

Finishing Up

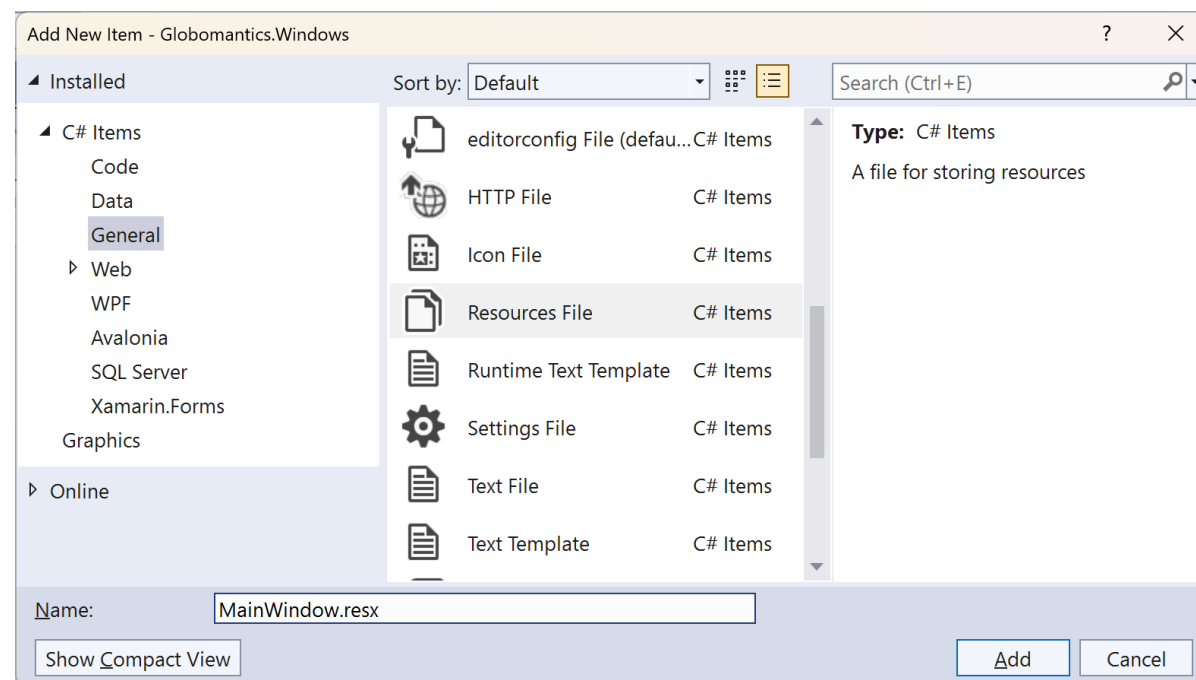


Filip Ekberg

Principal Consultant & CEO

@fekberg | fekberg.com





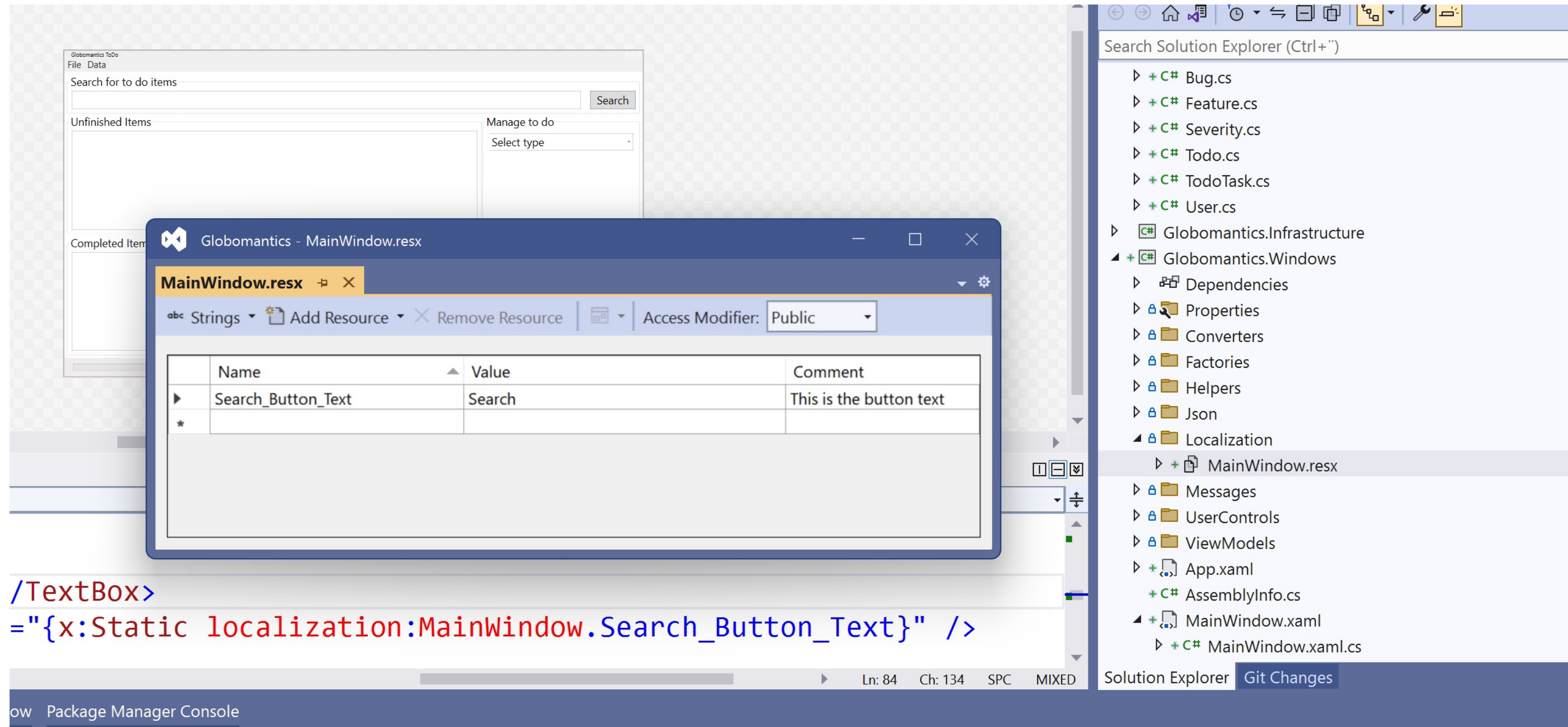
One resource file per use case

One resource file per language per use case

One default resource file per use case



Use Appropriate Naming



The screenshot shows a Visual Studio interface. In the background, a window titled 'Globomantics - MainWindow.resx' is open, displaying a table of resources. The table has columns for Name, Value, and Comment. The first row shows 'Search_Button_Text' with the value 'Search' and the comment 'This is the button text'. Below the table, there is a text box containing the XAML code: `</TextBox>` followed by `= "{x:Static localization:MainWindow.Search_Button_Text}" />`. The foreground shows the 'Solution Explorer' on the right side of the IDE, displaying a project structure. The project is named 'Globomantics' and contains several folders and files. The 'Localization' folder is expanded, showing the 'MainWindow.resx' file. The 'Solution Explorer' also shows a 'Git Changes' tab at the bottom.

Name	Value	Comment
Search_Button_Text	Search	This is the button text
*		

```
</TextBox>  
="{x:Static localization:MainWindow.Search_Button_Text}" />
```

Ln: 84 Ch: 134 SPC MIXED

ow Package Manager Console

Solution Explorer Git Changes



CultureInfo.InvariantCulture

“An invariant culture is culture-insensitive. Your application specifies the invariant culture by name using an empty string ("") or by its identifier. InvariantCulture defines an instance of the invariant culture.

It is associated with the English language but not with any country/region.

It is used in almost any method in the Globalization namespace that requires a culture.”

