

iRidium GUI Editor

the application for creating control interfaces for automation systems, configuring drivers and equipment used in automation projects and writing custom drivers with the help of [iRidium DDK](#). It is responsible for setting up connection between graphic and driver parts of iRidium projects (iRidium Project).

Main Terms and Definitions

- **iRidium Project** – a project file created in [iRidium GUI Editor](#). It consists of a graphic part (GUI) and a driver part (commands for controlling equipment). iRidium Projects set up in GUI Editor have *.IRP or *.IRPZ formats. Projects consisting of the interface and driver parts is processed by the [iRidium Transfer](#) application and uploaded on control panels with installed [iRidium Apps](#).
- **Graphic interface** – the complex of items of iRidium project graphic part (project pages and popups, graphic items).
- **Page /popup** – objects which define the interface structure. They are meant for placing graphic items. You can open only one project page at a time. It is a project base over which you can place any number of popups of different size.
- **Graphic item** – an object on a project page or popup (button, level, animated item, inertial list, etc.)
- **Control panel** – an iOS, Windows, Android, OS X based device with the installed iRidium App where it is required to launch iRidium project.
- **License** (license file) – an *.irl format file which enables connection between a panel and controlled equipment. The license is uploaded onto the panel together with the iRidium project.
- **Activation key** – a set of 160 symbols which serves as an iRidium license activation code. It contains information about the acquired license and enables you to generate a license file.
- **Licensing#Activation of iRidium Licenses[Process of key activation]]** – creation of a license file with the help of the key in the MY LICENSES section of the web site.
- **Control panel identifier (HWID/UDID)** – the unique identifier of the Apple iOS (UDID) or Windows (HWID) controlled device, which is required for iRidium key activation and license file receipt.
- **Emulator** – a tool for simulation of the project launch on the control panel. It enables you to see how the project works and check communication with the equipment.
- **Driver** – a description of a protocol for equipment operation created in iRidium. It allows iRidium projects to refer to the equipment of the corresponding type and receive data from it. iRidium supports a number of native (written preliminarily and stored in iRidium Apps) drivers. There is also a possibility to create custom drivers on the basis of [AV & Custom Systems](#) with the help of [iRidium DDK](#) – it enables profound configuration of the driver with the help of the script system.
- **iRidium DDK** – a tool for creating drivers. It includes iRidium Script machine, [iRidium Script API](#), [iRidium DDK](#) for writing drivers and examples of drivers.
- **Controlled device** – equipment (a controller, transmitter, media server, AV device, etc.) which uses one of the drivers described either in native or script part of iRidium project.

- **Command** – an instruction for sending data to controlled equipment. The command sending is activated by pressing on a graphic item, system events or scripts. It contains information about the data type and settings specific for particular equipment. The list of instructions (commands) is stored in the Project Device Panel tree.
- **Status channels (Feedback)** – an instruction for processing data received from the controlled equipment. Received data can affect GUI items (value output on the screen, switching item states). Status channels are stored in the Project Device Panel tree.
- **[[Project Token** – a variable for saving data received from control equipment or interface. Project Tokens serve for writing, storing data or transferring them inside the project. The data can be numerical (in the DEC format) or string (in the ASCII format). After the restart of [iRidium Apps](#) or control panels information written in Project tokens stays available (for example you can write a login and password for getting access to the equipment, connection properties – IP-address, device password, etc).
- **System Tokens** – variables storing data about the control panel state (time, date, data from sensors, etc.). System tokens cannot be affected via interfaces, they can be read only.
- **Driver Tokens** – variables storing data about the state of connection to project drivers and connection properties. Driver tokens cannot be affected via interfaces, they can be read only.

Contents

- [1 How to Work with iRidium GUI Editor](#)
 - [1.1 Installation of the iRidium Software Package](#)
 - [1.1.1 Where to Download from](#)
 - [1.1.2 Requirements to PC for Working with iRidium](#)
 - [1.2 Description of GUI Editor](#)
 - [1.3 Creating New Projects](#)
 - [1.4 Creation of the Graphic Part of iRidium Projects](#)
 - [1.5 Setting up Communication with Equipment](#)
 - [1.6 Emulation of Project Work](#)
 - [1.7 Uploading Projects on Control Panels](#)

How to Work with iRidium GUI Editor

Approximate sequence for actions for creating iRidium projects for controlling equipment of automation systems consists of the following steps:

1. Download, install and set up iRidium Environment for PC
2. Start iRidium GUI Editor - it has a number of working areas and tools
3. Create and set up a new project (properties for the project launch and work are set up on control panels)
4. Set up the graphic part of the project: graphic items, pages, popups
5. Use special iRidium capabilities to create a unique interface
6. Launch the ready project in Emulator to make sure it works correctly
7. Upload the project on the target control panel

Installation of the iRidium Software Package

Work on project creating starts with installing iRidium Environment for PC (Windows OS). This part of the software package is responsible for creating iRidium projects including graphic interfaces and instructions for communication with equipment. iRidium Environment for PC is also used for uploading projects on control panels with the installed [iRidium App](#).

Where to Download from

You can download iRidium Environment to your PC from iridiummobile.net, the "Downloads" section.

- Install components [iRidium GUI Editor](#), [iRidium Transfer](#) and [iRidium App](#) for Windows included in iRidium Environment:

[download the last version of iRidium Environment >>](#)

Requirements to PC for Working with iRidium

- Microsoft Windows XP/7/8 OS
- Not less than 2 Gb RAM
- Not less than 1 Gb of free space on HDD
- 3D-video adapter with memory not less than 512 Mb

Important! It is required to start iRidium components on PC under Administrator account or using Administrator rights for correct work.

Important! The correct work of iRidium components is not guaranteed when working on **Windows OS Emulator** (for example, WineHQ). iRidium licenses might not work due to changing of virtual machine HWID .

Important! When hard disks of your PC work in the **RAID array**, every time it generates new HWID. Do not activate licenses for your PC with the RAID array, this mode should be turned off.

Important! If video card drivers do not support **OpenGL** (for Windows XP) and **DirectX 10** (for Windows 7/8) there might be errors when displaying graphics. Update your video card drivers if the graphics is displayed incorrectly.

Description of GUI Editor

When launching Editor you will see a number of working areas where you can set up different properties for working with Editor. Size and position of working areas are regulated. Displaying of object properties and properties for working with projects are set in Editor.

To receive full information about iRidium GUI Editor components, tools and settings see the following sections:

- [Tool Box and the GUI Editor Interface](#) (description of Editor working area, tabs, tools)
- [Settings of GUI Editor Workspace](#) (description of properties which can be displayed on items in Workspace)
- [Settings of the GUI Editor Tools](#) (settings for importing graphics to Gallery, Emulation properties, Workspace fonts, settings of the Project Overview tree of pages, checking updates at the project launch, project auto saving)
- [Settings for Launching Projects in GUI Editor](#) (project displaying and launch).

Creating New Projects

iRidium projects consist of graphic and driver parts. when creating a new project you are required to indicate properties of the graphic part which include project resolution, its orientation on control panels and properties for launching on different types of control panels:

- [New iRidium project](#)

Creation of the Graphic Part of iRidium Projects

Creation of the graphic part of iRidium projects includes the following steps:

- [creating graphic items for controlling equipment](#)
- [creating Project Gallery of graphics and sounds](#)
- [working with images](#)
- [creating animation effects](#)
- [forming project pages and popups](#)
- [creating macros for equipment and interface control](#)
- [optimizing the graphic part of the project](#)

You can use a number of specific visualization capabilities for creating unique and user-friendly graphic interfaces in iRidium GUI Editor:

- Animated items
- Edit box
- Virtual keyboards
- Inertial lists
- Ready graphic and sound galleries
- Chameleon images

- Visualization effects of pressings
- Effects for page switching
- Passwords on actions
- Video from IP cameras
- Graphic informers
- Hyperlinks, OS commands
- Unicode support
- Emulation of projects work
- Scripts support

[↑ Back](#)

Setting up Communication with Equipment

Use instructions for working with separate types of drivers to set up communication with the equipment. You can find the list of instructions for iRidium supported drivers on Main Page and the side panel of iRidium Wiki.

You can find general information about working with equipment in iRidium GUI Editor [here >>](#)

[↑ Back](#)

Emulation of Project Work

When working with the graphic and driver parts of iRidium projects it is sometimes necessary to check their work in actual practice - behavior of the graphic part and connection to equipment.

iRidium projects can be launched in Editor with the help of Emulator which works similarly to iRidium App for Windows. On different OS the project appearance does not change as the iRidium application is multiplatform (works similarly on all OS). You can check general work with interfaces and connection to equipment in the Emulation mode.

Description of settings and work with [Emulator >>](#)

[↑ Back](#)

Uploading Projects on Control Panels

To upload created projects on control panels use the [iRidium Transfer](#) application included in iRidium Environment. Transfer stores interfaces created in GUI Editor, licenses and information about panels the interfaces should be uploaded on. You can send your project to your control panel directly from GUI Editor with the help of the "**Send to Panel**" button or you can add the saved project to Transfer.

[Uploading Projects on Control Panels >>](#)

[↑ Back](#)