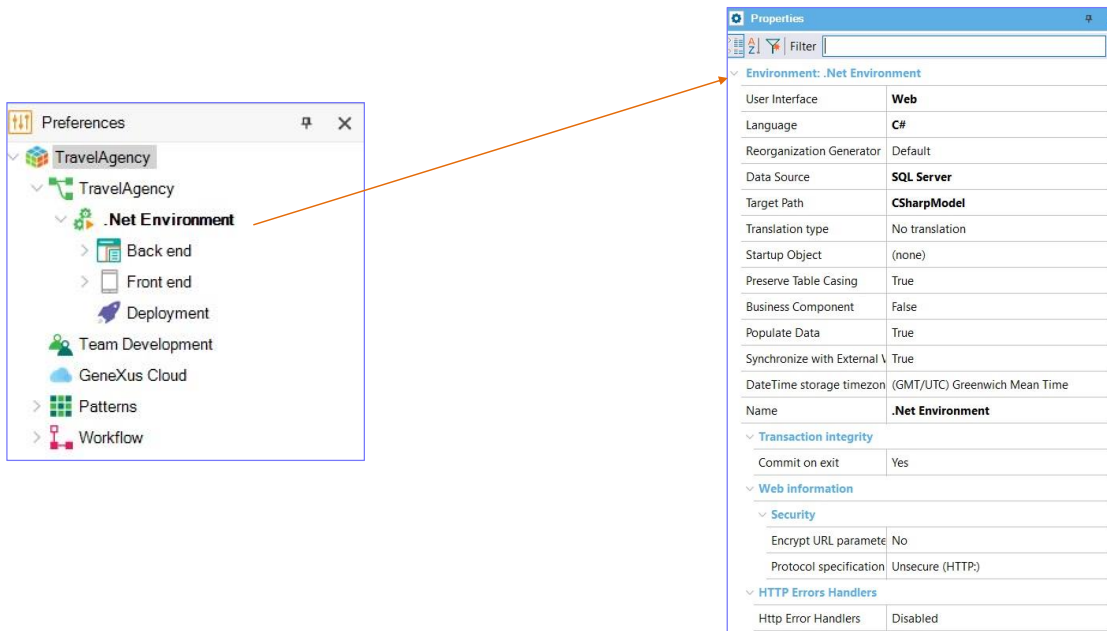


BACK-END



Lembremos que o Front end é a parte da aplicação que interage com os usuários, enquanto o Back end de uma aplicação é responsável por toda a sua lógica e programas, bem como pela conexão com a base de dados, servidores, etc.

GeneXus Full version - Environments



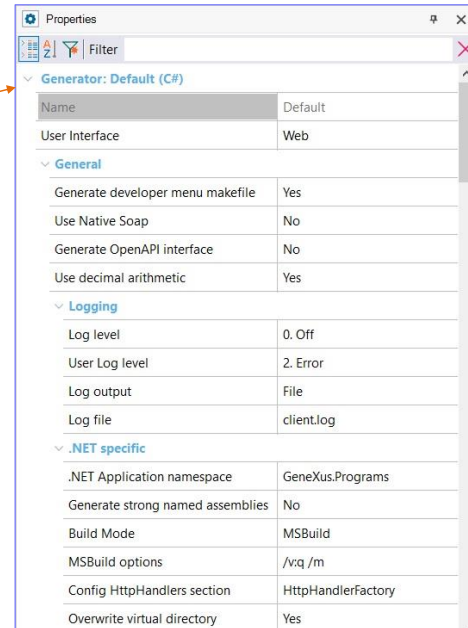
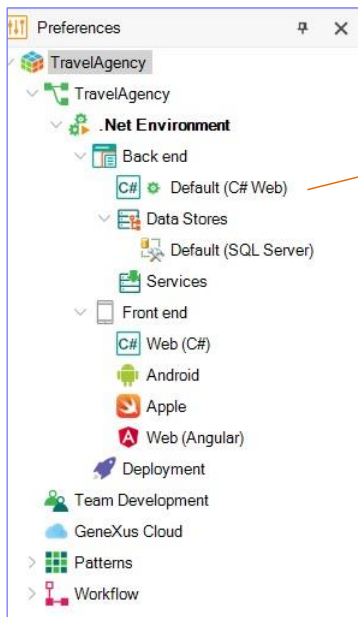
Já no IDE de uma versão full de GeneXus, todas as informações do Environment se encontram na janela de Preferences. Se observamos, o Environment tem definidas várias propriedades que determinam a plataforma de execução e suas características

.

Um Environment é composto pelos seguintes nós:

- Back end
- Front end
- E Deployment

GeneXus Full version – Environments

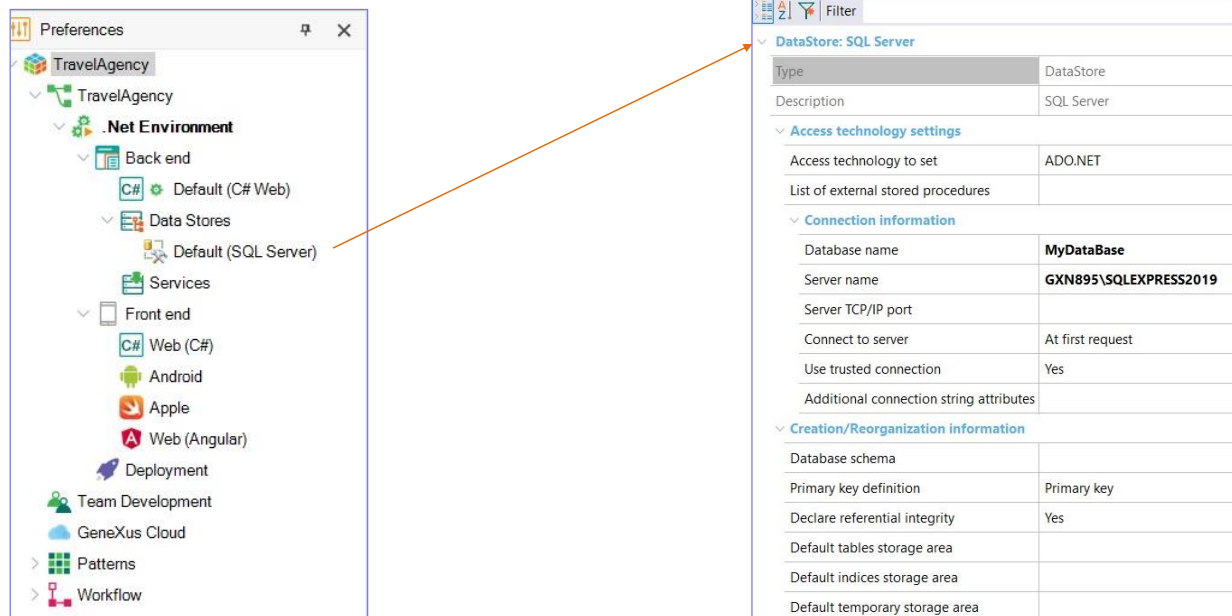


No nó Back end se definem as linguagens de programação que serão utilizadas para sua geração. Poderia ser C#, Java, .Net Core. Já vemos o gerador padrão, que foi o indicado no momento de criação da base de conhecimento.

É possível definir mais geradores sob este nó, simplesmente clicando com o botão direito e escolhendo New generator.

Nas propriedades do gerador são definidas as características dos objetos gerados, do servidor ou da linguagem de programação a ser utilizada.

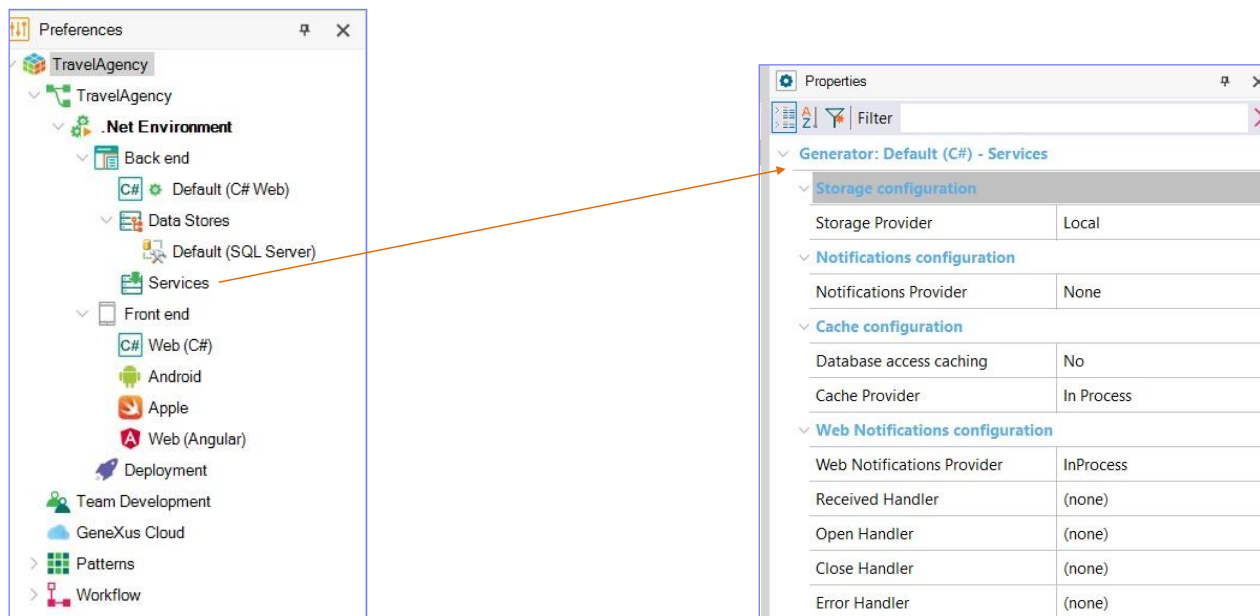
GeneXus Full version – Environments



Passemos agora para o nó Data Stores. Aqui se define o tipo de base de dados onde guardaremos os dados da nossa aplicação, e também é possível definir outras bases de dados às quais acessar para consultar informações.

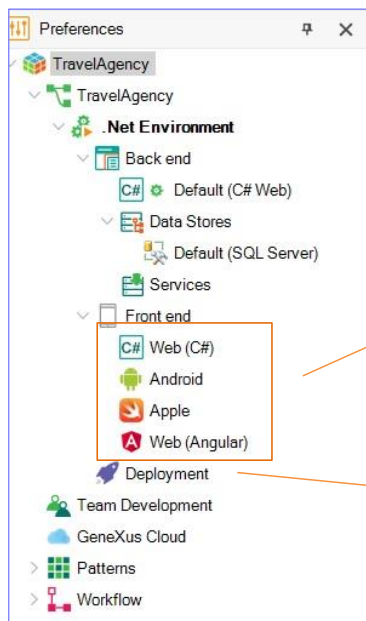
Se observamos as propriedades do Data Store, vemos que podemos definir os dados de conexão à base de dados, como o endereço do servidor, usuário, senha, versão da base de dados, etc.

GeneXus Full version – Environments



Quanto ao nó Services, em suas propriedades podemos configurar o manuseio de serviços, configurações de armazenamento, notificações, etc.

GeneXus Full version – Environments



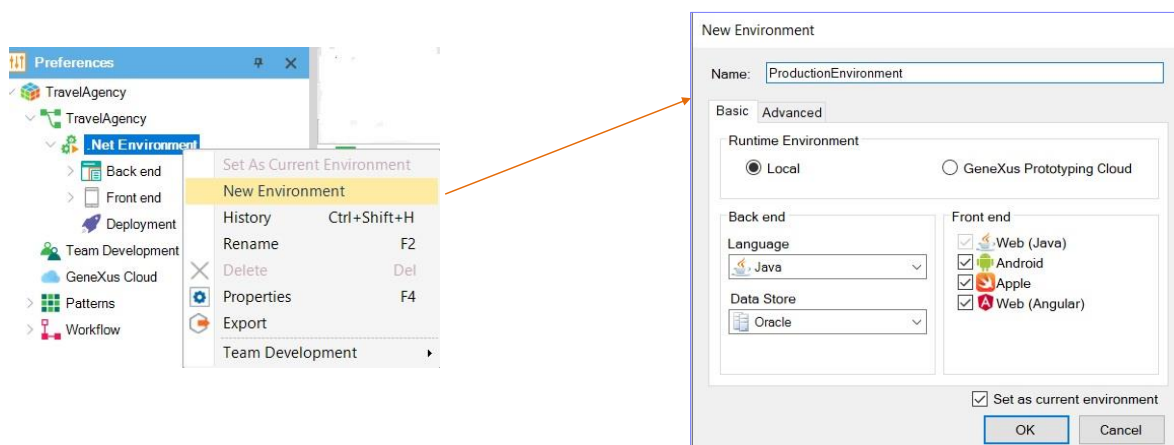
Geradores disponíveis para
desenho de front-end.

Unidade de Deployment:
indica o conjunto de objetos a
serem integrados ao fazer o deploy.

Passemos agora para o nó Front end.

Aqui vemos o conjunto de geradores disponíveis para o desenho deste componente da aplicação e, finalmente, o nó Deployment, onde podem ser definidas diferentes unidades de deployment. Vale ressaltar que uma unidade de Deployment indica o conjunto de objetos a serem integrados ao fazer o deploy.

GeneXus Full version – Como definir um novo Environment



GeneXus nos permite definir diferentes ambientes de execução para a mesma base de conhecimento

Por exemplo, é possível definir um ambiente de execução para a etapa de desenvolvimento onde nos conectamos a uma base com dados de teste e, em seguida, outro ambiente de produção onde se define o servidor e base de dados para a aplicação concluída. Então, vamos definir um novo ambiente.

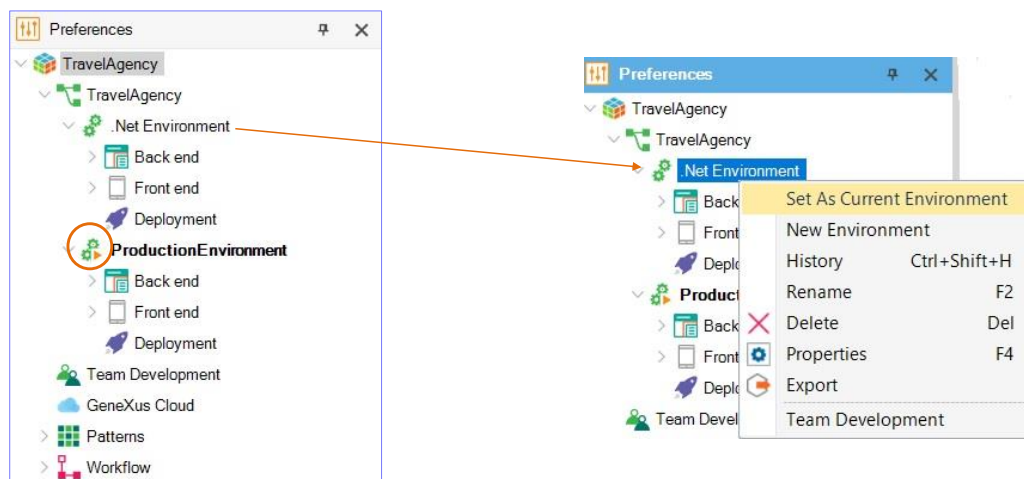
Para isso nos posicionamos sobre o nome do ambiente já definido, clicamos com o botão direito e selecionamos New Environment.

Devemos indicar:

- O nome do novo ambiente. Vamos indicar que é um ambiente de produção
- Depois devemos indicar se vai ser prototipado localmente ou na nuvem de GeneXus.
- Escolher a linguagem para o Back end, e
- Selecionar o Data Store que será utilizado,
- e os geradores para o Front end.

Observemos que no caso de selecionar que será prototipado na nuvem GeneXus, então, fica desabilitada a possibilidade de escolher o Data Store.

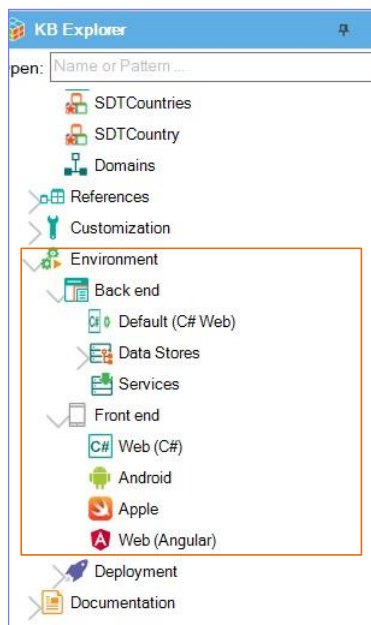
GeneXus Full version – Current Environment



Uma vez definido o novo ambiente, observemos que aquele com o nome destacado, e o ícone de play, será o ambiente ativo e aquele que será aplicado ao pressionar F5.

No caso de querer construir a aplicação em outro ambiente, basta nos posicionar no nome do referido ambiente, clicar com o botão direito, Set as current environment.

GeneXus Trial version – Default Environment



Finalmente, no caso de estar utilizando a versão de teste de GeneXus, não é possível definir novos Environments. No momento de criação da base de conhecimento é definido por padrão, o único ambiente de execução que prototipará na nuvem de GeneXus, terá C# como gerador para o Back end e, C#, Android, Apple e Angular para a geração do Front end.

Se formos agora ao IDE da versão de teste de GeneXus, vemos que o nó de Environments está integrado na janela KBExplorer

Podemos ver as opções do Back end e também do Front end. Se clicamos com o botão direito vemos que não temos disponível a possibilidade de definir um novo ambiente.



training.genexus.com

wiki.genexus.com

training.genexus.com/certifications