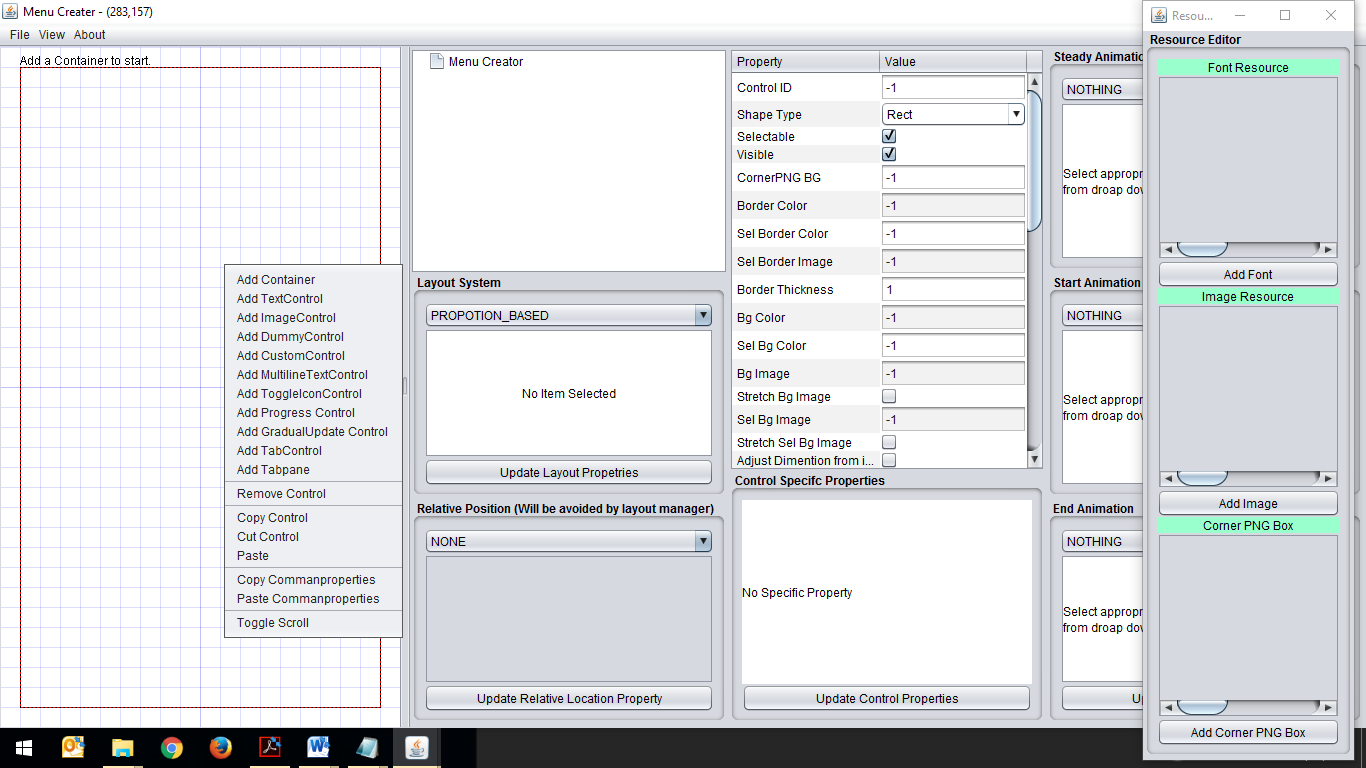
MenuCreator is a desktop tool which can be used to create different UI layouts and screen by drag and drop functionality.

1. Right click left hand side editor to add any control. Below is the list of the controls supported by the Menu Creator.



1. Containers behave in the same way as JPanel behaves in java, where layout can be applied and more control can be added.
2. Finally once all UI is created export the UI layout using File -> Export
3. For getting more detailed information about using MenuCreator please email at: [swaroop@appon.co.in](mailto:swaroop@appon.co.in)

**Technical integration of MenuCreator with game is explained below:**

1. First load the .menuex file stored in assets/all folder to use.

private ScrollableContainer parentContainer

// ResourceManager class is used to manage font & images

ResourceManager.getInstance().setFontResource(0, MyGameConstants.Font);

ResourceManager.getInstance().setImageResource(0,MyGameConstants.BUTTON\_BG.getImage())

ResourceManager.getInstance().setImageResource(1,MyGameConstants.PLAY.getImage());

ResourceManager.getInstance().setImageResource(2,MyGameConstants.EXIT.getImage());

parentContainer=com.appon.miniframework.Util.loadContainer(GTantra.getFileByteDat("/mainmenu.menuex",GameActivity.getInstance))

,masterWidth,masterHeight,deviceWidth,deviceHeight,isTouchDevice);

2. Button press events are handled as follows:

parentContainer.setEventManager(new EventManager() {

public void event(Event event) {

if(event.getEventId() == FIRE\_PRESSED || event.getEventId() == CONTROL\_CLICKED){

switch (event.getSource().getId()) {

// 1 & 2 button ids

case 1:

// button play

break;

case 2:

// button close

break;

}

}

}

});

3. To paint menu call method mentioned below:

parentContainer.paintUI(c, paintObject);

4. Also call pointer press, release and drag method to create events.

parentContainer.pointerPressed(x, y);

parentContainer.pointerReleased(x, y);

parentContainer.pointerDragged(x, y);

5. We can get controls used inside the menu and programatically change properties.

E.g.-

childControl=com.appon.miniframework.Util.findControl(parentContainer, id);

childControl.setVisible(true);

and many more properties.

6. After changing properties we have to call the method below to apply changes to the menu.

com.appon.miniframework.Util.reallignContainer(parentContainer);

--------For more information checkout the MainMenu Class of Template