**The Graphical User Interface (GUI)**

We are using *GTK+* in C programming language to generate the Graphical User Interface in order to view the different data and also control different settings.

The *GTK+* is a library for creating graphical user interfaces. The library is created in C programming language. The GTK+ library is also called the GIMP toolkit. Originally, the library was created while developing the GIMP image manipulation program. Since then, the GTK+ became one of the most popular toolkits under Linux and BSD Unix. Today, most of the GUI software in the open source world is created in Qt or in GTK+. The GTK+ is an object oriented application programming interface. The object oriented system is created with the Glib object system, which is a base for the GTK+ library. The *GObject* also enables to create language bindings for various other programming languages. Language bindings exist for C++, Python, Perl, Java, C#, and other programming languages.

The GTK+ itself depends on the following libraries.

* ***Glib***

GLib provides the core application building blocks for libraries and applications written in C. It provides the core object system used in GNOME, the main loop implementation, and a large set of utility functions for strings and common data structures.

***Description***

GLib defines a number of commonly used types, which can be divided into 4 groups:

* New types which are not part of standard C (but are defined in various C standard library header files) - [gboolean](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html" \l "gboolean" \o "gboolean), [gsize](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html" \l "gsize" \o "gsize), [gssize](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html" \l "gssize" \o "gssize),[goffset](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html#goffset), [gintptr](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html" \l "gintptr" \o "gintptr), [guintptr](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html" \l "guintptr" \o "guintptr).
* Integer types which are guaranteed to be the same size across all platforms - [gint8](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html#gint8), [guint8](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html#guint8), [gint16](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html#gint16), [guint16](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html#guint16), [gint32](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html#gint32), [guint32](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html#guint32),[gint64](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html#gint64), [guint64](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html#guint64).
* Types which are easier to use than their standard C counterparts - [gpointer](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html" \l "gpointer" \o "gpointer), [gconstpointer](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html" \l "gconstpointer" \o "gconstpointer), [guchar](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html" \l "guchar" \o "guchar), [guint](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html" \l "guint" \o "guint), [gushort](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html" \l "gushort" \o "gushort), [gulong](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html" \l "gulong" \o "gulong).
* Types which correspond exactly to standard C types, but are included for completeness - [gchar](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html" \l "gchar" \o "gchar), [gint](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html" \l "gint" \o "gint), [gshort](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html" \l "gshort" \o "gshort), [glong](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html" \l "glong" \o "glong), [gfloat](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html" \l "gfloat" \o "gfloat),[gdouble](https://developer.gnome.org/glib/2.44/glib-Basic-Types.html#gdouble).

GLib also defines macros for the limits of some of the standard integer and floating point types, as well as macros for suitableprintf() formats for these types.

[**Standard Macros**](https://developer.gnome.org/glib/2.44/glib-Standard-Macros.html) — commonly-used macros

[**Type Conversion Macros**](https://developer.gnome.org/glib/2.44/glib-Type-Conversion-Macros.html) — portably storing integers in pointer variables

[**Byte Order Macros**](https://developer.gnome.org/glib/2.44/glib-Byte-Order-Macros.html) — a portable way to convert between different byte orders

[**Numerical Definitions**](https://developer.gnome.org/glib/2.44/glib-Numerical-Definitions.html) — mathematical constants, and floating point decomposition

[**Miscellaneous Macros**](https://developer.gnome.org/glib/2.44/glib-Miscellaneous-Macros.html) — specialized macros which are not used often

[**Atomic Operations**](https://developer.gnome.org/glib/2.44/glib-Atomic-Operations.html) — basic atomic integer and pointer operations

[**GLib Core Application Support**](https://developer.gnome.org/glib/2.44/glib-core.html)

[**The Main Event Loop**](https://developer.gnome.org/glib/2.44/glib-The-Main-Event-Loop.html) — manages all available sources of events

[**Threads**](https://developer.gnome.org/glib/2.44/glib-Threads.html) — portable support for threads, mutexes, locks, conditions and thread private data

[**Thread Pools**](https://developer.gnome.org/glib/2.44/glib-Thread-Pools.html) — pools of threads to execute work concurrently

[**Asynchronous Queues**](https://developer.gnome.org/glib/2.44/glib-Asynchronous-Queues.html) — asynchronous communication between threads

[**Dynamic Loading of Modules**](https://developer.gnome.org/glib/2.44/glib-Dynamic-Loading-of-Modules.html) — portable method for dynamically loading 'plug-ins'

[**Memory Allocation**](https://developer.gnome.org/glib/2.44/glib-Memory-Allocation.html) — general memory-handling

[**Memory Slices**](https://developer.gnome.org/glib/2.44/glib-Memory-Slices.html) — efficient way to allocate groups of equal-sized chunks of memory

[**IO Channels**](https://developer.gnome.org/glib/2.44/glib-IO-Channels.html) — portable support for using files, pipes and sockets

[**Error Reporting**](https://developer.gnome.org/glib/2.44/glib-Error-Reporting.html) — a system for reporting errors

[**Message Output and Debugging Functions**](https://developer.gnome.org/glib/2.44/glib-Warnings-and-Assertions.html) — functions to output messages and help debug applications

[**Message Logging**](https://developer.gnome.org/glib/2.44/glib-Message-Logging.html) — versatile support for logging messages with different levels of importance

[**GLib Utilities**](https://developer.gnome.org/glib/2.44/glib-utilities.html)

[**String Utility Functions**](https://developer.gnome.org/glib/2.44/glib-String-Utility-Functions.html) — various string-related functions

[**Character Set Conversion**](https://developer.gnome.org/glib/2.44/glib-Character-Set-Conversion.html) — convert strings between different character sets

[**Unicode Manipulation**](https://developer.gnome.org/glib/2.44/glib-Unicode-Manipulation.html) — functions operating on Unicode characters and UTF-8 strings

[**Base64 Encoding**](https://developer.gnome.org/glib/2.44/glib-Base64-Encoding.html) — encodes and decodes data in Base64 format

[**Data Checksums**](https://developer.gnome.org/glib/2.44/glib-Data-Checksums.html) — computes the checksum for data

[**Secure HMAC Digests**](https://developer.gnome.org/glib/2.44/glib-Data-HMACs.html) — computes the HMAC for data

[**Internationalization**](https://developer.gnome.org/glib/2.44/glib-I18N.html) — gettext support macros

[**Date and Time Functions**](https://developer.gnome.org/glib/2.44/glib-Date-and-Time-Functions.html) — calendrical calculations and miscellaneous time stuff

[**GTimeZone**](https://developer.gnome.org/glib/2.44/glib-GTimeZone.html) — a structure representing a time zone

[**GDateTime**](https://developer.gnome.org/glib/2.44/glib-GDateTime.html) — a structure representing Date and Time

[**Random Numbers**](https://developer.gnome.org/glib/2.44/glib-Random-Numbers.html) — pseudo-random number generator

[**Hook Functions**](https://developer.gnome.org/glib/2.44/glib-Hook-Functions.html) — support for manipulating lists of hook functions

[**Miscellaneous Utility Functions**](https://developer.gnome.org/glib/2.44/glib-Miscellaneous-Utility-Functions.html) — a selection of portable utility functions

[**Lexical Scanner**](https://developer.gnome.org/glib/2.44/glib-Lexical-Scanner.html) — a general purpose lexical scanner

[**Timers**](https://developer.gnome.org/glib/2.44/glib-Timers.html) — keep track of elapsed time

[**Spawning Processes**](https://developer.gnome.org/glib/2.44/glib-Spawning-Processes.html) — process launching

[**File Utilities**](https://developer.gnome.org/glib/2.44/glib-File-Utilities.html) — various file-related functions

[**URI Functions**](https://developer.gnome.org/glib/2.44/glib-URI-Functions.html) — manipulating URIs

[**Hostname Utilities**](https://developer.gnome.org/glib/2.44/glib-Hostname-Utilities.html) — Internet hostname utilities

[**Shell-related Utilities**](https://developer.gnome.org/glib/2.44/glib-Shell-related-Utilities.html) — shell-like commandline handling

[**Commandline option parser**](https://developer.gnome.org/glib/2.44/glib-Commandline-option-parser.html) — parses commandline options

[**Glob-style pattern matching**](https://developer.gnome.org/glib/2.44/glib-Glob-style-pattern-matching.html) — matches strings against patterns containing '\*' (wildcard) and '?' (joker)

[**Perl-compatible regular expressions**](https://developer.gnome.org/glib/2.44/glib-Perl-compatible-regular-expressions.html) — matches strings against regular expressions

[**Regular expression syntax**](https://developer.gnome.org/glib/2.44/glib-regex-syntax.html) — syntax and semantics of regular expressions supported by GRegex

[**Simple XML Subset Parser**](https://developer.gnome.org/glib/2.44/glib-Simple-XML-Subset-Parser.html) — parses a subset of XML

[**Key-value file parser**](https://developer.gnome.org/glib/2.44/glib-Key-value-file-parser.html) — parses .ini-like config files

[**Bookmark file parser**](https://developer.gnome.org/glib/2.44/glib-Bookmark-file-parser.html) — parses files containing bookmarks

[**Testing**](https://developer.gnome.org/glib/2.44/glib-Testing.html) — a test framework

[**UNIX-specific utilities and integration**](https://developer.gnome.org/glib/2.44/glib-UNIX-specific-utilities-and-integration.html) — pipes, signal handling

[**Windows Compatibility Functions**](https://developer.gnome.org/glib/2.44/glib-Windows-Compatibility-Functions.html) — UNIX emulation on Windows

[**GLib Data Types**](https://developer.gnome.org/glib/2.44/glib-data-types.html)

[**Doubly-Linked Lists**](https://developer.gnome.org/glib/2.44/glib-Doubly-Linked-Lists.html) — linked lists that can be iterated over in both directions

[**Singly-Linked Lists**](https://developer.gnome.org/glib/2.44/glib-Singly-Linked-Lists.html) — linked lists that can be iterated in one direction

[**Double-ended Queues**](https://developer.gnome.org/glib/2.44/glib-Double-ended-Queues.html) — double-ended queue data structure

[**Sequences**](https://developer.gnome.org/glib/2.44/glib-Sequences.html) — scalable lists

[**Trash Stacks**](https://developer.gnome.org/glib/2.44/glib-Trash-Stacks.html) — maintain a stack of unused allocated memory chunks

[**Hash Tables**](https://developer.gnome.org/glib/2.44/glib-Hash-Tables.html) — associations between keys and values so that given a key the value can be found quickly

[**Strings**](https://developer.gnome.org/glib/2.44/glib-Strings.html) — text buffers which grow automatically as text is added

[**String Chunks**](https://developer.gnome.org/glib/2.44/glib-String-Chunks.html) — efficient storage of groups of strings

[**Arrays**](https://developer.gnome.org/glib/2.44/glib-Arrays.html) — arrays of arbitrary elements which grow automatically as elements are added

[**Pointer Arrays**](https://developer.gnome.org/glib/2.44/glib-Pointer-Arrays.html) — arrays of pointers to any type of data, which grow automatically as new elements are added

[**Byte Arrays**](https://developer.gnome.org/glib/2.44/glib-Byte-Arrays.html) — arrays of bytes

[**Balanced Binary Trees**](https://developer.gnome.org/glib/2.44/glib-Balanced-Binary-Trees.html) — a sorted collection of key/value pairs optimized for searching and traversing in order

[**N-ary Trees**](https://developer.gnome.org/glib/2.44/glib-N-ary-Trees.html) — trees of data with any number of branches

[**Quarks**](https://developer.gnome.org/glib/2.44/glib-Quarks.html) — a 2-way association between a string and a unique integer identifier

[**Keyed Data Lists**](https://developer.gnome.org/glib/2.44/glib-Keyed-Data-Lists.html) — lists of data elements which are accessible by a string or GQuark identifier

[**Datasets**](https://developer.gnome.org/glib/2.44/glib-Datasets.html) — associate groups of data elements with particular memory locations

[**GVariantType**](https://developer.gnome.org/glib/2.44/glib-GVariantType.html) — introduction to the GVariant type system

[**GVariant**](https://developer.gnome.org/glib/2.44/glib-GVariant.html) — strongly typed value datatype

[**GVariant Format Strings**](https://developer.gnome.org/glib/2.44/gvariant-format-strings.html) — varargs conversion of GVariants

[**GVariant Text Format**](https://developer.gnome.org/glib/2.44/gvariant-text.html) — textual representation of GVariants

[**Deprecated APIs**](https://developer.gnome.org/glib/2.44/deprecated.html)

[**Deprecated thread API**](https://developer.gnome.org/glib/2.44/glib-Deprecated-Thread-APIs.html) — old thread APIs (for reference only)

[**Caches**](https://developer.gnome.org/glib/2.44/glib-Caches.html) — caches allow sharing of complex data structures to save resources

[**Relations and Tuples**](https://developer.gnome.org/glib/2.44/glib-Relations-and-Tuples.html) — tables of data which can be indexed on any number of fields

[**Automatic String Completion**](https://developer.gnome.org/glib/2.44/glib-Automatic-String-Completion.html) — support for automatic completion using a group of target strings

[**GLib Tools**](https://developer.gnome.org/glib/2.44/tools.html)

[**glib-gettextize**](https://developer.gnome.org/glib/2.44/glib-gettextize.html) — gettext internationalization utility

[**gtester**](https://developer.gnome.org/glib/2.44/gtester.html) — test running utility

[**gtester-report**](https://developer.gnome.org/glib/2.44/gtester-report.html) — test report formatting utility

* ***Pango***

Pango is a library for laying out and rendering of text, with an emphasis on internationalization. Pango can be used anywhere that text layout is needed, though most of the work on Pango so far has been done in the context of the GTK+ widget toolkit. Pango forms the core of text and font handling for GTK+-2.x.

Pango is designed to be modular; the core Pango layout engine can be used with different font backends. There are three basic backends, with multiple options for rendering with each.

* Client side fonts using the FreeType and fontconfig libraries, using [HarfBuzz](http://www.pango.org/HarfBuzz) for complex-text handling. Rendering can be with with Cairo or Xft libraries, or directly to an in-memory buffer with no additional libraries.
* Native fonts on Microsoft Windows using Uniscribe for complex-text handling. Rendering can be done via Cairo or directly using the native Win32 API.
* Native fonts on MacOS X using [CoreText](http://www.pango.org/CoreText) for complex-text handling, rendering via Cairo.

The integration of Pango with Cairo (<http://cairographics.org/>) provides a complete solution with high quality text handling and graphics rendering.

Dynamically loaded modules then handle text layout for particular combinations of script and font backend. Pango ships with a wide selection of modules, including modules for Hebrew, Arabic, Hangul, Thai, and a number of Indic scripts. Virtually all of the world's major scripts are supported.

As well as the low level layout rendering routines, Pango includes PangoLayout, a high level driver for laying out entire blocks of text, and routines to assist in editing internationalized text.

Pango depends on 2.x series of the GLib library; more information about GLib can be found at <http://www.gtk.org/>.

* ***ATK***

ATK provides the set of accessibility interfaces that are implemented by other toolkits and applications. Using the ATK interfaces, accessibility tools have full access to view and control running applications.

[**Base accessibility object**](https://developer.gnome.org/atk/2.16/atkobject.html)

[**AtkObject**](https://developer.gnome.org/atk/2.16/AtkObject.html) — The base object class for the Accessibility Toolkit API.

[**Event and toolkit support**](https://developer.gnome.org/atk/2.16/toolkit.html)

[**AtkUtil**](https://developer.gnome.org/atk/2.16/AtkUtil.html) — A set of ATK utility functions for event and toolkit support.

[**ATK Interfaces**](https://developer.gnome.org/atk/2.16/interfaces.html)

[**AtkAction**](https://developer.gnome.org/atk/2.16/AtkAction.html) — The ATK interface provided by UI components which the user can activate/interact with.

[**AtkComponent**](https://developer.gnome.org/atk/2.16/AtkComponent.html) — The ATK interface provided by UI components which occupy a physical area on the screen. which the user can activate/interact with.

[**AtkDocument**](https://developer.gnome.org/atk/2.16/AtkDocument.html) — The ATK interface which represents the toplevel container for document content.

[**AtkEditableText**](https://developer.gnome.org/atk/2.16/AtkEditableText.html) — The ATK interface implemented by components containing user-editable text content.

[**AtkHyperlinImpl**](https://developer.gnome.org/atk/2.16/atk-AtkHyperlinkImpl.html) — An interface from which the AtkHyperlink associated with an AtkObject may be obtained.

[**AtkHypertext**](https://developer.gnome.org/atk/2.16/AtkHypertext.html) — The ATK interface which provides standard mechanism for manipulating hyperlinks.

[**AtkImage**](https://developer.gnome.org/atk/2.16/AtkImage.html) — The ATK Interface implemented by components which expose image or pixmap content on-screen.

[**AtkSelection**](https://developer.gnome.org/atk/2.16/AtkSelection.html)The ATK interface implemented by container objects whose AtkObject children can be selected.

[**AtkStreamableContent**](https://developer.gnome.org/atk/2.16/AtkStreamableContent.html) — The ATK interface which provides access to streamable content.

[**AtkTable**](https://developer.gnome.org/atk/2.16/AtkTable.html) — The ATK interface implemented for UI components which contain tabular or row/column information.

[**AtkTableCell**](https://developer.gnome.org/atk/2.16/AtkTableCell.html)The ATK interface implemented for a cell inside a two-dimentional AtkTable

[**AtkText**](https://developer.gnome.org/atk/2.16/AtkText.html) — The ATK interface implemented by components with text content.

[**AtkValue**](https://developer.gnome.org/atk/2.16/AtkValue.html) — The ATK interface implemented by valuators and components which display or select a value from a bounded range of values.

[**AtkWindow**](https://developer.gnome.org/atk/2.16/AtkWindow.html) — The ATK Interface provided by UI components that represent a top-level window.

[**Basic accessible data types**](https://developer.gnome.org/atk/2.16/data.html)

[**AtkRange**](https://developer.gnome.org/atk/2.16/atk-AtkRange.html)A given range or subrange, to be used with AtkValue

[**AtkRelation**](https://developer.gnome.org/atk/2.16/AtkRelation.html) — An object used to describe a relation between a object and one or more other objects.

[**AtkRelationSet**](https://developer.gnome.org/atk/2.16/AtkRelationSet.html) — A set of AtkRelations, normally the set of AtkRelations which an AtkObject has.

[**AtkState**](https://developer.gnome.org/atk/2.16/atk-AtkState.html) — An AtkState describes a single state of an object.

[**AtkStateSet**](https://developer.gnome.org/atk/2.16/AtkStateSet.html) — An AtkStateSet contains the states of an object.

[**Custom accessible objects**](https://developer.gnome.org/atk/2.16/accessibles.html)

[**AtkGObjectAccessible**](https://developer.gnome.org/atk/2.16/AtkGObjectAccessible.html) — This object class is derived from AtkObject and can be used as a basis implementing accessible objects.

[**AtkHyperlink**](https://developer.gnome.org/atk/2.16/AtkHyperlink.html) — An ATK object which encapsulates a link or set of links in a hypertext document.

[**AtkNoOpObject**](https://developer.gnome.org/atk/2.16/AtkNoOpObject.html) — An AtkObject which purports to implement all ATK interfaces.

[**AtkPlug**](https://developer.gnome.org/atk/2.16/atk-AtkPlug.html) — Toplevel for embedding into other processes

[**AtkSocket**](https://developer.gnome.org/atk/2.16/atk-AtkSocket.html) — Container for AtkPlug objects from other processes

[**Utilities**](https://developer.gnome.org/atk/2.16/utilities.html)

[**AtkNoOpObjectFactory**](https://developer.gnome.org/atk/2.16/AtkNoOpObjectFactory.html) — The AtkObjectFactory which creates an AtkNoOpObject.

[**AtkObjectFactory**](https://developer.gnome.org/atk/2.16/AtkObjectFactory.html) — The base object class for a factory used to create accessible objects for objects of a specific GType.

[**AtkRegistry**](https://developer.gnome.org/atk/2.16/AtkRegistry.html) — An object used to store the GType of the factories used to create an accessible object for an object of a particular GType.

[**Versioning macros**](https://developer.gnome.org/atk/2.16/atk-Versioning-Utilities.html) — Variables and functions to check the ATK version

[**Deprecated Interfaces**](https://developer.gnome.org/atk/2.16/deprecated.html)

[**AtkMisc**](https://developer.gnome.org/atk/2.16/AtkMisc.html) — A set of ATK utility functions for thread locking

* ***GDK***

[**I. API Reference**](https://developer.gnome.org/gdk3/stable/reference.html)

[**General**](https://developer.gnome.org/gdk3/stable/gdk3-General.html) — Library initialization and miscellaneous functions

[**GdkDisplayManager**](https://developer.gnome.org/gdk3/stable/GdkDisplayManager.html) — Maintains a list of all open GdkDisplays

[**GdkDisplay**](https://developer.gnome.org/gdk3/stable/GdkDisplay.html) — Controls a set of GdkScreens and their associated input devices

[**GdkScreen**](https://developer.gnome.org/gdk3/stable/GdkScreen.html) — Object representing a physical screen

[**GdkDeviceManager**](https://developer.gnome.org/gdk3/stable/GdkDeviceManager.html) — Functions for handling input devices

[**GdkDevice**](https://developer.gnome.org/gdk3/stable/GdkDevice.html) — Object representing an input device

[**Points and Rectangles**](https://developer.gnome.org/gdk3/stable/gdk3-Points-Rectangles-and-Regions.html) — Simple graphical data types

[**Pixbufs**](https://developer.gnome.org/gdk3/stable/gdk3-Pixbufs.html) — Functions for obtaining pixbufs

[**RGBA Colors**](https://developer.gnome.org/gdk3/stable/gdk3-RGBA-Colors.html) — RGBA colors

[**Visuals**](https://developer.gnome.org/gdk3/stable/gdk3-Visuals.html) — Low-level display hardware information

[**Cursors**](https://developer.gnome.org/gdk3/stable/gdk3-Cursors.html) — Standard and pixmap cursors

[**Windows**](https://developer.gnome.org/gdk3/stable/gdk3-Windows.html) — Onscreen display areas in the target window system

[**Frame clock**](https://developer.gnome.org/gdk3/stable/GdkFrameClock.html) — Frame clock syncs painting to a window or display

[**Frame timings**](https://developer.gnome.org/gdk3/stable/gdk3-GdkFrameTimings.html) — Object holding timing information for a single frame

[**GdkGLContext**](https://developer.gnome.org/gdk3/stable/GdkGLContext.html) — OpenGL context

[**Events**](https://developer.gnome.org/gdk3/stable/gdk3-Events.html) — Functions for handling events from the window system

[**Event Structures**](https://developer.gnome.org/gdk3/stable/gdk3-Event-Structures.html) — Data structures specific to each type of event

[**Key Values**](https://developer.gnome.org/gdk3/stable/gdk3-Keyboard-Handling.html) — Functions for manipulating keyboard codes

[**Selections**](https://developer.gnome.org/gdk3/stable/gdk3-Selections.html) — Functions for transfering data via the X selection mechanism

[**Drag And Drop**](https://developer.gnome.org/gdk3/stable/gdk3-Drag-and-Drop.html) — Functions for controlling drag and drop handling

[**Properties and Atoms**](https://developer.gnome.org/gdk3/stable/gdk3-Properties-and-Atoms.html) — Functions to manipulate properties on windows

[**Threads**](https://developer.gnome.org/gdk3/stable/gdk3-Threads.html) — Functions for using GDK in multi-threaded programs

[**Pango Interaction**](https://developer.gnome.org/gdk3/stable/gdk3-Pango-Interaction.html) — Using Pango in GDK

[**Cairo Interaction**](https://developer.gnome.org/gdk3/stable/gdk3-Cairo-Interaction.html) — Functions to support using cairo

[**X Window System Interaction**](https://developer.gnome.org/gdk3/stable/gdk3-X-Window-System-Interaction.html) — X backend-specific functions

[**Wayland Interaction**](https://developer.gnome.org/gdk3/stable/gdk3-Wayland-Interaction.html) — Wayland backend-specific functions

[**Application launching**](https://developer.gnome.org/gdk3/stable/gdk3-Application-launching.html) — Startup notification for applications

[**Testing**](https://developer.gnome.org/gdk3/stable/gdk3-Testing.html) — Test utilities

[**II. Deprecated**](https://developer.gnome.org/gdk3/stable/rn02.html)

[**Colors**](https://developer.gnome.org/gdk3/stable/gdk3-Colors.html) — Manipulation of colors

* ***GdkPixbuf***

[**I. API Reference**](https://developer.gnome.org/gdk-pixbuf/stable/rn01.html)

[**Initialization and Versions**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-Versioning.html) — Library version numbers.

[**The GdkPixbuf Structure**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-The-GdkPixbuf-Structure.html) — Information that describes an image.

[**Reference Counting and Memory Mangement**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-Reference-Counting-and-Memory-Mangement.html) — Functions for reference counting and memory management on pixbufs.

[**File Loading**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-File-Loading.html) — Loading a pixbuf from a file.

[**File saving**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-File-saving.html) — Saving a pixbuf to a file.

[**Image Data in Memory**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-Image-Data-in-Memory.html) — Creating a pixbuf from image data that is already in memory.

[**Inline data**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-Inline-data.html) — Functions for inlined pixbuf handling.

[**Scaling**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-Scaling.html) — Scaling pixbufs and scaling and compositing pixbufs

[**Rendering**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-gdk-pixbuf-rendering.html) — Rendering a pixbuf to a GDK drawable.

[**Drawables to Pixbufs**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-gdk-pixbuf-from-drawables.html) — Getting parts of a GDK drawable's image data into a pixbuf.

[**Utilities**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-Utilities.html) — Utility and miscellaneous convenience functions.

[**Animations**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-Animations.html) — Animated images.

[**GdkPixbufLoader**](https://developer.gnome.org/gdk-pixbuf/stable/GdkPixbufLoader.html) — Application-driven progressive image loading.

[**Module Interface**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-Module-Interface.html) — Extending GdkPixBuf

[**gdk-pixbuf Xlib initialization**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-gdk-pixbuf-Xlib-initialization.html) — Initializing the gdk-pixbuf Xlib library.

[**Xlib Rendering**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-Xlib-Rendering.html) — Rendering a pixbuf to an X drawable.

[**X Drawables to Pixbufs**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-X-Drawables-to-Pixbufs.html) — Getting parts of an X drawable's image data into a pixbuf.

[**XlibRGB**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-XlibRGB.html) — Rendering RGB buffers to X drawables.

[**II. Tools Reference**](https://developer.gnome.org/gdk-pixbuf/stable/rn02.html)

[**gdk-pixbuf-csource**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-csource.html) — C code generation utility for GdkPixbuf images

[**gdk-pixbuf-query-loaders**](https://developer.gnome.org/gdk-pixbuf/stable/gdk-pixbuf-query-loaders.html) — GdkPixbuf loader registration utility

* ***Cairo***

A Vector Graphics Library

[**Drawing**](http://cairographics.org/manual/cairo-drawing.html)

[cairo\_t](http://cairographics.org/manual/cairo-cairo-t.html) — The cairo drawing context

[Paths](http://cairographics.org/manual/cairo-Paths.html) — Creating paths and manipulating path data

[cairo\_pattern\_t](http://cairographics.org/manual/cairo-cairo-pattern-t.html) — Sources for drawing

[Regions](http://cairographics.org/manual/cairo-Regions.html) — Representing a pixel-aligned area

[Transformations](http://cairographics.org/manual/cairo-Transformations.html) — Manipulating the current transformation matrix

[text](http://cairographics.org/manual/cairo-text.html) — Rendering text and glyphs

[Raster Sources](http://cairographics.org/manual/cairo-Raster-Sources.html) — Supplying arbitrary image data

[**Fonts**](http://cairographics.org/manual/cairo-fonts.html)

[cairo\_font\_face\_t](http://cairographics.org/manual/cairo-cairo-font-face-t.html) — Base class for font faces

[cairo\_scaled\_font\_t](http://cairographics.org/manual/cairo-cairo-scaled-font-t.html) — Font face at particular size and options

[cairo\_font\_options\_t](http://cairographics.org/manual/cairo-cairo-font-options-t.html) — How a font should be rendered

[FreeType Fonts](http://cairographics.org/manual/cairo-FreeType-Fonts.html) — Font support for FreeType

[Win32 Fonts](http://cairographics.org/manual/cairo-Win32-Fonts.html) — Font support for Microsoft Windows

[Quartz (CGFont) Fonts](http://cairographics.org/manual/cairo-Quartz-(CGFont)-Fonts.html) — Font support via CGFont on OS X

[User Fonts](http://cairographics.org/manual/cairo-User-Fonts.html) — Font support with font data provided by the user

[**Surfaces**](http://cairographics.org/manual/cairo-surfaces.html)

[cairo\_device\_t](http://cairographics.org/manual/cairo-cairo-device-t.html) — interface to underlying rendering system

[cairo\_surface\_t](http://cairographics.org/manual/cairo-cairo-surface-t.html) — Base class for surfaces

[Image Surfaces](http://cairographics.org/manual/cairo-Image-Surfaces.html) — Rendering to memory buffers

[PDF Surfaces](http://cairographics.org/manual/cairo-PDF-Surfaces.html) — Rendering PDF documents

[PNG Support](http://cairographics.org/manual/cairo-PNG-Support.html) — Reading and writing PNG images

[PostScript Surfaces](http://cairographics.org/manual/cairo-PostScript-Surfaces.html) — Rendering PostScript documents

[Recording Surfaces](http://cairographics.org/manual/cairo-Recording-Surfaces.html) — Records all drawing operations

[Win32 Surfaces](http://cairographics.org/manual/cairo-Win32-Surfaces.html) — Microsoft Windows surface support

[SVG Surfaces](http://cairographics.org/manual/cairo-SVG-Surfaces.html) — Rendering SVG documents

[Quartz Surfaces](http://cairographics.org/manual/cairo-Quartz-Surfaces.html) — Rendering to Quartz surfaces

[XCB Surfaces](http://cairographics.org/manual/cairo-XCB-Surfaces.html) — X Window System rendering using the XCB library

[XLib Surfaces](http://cairographics.org/manual/cairo-XLib-Surfaces.html) — X Window System rendering using XLib

[XLib-XRender Backend](http://cairographics.org/manual/cairo-XLib-XRender-Backend.html) — X Window System rendering using XLib and the X Render extension

[Script Surfaces](http://cairographics.org/manual/cairo-Script-Surfaces.html) — Rendering to replayable scripts

[**Utilities**](http://cairographics.org/manual/cairo-support.html)

[cairo\_matrix\_t](http://cairographics.org/manual/cairo-cairo-matrix-t.html) — Generic matrix operations

[Error handling](http://cairographics.org/manual/cairo-Error-handling.html) — Decoding cairo's status

[Version Information](http://cairographics.org/manual/cairo-Version-Information.html) — Compile-time and run-time version checks.

[Types](http://cairographics.org/manual/cairo-Types.html) — Generic data types

**Reasons for using *GTK+***

* *Language Bindings*

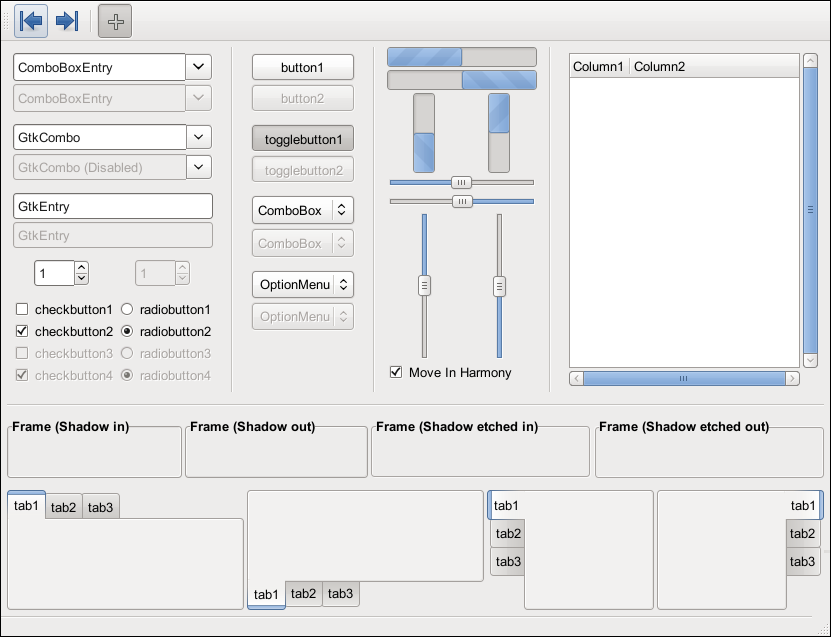
GTK+ is available in many other programming languages thanks to the language bindings available. This makes GTK+ quite an attractive toolkit for application development.

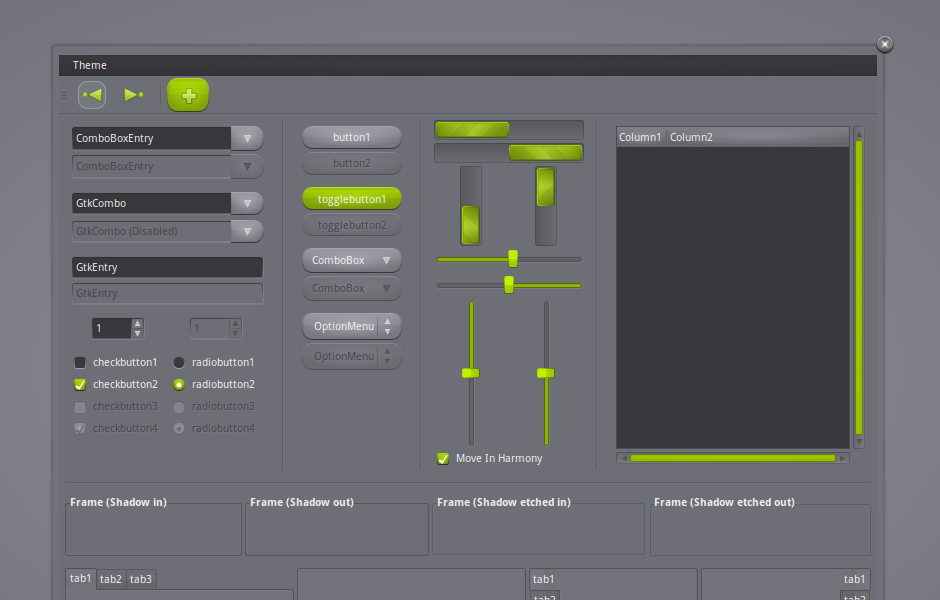
* *Interfaces*

GTK+ has a comprehensive collection of core widgets and interfaces for use in your application.

* Windows (normal window or dialog, about and assistant dialogs)
* Displays (label, image, progress bar, status bar)
* Buttons and toggles (check buttons, radio buttons, toggle buttons and link buttons)
* Numerical (horizontal or vertical scales and spin buttons) and text data entry (with or without completion)
* Multi-line text editor
* Tree, list and icon grid viewer (with customizable renderers and model/view separation)
* Combo box (with or without an entry)
* Menus (with images, radio buttons and check items)
* Toolbars (with radio buttons, toggle buttons and menu buttons)
* GtkBuilder (creates your user interface from XML)
* Selectors (color selection, file chooser, font selection)
* Layouts (tabulated widget, table widget, expander widget, frames, separators and more)
* Status icon (notification area on Linux, tray icon on Windows)
* Printing widgets
* Recently used documents (menu, dialog and manager)

**Examples of GUIs created using GTK+**





**Proposed Layout**

