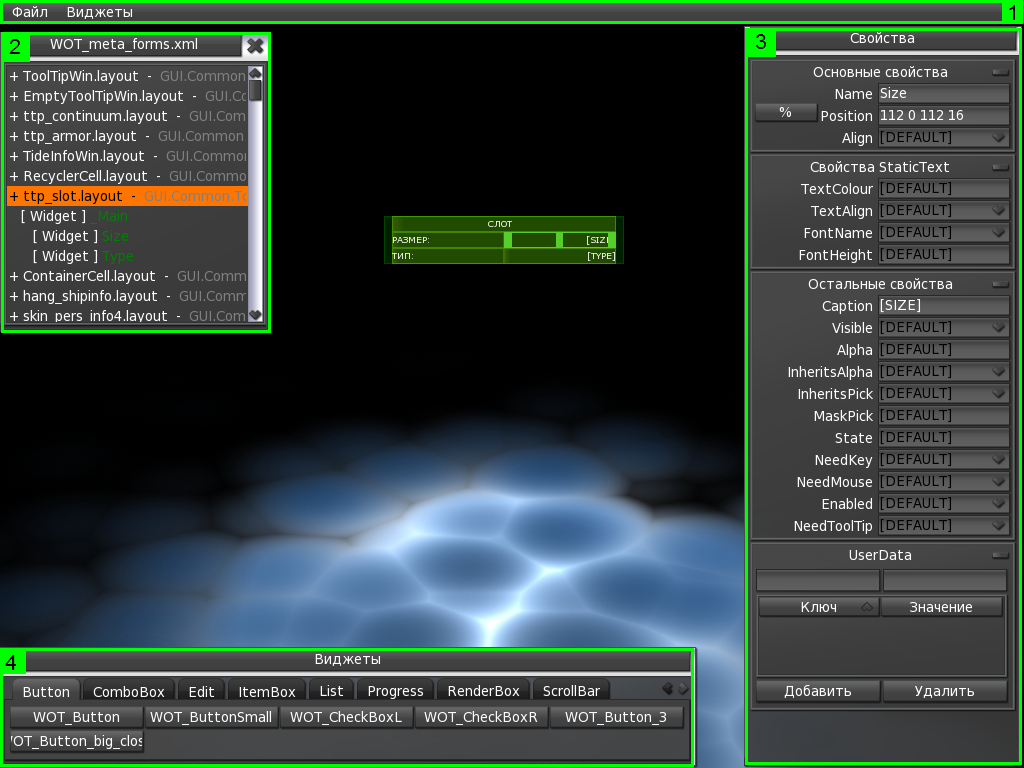
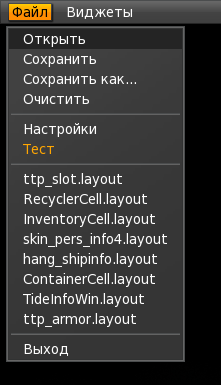
**Interface Layout Editor**  
  
Efficiency depends on the tools that we use . But even the most powerful tool will not help if they are not adept or misused . In this connection there is an interesting pattern , the more money provides a tool , so it is difficult. From this situation there are two ways , the first - a simple and intuitive interface , which is an art and the second - writing detailed documentation . Well, in this paper I will go the second way and tell you the highlights use tool that allows you to create user interfaces (forms ) . So start LayoutEditor ...  
  
  
Key terms and their meanings  
  
1. Widget - ( control) the common name of the element that makes up the form.  
2 . Form - ( layout ) widget hierarchy , usually with one main widget in the bud.  
3 . Skin - ( hide ) a combination of properties and visual elements , determines how the widget looks .  
  
  
Main panel and controls



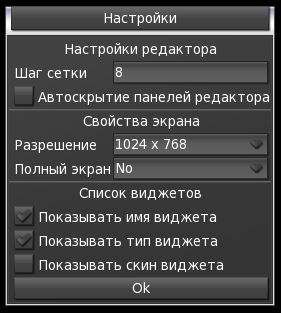
As we can see, the space online mainly consists of panels and menus. Some panels may be missing, some may appear when needed. It is also possible mode when the panel, loss of focus, I stop at the edge of the editor. Rest of the space available for editing and can contain arbitrary widgets, and forms.

**Menu (1)**

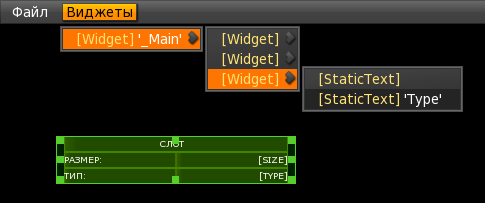
The menu is a classic version, if enabled auto-hidden, the menu is cleaned up. In the menu, there are two main sections:  
1. File - basic settings and commands  
2. Widgets - Create widget hierarchy  
  
  
Section "File" menu



Commands are grouped in the following order:  
1. Commands for opening, saving, and mold cleaning.  
2. Configure and test mode created forms  
3. List of last loaded forms  
4. exit  
  
When saving and opening a dialog window with a selection file. It is also possible to drag a file into the editor window directly from Windows Explorer.  
  
When you select the settings window opens, in which the basic settings are available online.



In the first group, you can set the grid spacing , which "sticks" frame creation and selection widget that allows you to move the widgets and change their size is not fit to pixel by pixel , which is very easy to make a mistake . Also, there is a check mark to activate the auto-hidden panels, while the mode panel go beyond the edge of the screen , leaving a small corner of the panel, when you hover over the panel which leaves all the way back and become available .  
In the second group display settings .  
In the third group are configured the desired properties of the widget that you want to show in the hierarchy. In the hierarchy itself , which is available in the second section of the menu "Widgets" . (more on this below)  
  
  
Section menu "Widgets"



This section shows the hierarchy of widgets created, clicking on the widget menu, the corresponding widget released in the form. This organization is more intuitive and easy to navigate as well as allows you to learn the basic parameters of the widget, such as type, name, and skin. (optional, configurable in settings)

**Solution (2)**

Solyushen this panel, which displays the draft forms. Panel is optional, it appears, if you drag a file into the editor of a solution. As well it can always be close. File of a solution generated utilities or manually. Is a meta-description forms which are expected in the project, in the form prescribed types of widgets and their names. Just in solyushenami can optionally describe available skins and skins to create a default.  
  
Example, the following meta-description:

<MetaForm desc="GUI.Common.ToolTips.ContainerItemParts.QuestItem" layout="ttp\_questItem.layout">

<MetaWidget name="\_Main" type="Widget">

<MetaWidget name="Name" type="Widget" />

<MetaWidget name="Quest" type="Widget" />

<MetaWidget name="Desc" type="Widget" />

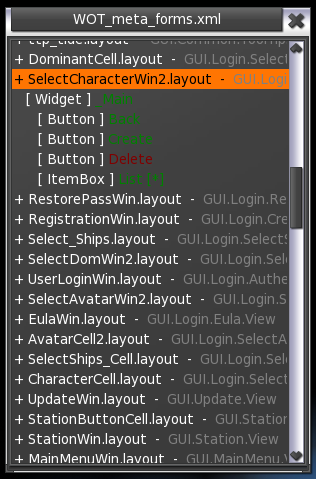
</MetaWidget>

</MetaForm>

Will be as follows:



When using a window of a solution should be understood that it is a certain clue that says that you need to create the type of widget should have and what the name, everything else depends on the developer forms. Consider the following example:

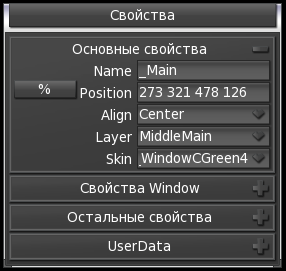


When you double-click on the name of the form , it unfolds , and we are presented a list of all the necessary widgets to this form. It is also seen that in this example, there is a certain hierarchy . ie Requires widget named «\_Main» and already will be attended by all the other widgets. But it does not impose a limit on the additional widgets, ie for example , the button «Back» can be on any widget or widgets that will create user ( decorative panels or pictures) . The list indicates the type of widget that is expected and the name. The name can have different colors , depending on whether it is created in the moment or not , for example red flagged widget that has not been created (in the example «Delete») and the green one is already present.

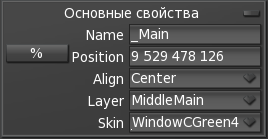
 When you single-click on the name of the widget, it will be released as if created by double , is created if it is not already created. When you create a default skin used and the default size is created on the current selection in the widget.

After the name of the widget can be a sign of [\*] (In the example «List [\*]») , this means that the widget is not final , and it is located on another form . For example ItemBox may contain cells that are different form . When you double-click on this widget to open the form , which is a subsidiary to this widget. But for the user this does not complicate the task , for additional action is required , still need to create a widget with type and name.

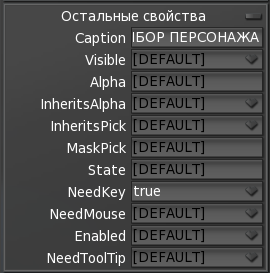
**Property (3)**



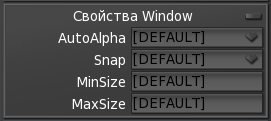
Properties window appears only if you select any widget , as well , the panel supports auto-hidden . In the panel is acceptable to change the size for more convenient location . In this panel available properties that apply to the current widget , as well as additional data. The panel is a context , ie it may be missing some section or sections within certain parameters .  
Just note that the input data if they are logically flawed , I change the color to red (or another depending on the topic ) . Some properties have a certain set of values ​​, and have the opportunity to select the desired . Some properties require direct user input.  
Section , you can manually turn off , for this you need to click on the button in the form minus the right of the section after this section will be curtailed , and the button will take the form of a plus for the disclosure section must click on the "plus" .  
  
  
basic properties



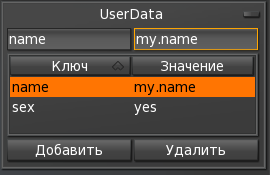
Section, which contains properties that apply to any widget - name, position, alignment and skin. Property «Layer» is not mandatory, and is usually present at the main widget form. This property determines which logical layer, you must attach a widget. Property «Position» has an additional switch, which allows you to specify the position and size in pixels, not only, but also as a percentage of the size of his father.  
  
  
Other properties



Section, in which there are properties that are present in all widgets. Of the main title, visibility, transparency, prompt, support mouse and keyboard.  
  
  
Properties of a specific type

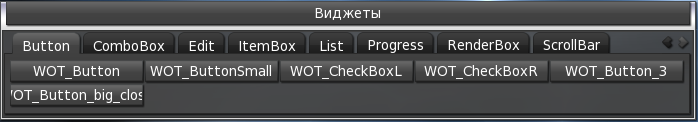


This section adapts to the specific type of widget (ex Window), and only displays its properties. Section may be omitted if no properties.  
  
  
user data

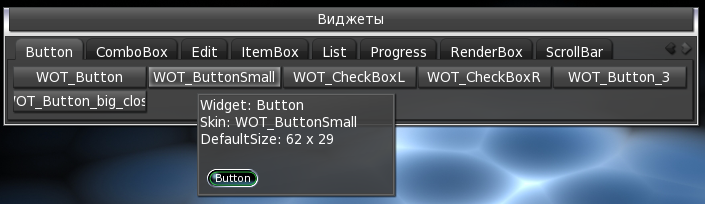


This section allows you to associate arbitrary data with a widget and the code to access them. Access to the value by a key.

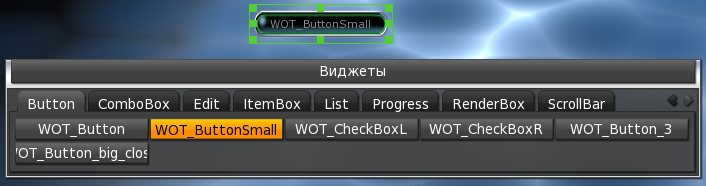
**Widgets (4)**



Window widget is a tab that contains all the available skins that you can use . Grouping by tabs is arbitrary , and is accessible via xml configuration, but with a large number of skins, convenient to each tab was a type , and it contained the skins to be used for this type. When you hover over the button skin , you will be prompted , in which it is possible to see how the widget will look like without its creation. Just a tip size is specified for which the created widget , usually is the recommended size , if the widget is not planned to stretch .



To create a widget, you need to choose the skin by clicking on its button, then the button is pressed. Next, you need the right place, or form window, click the left button and hold it, stretch the widget to the desired size. In the future, the size can be changed via the properties panel or directly dragging a selection box widget. To remove a widget, you can press «Delete» key.



additional information  
  
Select and move  
  
Currently selected widget framed special frame, which allows you to change its size. Just highlighted the widget can be moved by clicking and dragging it in any place.



To change the position or size without reference to the grid, you need to press and hold the «Shift» with the size or position will be changed per pixel.