**Creating ADF Page Templates**

## Why Page Templates

In our major applications the challenge that we face is ensuring consistency in terms of look and feel on all the pages.

In ADF Page templates allows us to create a basic layout that will be used on all the pages.

All the other pages in the application will be based on this main template to allow reusability in the application. This ensures consistent look and feel in the application. All the pages are known to be referenced, that means, if any changes are made to the main template e.g some change in the image in the template ,all the pages based on that template will update automatically.

## Parts of page templates

Page templates has two parts

### Facets

These are the areas in the template where the users can place their own content. User cannot modify the other areas in the template

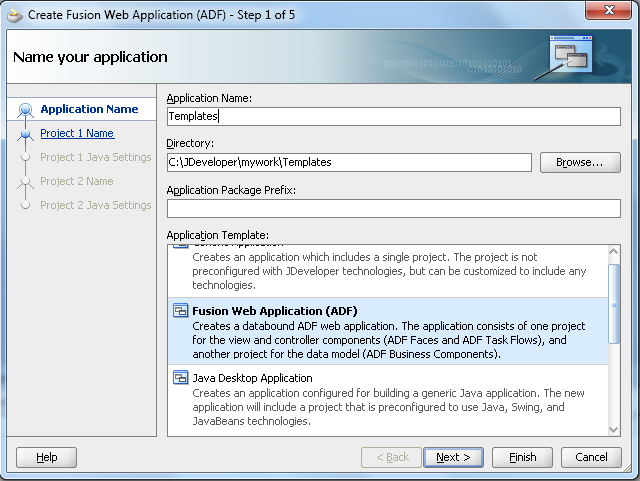
### Attributes

It’s a way to define the parameters or attributes to the template. e.g the areas that are not editable can be changed by passing parameters. Different users can define different labels on pages as per their convenience with the use of attribut

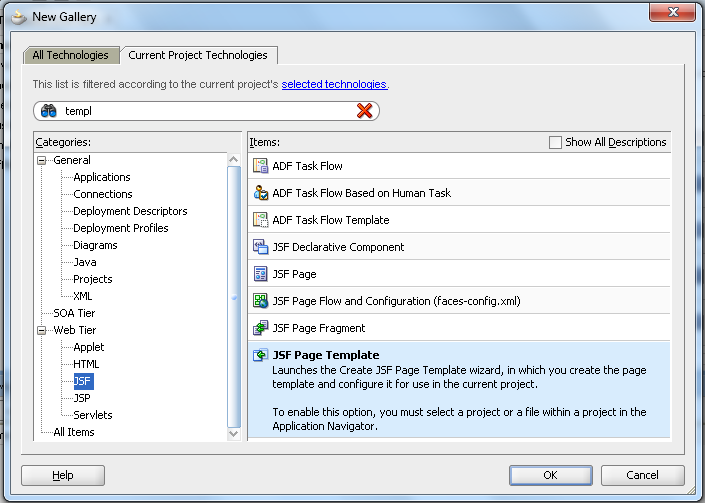
## Use Case

We will explain these with a use case. Here we will create a page template and then a jsp page based on that template.

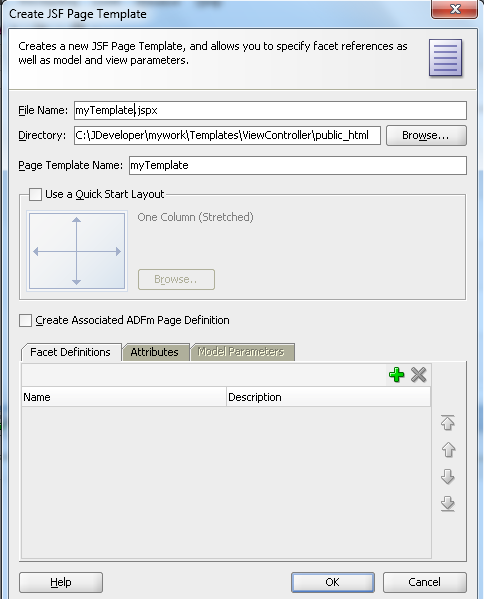
1. Create a fusion web application and name it as ‘Templates.’



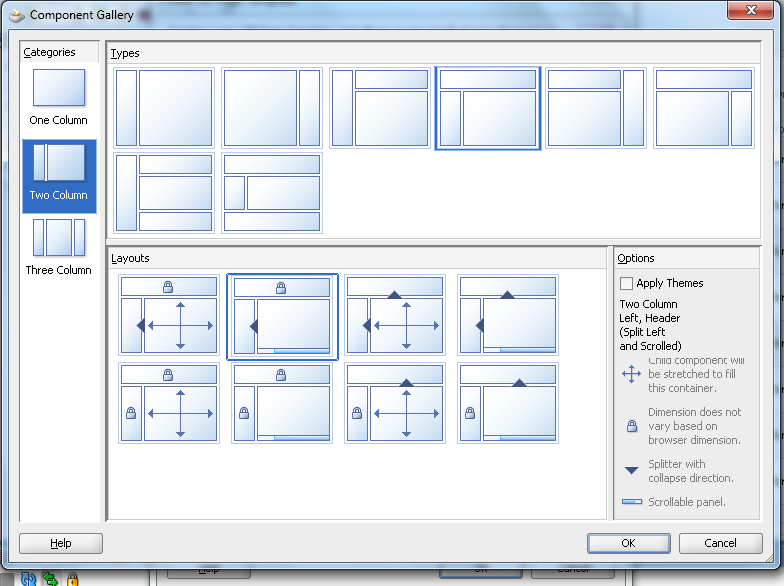
1. Create model and View Controller projects and finish the app creation. Right click on the View controller and create JSF page template.



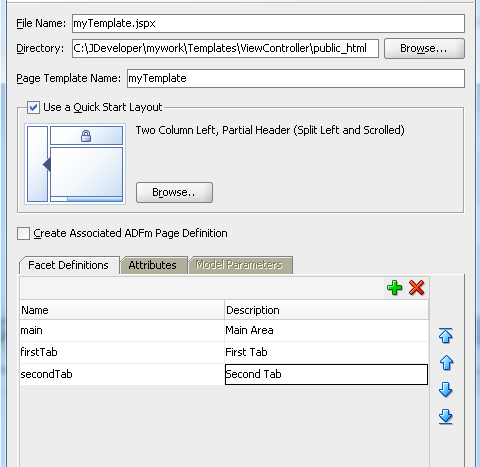
1. Name it as ‘myTemplate’



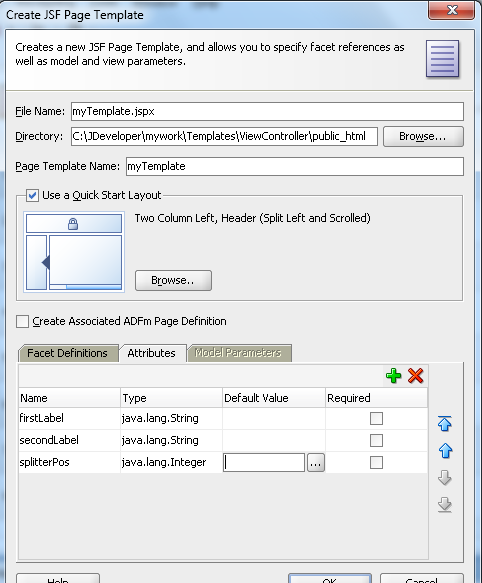
1. Select ‘Use a Quick Start Layout’ and select some default design .We will select ‘Two Column’ with the locked header



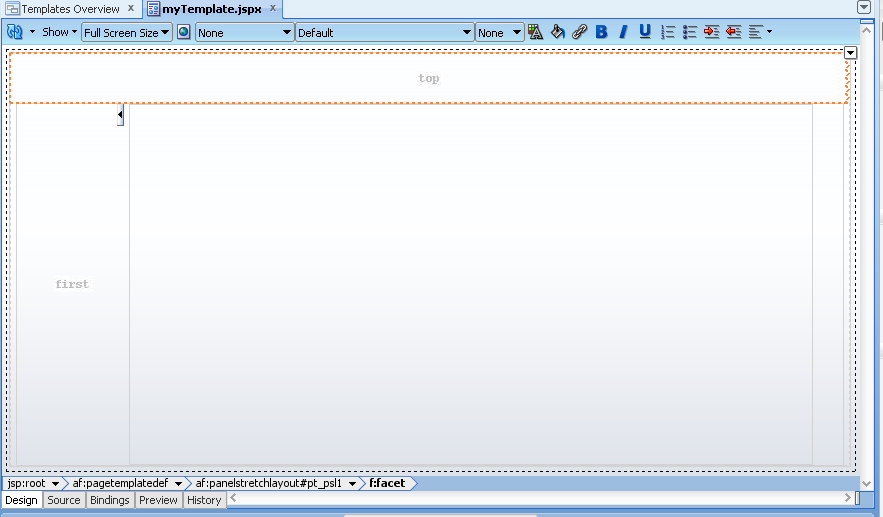
1. Define the facet attributes where we will allow the user to place his content. Here we will define 3 facets on the template. One will be the main area and we will have a panelTabbed component with 2 tabs, named as First and Second tab for reference



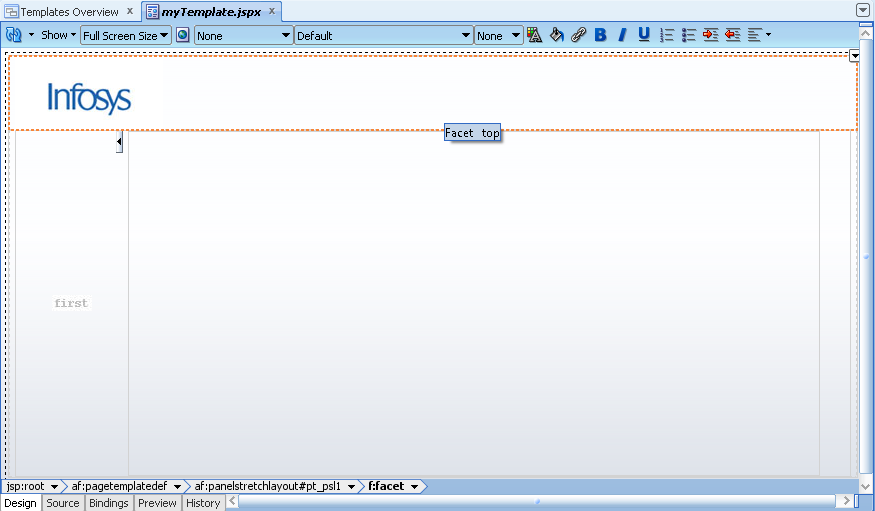
1. Define the attributes to be used, that will act as parameters for the users to pass different values on different pages. Two labels for the tabs and the splitter position.



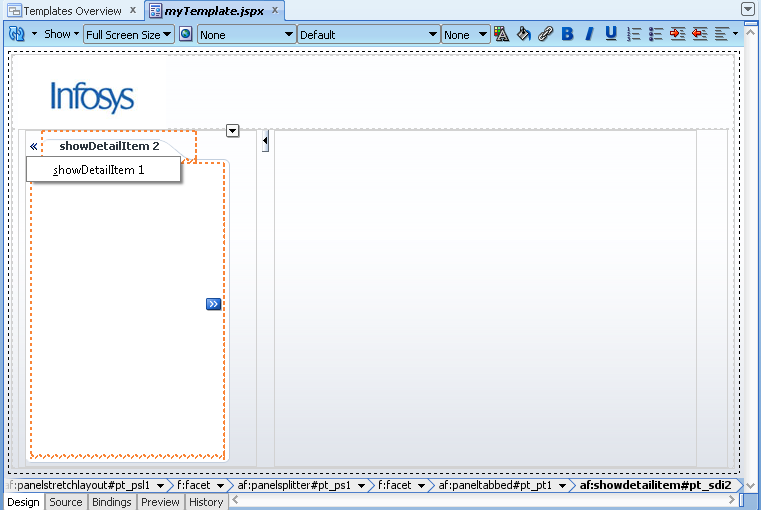
1. The page template will look like the one shown in the below screenshot



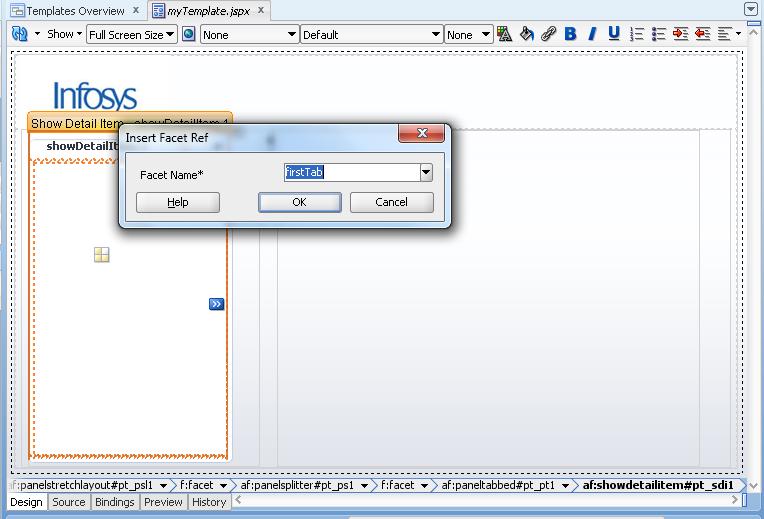
1. Place a company logo to the top area.

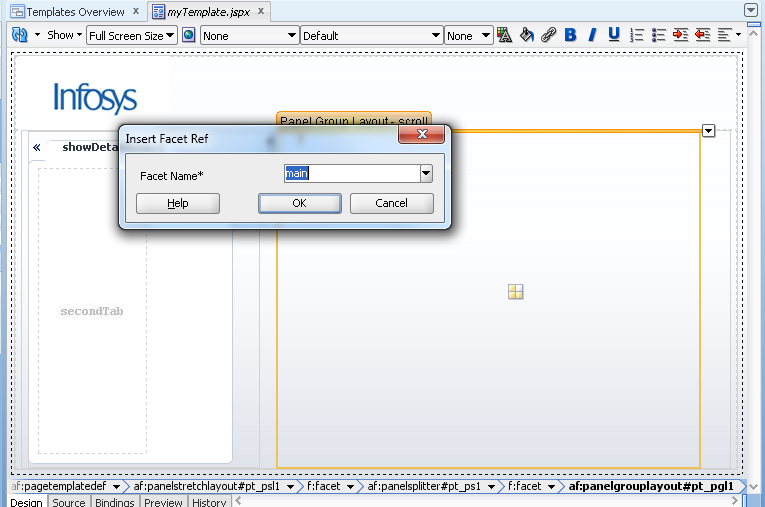


1. Now define the areas where the user can place his content. In the left hand side area, we will have a panel Tabbed component with 2 different tabs inside that.

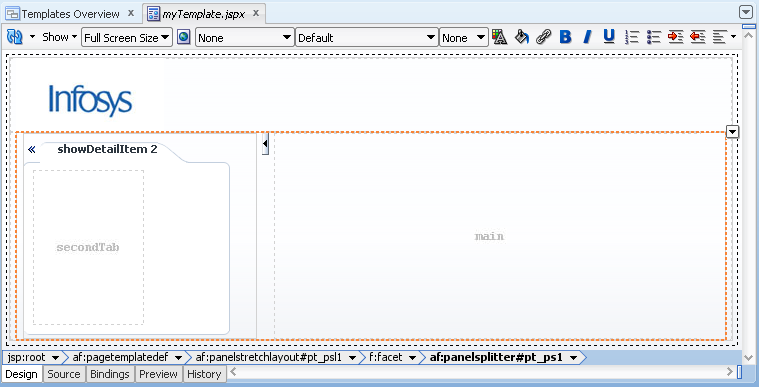


1. In the first tab we will have a facet named as the ‘fisrtTab’ as created earlier. Similarly we will have facets for ‘secondTab’ and ‘main’ area.

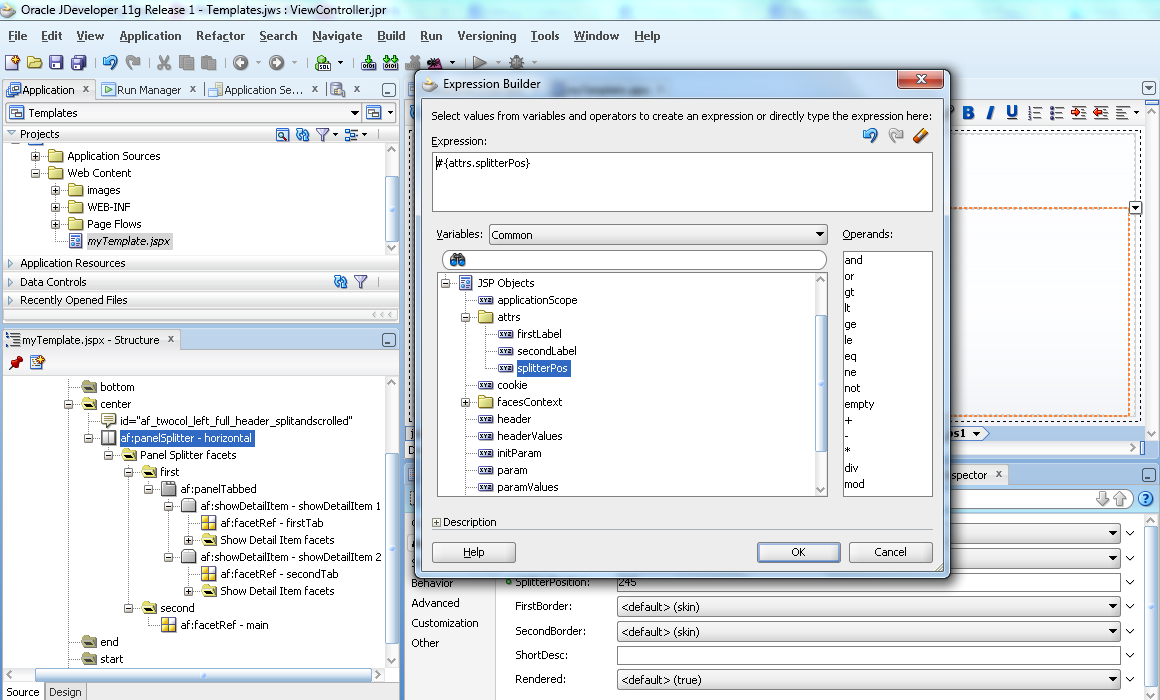




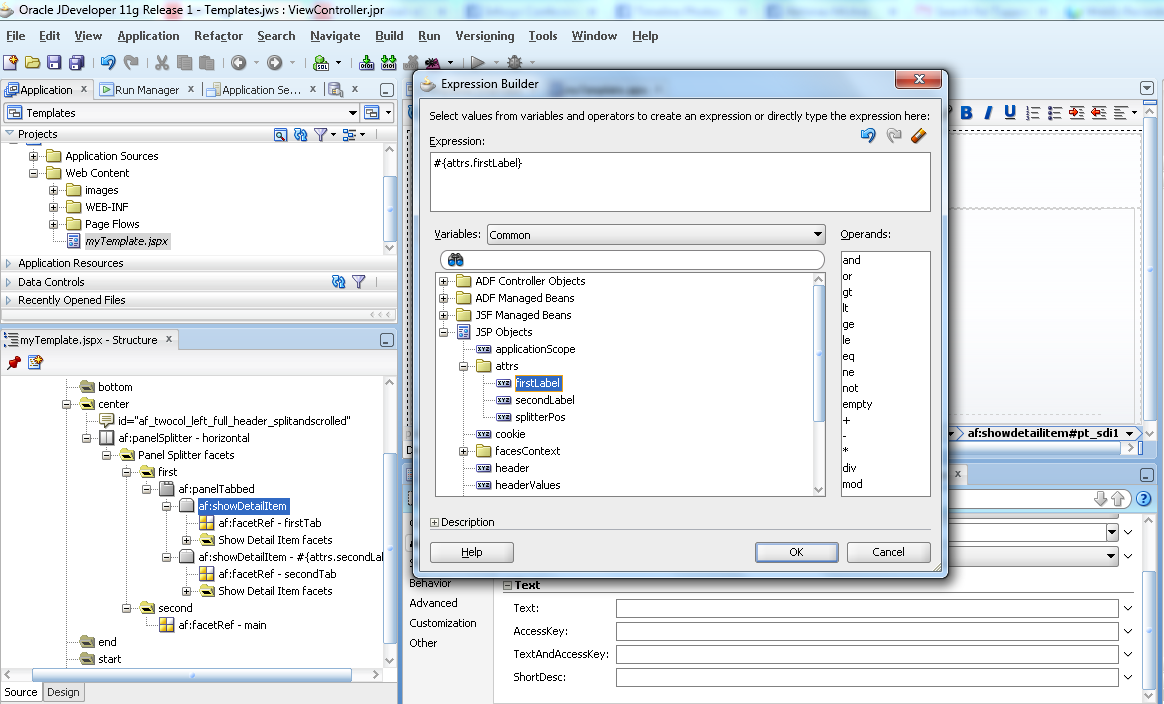
1. After placing the facets, the page will look like this.



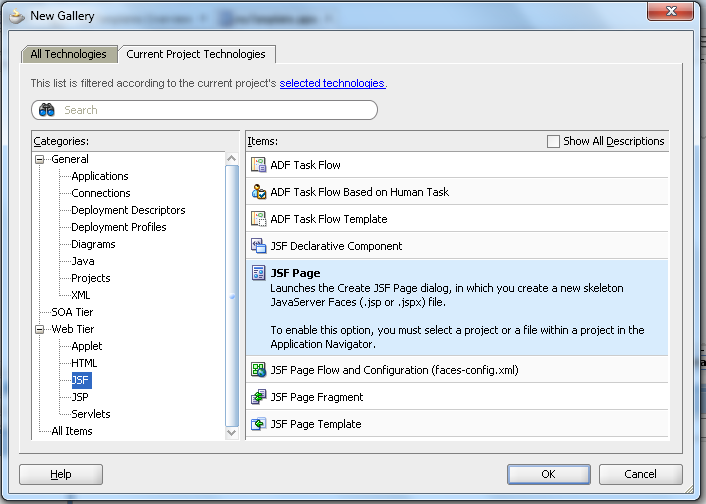
1. Now use the attributes defined earlier.Go to the properties and use the expression builder to define the splitter position.Use the splitterPos attribute. This means the user can define different positions of the splitter on different pages.



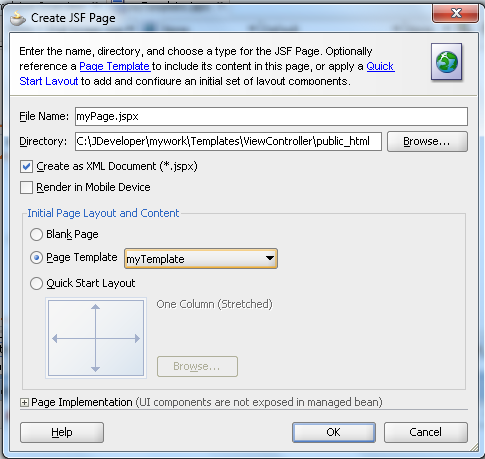
1. Similarly, user can define the labels using the attributes as shown.



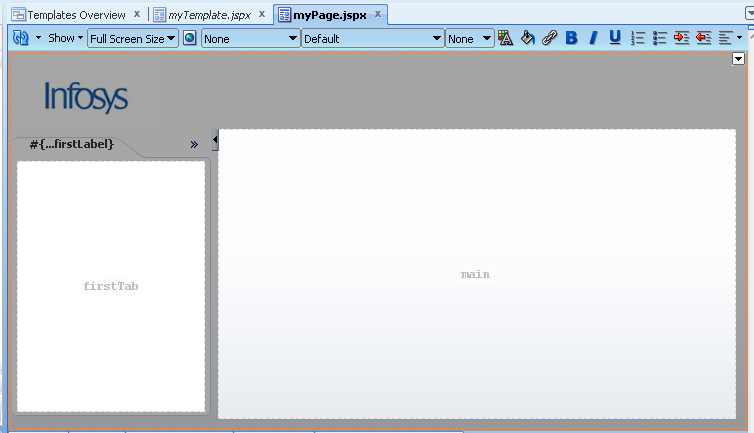
1. Now our template is ready to use for other pages. We will now create a JSF page. Click on view controller and create a new JSF Page.



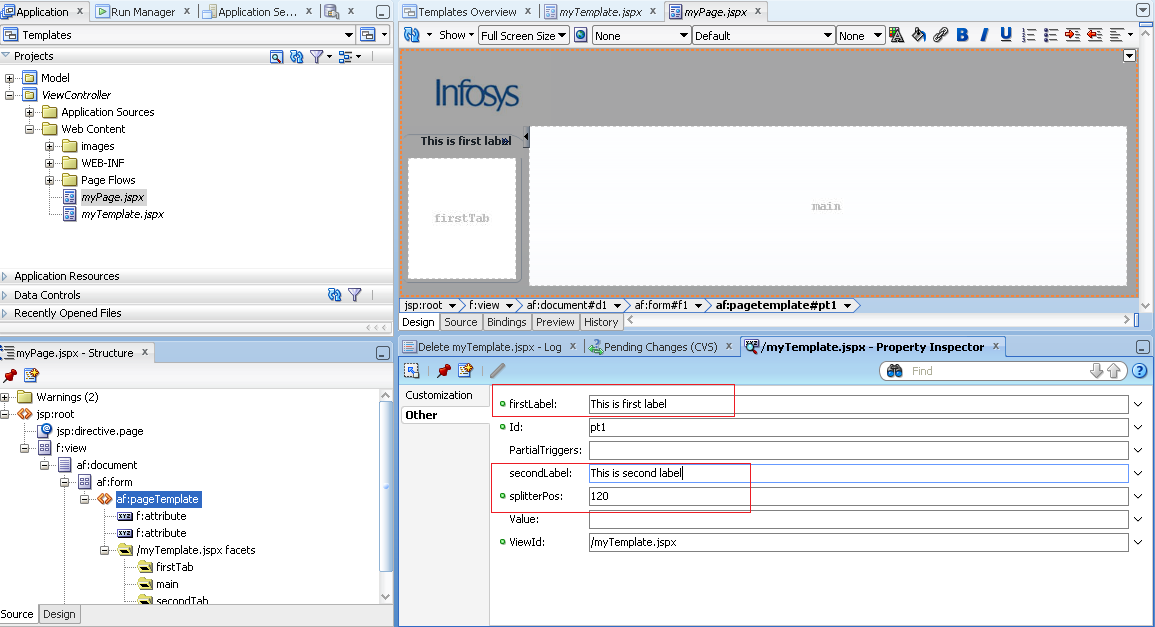
1. Name it as ‘myPage’ and select the page template that we created earlier i.e. ‘myTemplate’



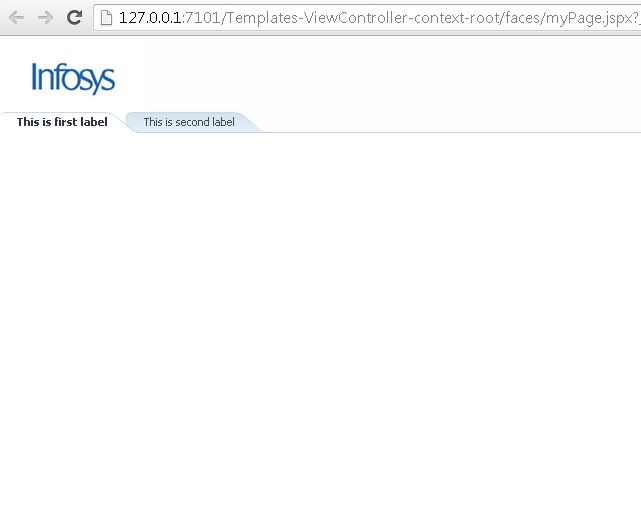
1. Now, we can see the grey areas, that are the part of the ‘pageTemplate’ and user cannot change these areas. Only place where user can place his content is the facet area defined.



1. In the properties, we can define the labels and the splitter post ion for this page as shown.



1. Run the page and the will look like:-



**References**: NA