**Visual Paradigm**

Desktop Application Wireframes
Written Date : September 05, 2016

Desktop application wireframes are screen sketches of applications that run on a desktop computer. We create desktop wireframe for confirming application design ideas with end users.
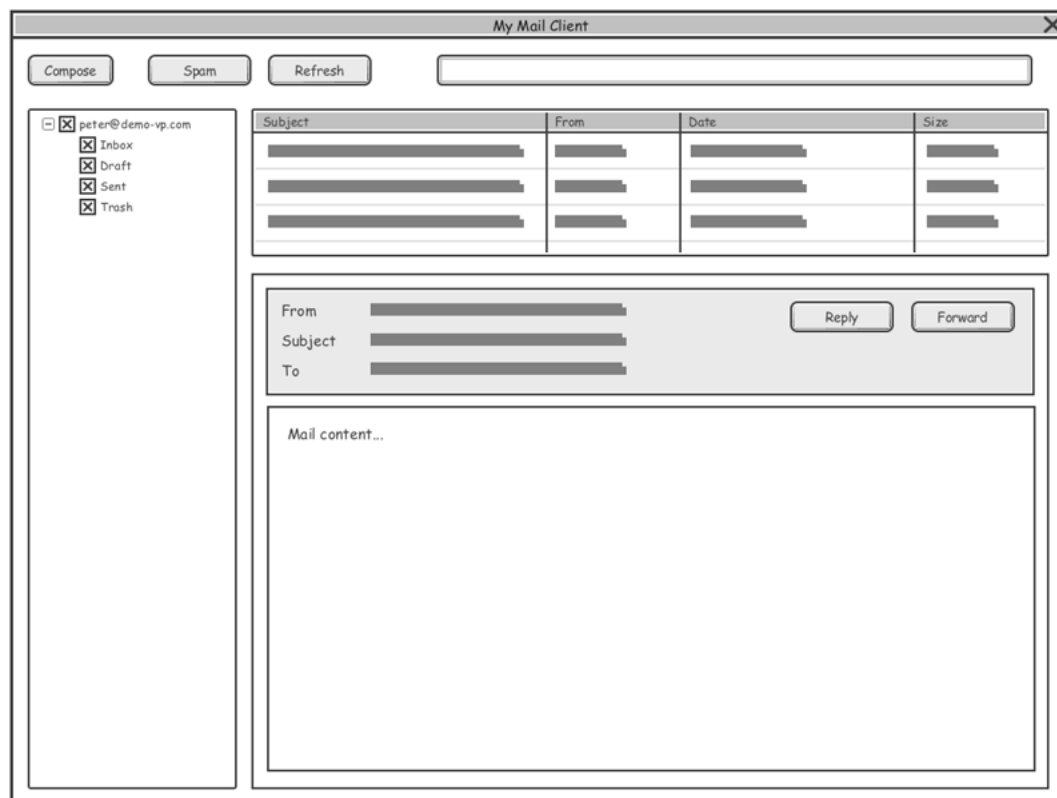
What is a Desktop Application Wireframe?
Desktop application wireframes are screen sketches of application that run on a desktop computer. It helps you present and explain design ideas of applications to customers, which ultimately leads to a consensus on the ideas proposed.
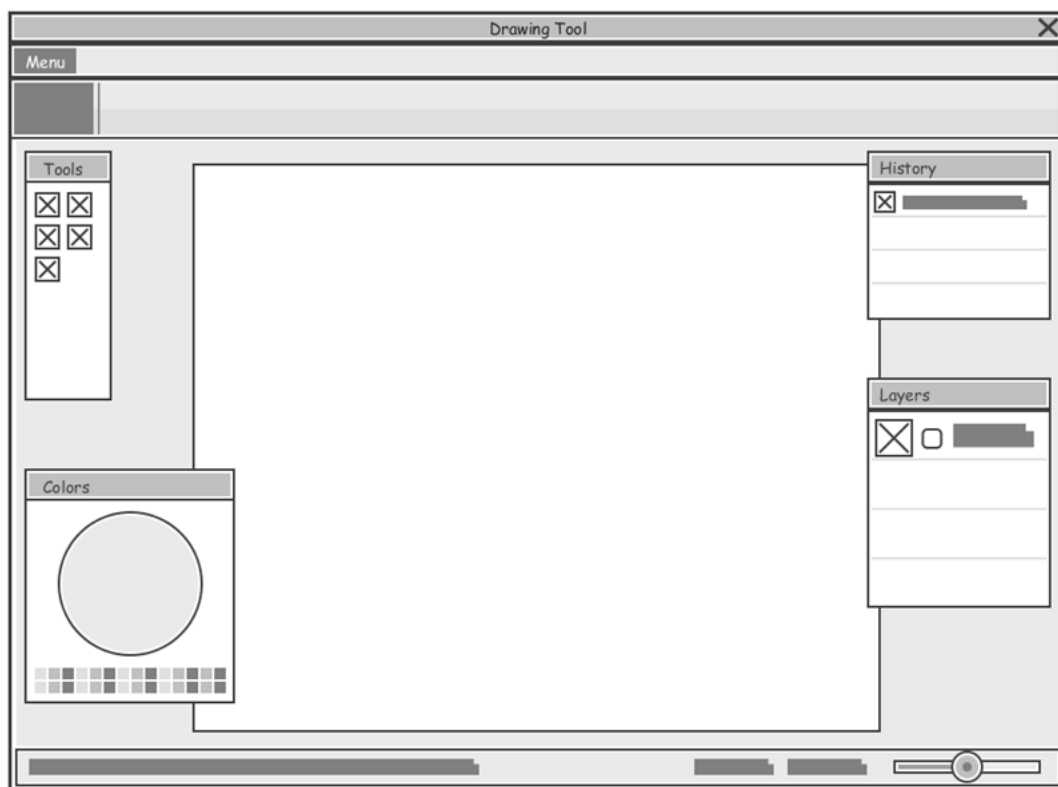
Wireframes are intended to use to demonstrate the functionalities, user interactions and screen flows, without explicitly specifying the actual screen components should look like and how the components should behave in order to keep the upfront development effort and cost in its lowest minimum.

Desktop application wireframe example

Here is a desktop application wireframe example created for a general mail program. Although the wireframe is formed by simple components that represent buttons, labels (placeholder), checkboxes, text box and panels, it's already clear enough for stakeholder to get the design ideas.


Desktop application wireframe components

A desktop application wireframe is only a low-fidelity version of an interface. Each desktop application wireframe comprises basic graphic elements that represents the screen components, or the placeholders of screen components. The following are the desktop wireframe components you can use in creating a wireframe.

Image

An image can be an icon or any image data.
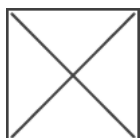
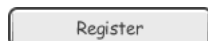**Label**

A label is a text on screen.

Text Field

A text field allows user to type text. Clicking inside the text field shows the cursor at the clicked place. User can then start typing. Text fields also supports functions like ranged text selection, copy, cut and paste.

search...

Enter your name
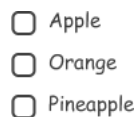
Hello World

Button

A button allows user to click on it for triggering an action. It consists of a text, an icon, or both text and icon that shows the action to trigger.

Register

Checkbox

Checkboxes allow user to select one or more options from a set.
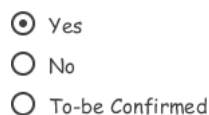
☐ Apple

☐ Orange

☐ Pineapple

Radio Button

Radio buttons allow the user to select one and only one option from a set. If you think that the user needs to see all available options on the screen, use radio buttons. Otherwise, use a radio button instead.
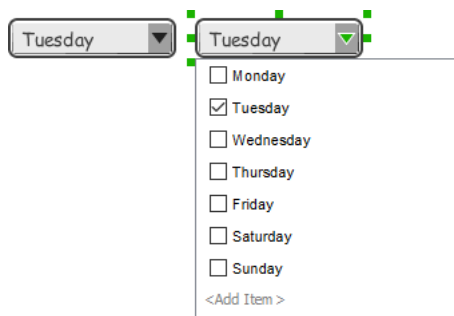
Attending?

⦿ Yes

◯ No

◯ To-be Confirmed

Combo box

A combo box provides a drop-down menu for the quick selection of a value from a set. A combo box shows in its 'body' the currently selected value. By clicking on a combo box, a drop-down menu will be displayed, listing all the available values from which the user can make a selection.

Tuesday ▼   Tuesday ▼
☐ Monday
☑ Tuesday
☐ Wednesday
☐ Thursday
☐ Friday
☐ Saturday
☐ Sunday
<Add Item >

Spinner

Spinner lets user to choose a value from a range of available choices, which behaves a bit like a combo box. But unlike combo boxes, spinners do not have a drop-down menu. You can change a value by pressing the up and down button attached to the spinner.

1
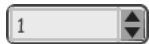
Progress Bar

A progress bar is a bar that indicates the progress of certain running operation. It displays how far the operation has progressed.

Slider

A seek bar allows the selection of progress level. A seek bar has a draggable thumb. User can drag the thumb to left or right to set the current progress level or use the arrow keys.

Panel

A panel is a visual container of wireframe components.

Split bar
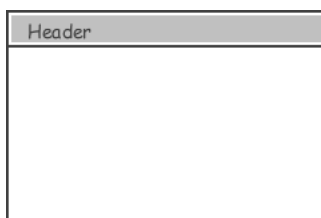
A split bar provides visual separation between two panels.

Accordion Panel

An accordion panel is a collapsible panel that allows user to display or hide a section of content, as presented in the panel.

Header

Scroll bar

A scroll bar allows user to update the visual content in a widget by dragging its knob up and down, or left and right, depending on the orientation of the bar.

List View

A list view displays a list of scrollable items.

Computer
Camera
TV

Table

A table allows the structure representation of components.
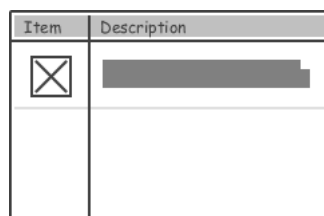
Tree

A tree allows the representation of items in the form of a hierarchy.

Dialog

A dialog is a small window that prompts the user to make a decision (e.g. Confirm/Cancel). A dialog is shown on top of all the other components on a screen. It is used for modal events that require users to take an action before they can continue.
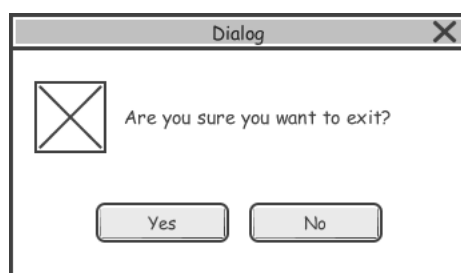
Toolbar

A toolbar provides user with access to available user actions by clicking on the buttons on the bar.

Menubar

A menubar provides user with access to available menus.

Menu

A menu provides user with access to available user actions.
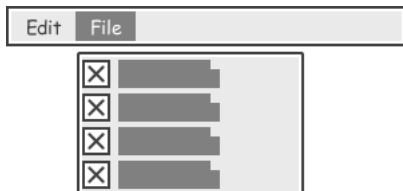
Segment Control

A segment control allows user to change the selection between two or more choices by clicking on it.

Annotation

An annotation is a piece of text that can be used to describe a control or behavior on a wireframe.
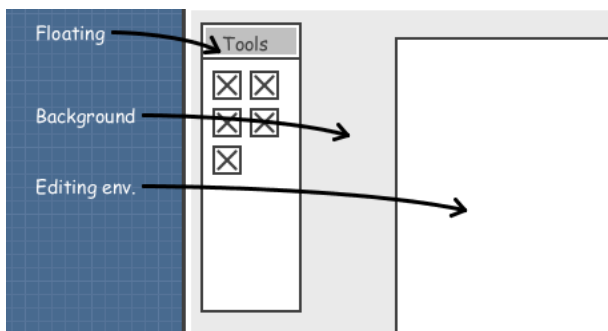
Rectangle

A rectangle shape that can be used for annotation and highlight in the wireframe.

Oval

An oval shape that can be used for annotation and highlight in the wireframe.

Polygon

A polygon that can be used for annotation or highlight in the wireframe. You can add more points to the borders of a polygon and adjust their position.

Line

A line that can be used for annotation or highlight in the wireframe.

Benefits of using wireframe

Clarify user interface

Usually, clients do not understand technical screen design jargons like input control, gridlayout, picker, etc. Desktop application wireframe comprise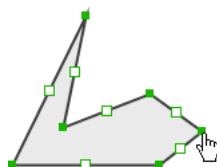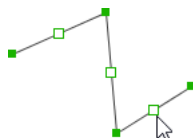s basic graphic elements that everyone can understand. Clients knows how the app will function and how they can interact with it to achieve their expectations.

Cost-efficient

Full-blown, high-fi screen designs takes time to develop, which lead to high development cost. Wireframing is easy to create and edit. It is an inexpensive way to create basic screen sketches. It also makes tweaking or even overhauling sketches simple and inexpensive.

Engaged clients

Because wireframes are rough and loose, they give clients the room to brainstorm and voice their suggestions. And, because the turnaround time of refining wireframe is short, clients are actively involved in providing feedback, which makes them more likely to sign off the final design.

More willing to make changes

The problem with confirming design ideas using complete screen mockup or prototype, is that a considerable amount of work has already been done, and will involve a considerable amount of work to revise (or restart), which means extra time, effort and expense. In that situation, customers and development team are more reluctant to voice concerns and request changes. On the contrary, it takes much less time to produce and revise wireframes. Everyone will be more willing to request and make changes.

Early consideration of usability

User experience (UX) is extremely important in app development. The use of wireframe brings the consideration of user experience to the beginning of project. Without developing any prototypes or drawing any real screen design, users can still experience how the app will work.

How to use desktop application wireframe effectively?

The use of wireframe can bring many benefits to both the development team and clients, but this is the case only when you use it wisely and properly. A typical misuse of wireframe is to treat it as a replacement of screen design. This makes the production and refinement difficult and costly, reducing the usefulness of wireframing. In this section, we will go through some of the effective wireframing tips.

A wireframe is intended to be simple and just enough. It is simple so that it can be produced quickly and easily, and makes no hesitation for a discard and re-work. The low-fi presentation also makes it more comprehensive and communicative. Therefore, do not need to spend too much time on beautifying the drawing, aligning things, or using pretty typology and etc. In a wireframe, instead of showing any actual content, we can replace a large chunk of text (the actual context) with a placeholder of text. This is to avoid time being spent on preparing the content unnecessarily, and to prevent the readers from being distracted by the text content. But if the displaying of text is needed, you may consider placing some dummy text there instead. You can easily find a dummy text generator on the internet. The use of annotation helps you describes an element (e.g. "Company logo") or to explain something related to its behavior (e.g. "Hide in 5 seconds"). Use it if necessary. But again, don't attempt to document each of the wireframe elements. You should only use annotation whenever it necessary.
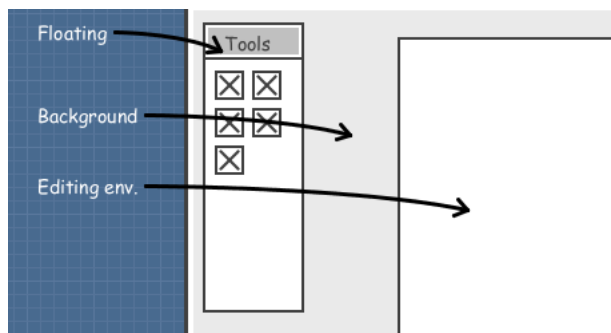
 Wireframes can be hand-drawn, but we usually create wireframes with software for more efficient and easier to manage of our works. Besides, some wireframe software provides you with features that paper-and-pencil cannot accomplish. Here are three of them:

1. State - The wireframing tool of Visual Paradigm supports the concept of state, which allows you to create a child wireframe based on an existing one. It is not only save you time in creating a screen flow with a sequence of similar child wireframes, it also makes refinements of the related child wireframes much easier (as we make changes in the initial state of a wireframe, the changes will also be reflected in all its' child states wireframes)

2.   Storyboard - A storyboard presents the screen flow of a particular scenario. It makes the wireframes more manageable and the presentation much easier.

3.   Managing wireframes by User story - User story is an agile tool for recording user's concerns and requirements. To include wireframes as part of a user story's scenario shows how user will use the feature in performing part of their job described in that user story. Besides, when developer start implementing the user story, he can check the wireframe to gain ideas about user's expectation.

Visual Paradigm provides all the wireframe tools and elements you need in drawing desktop application wireframes, Android phone wireframes, Android tablet wireframes, web wireframe and iOS apps wireframe (iPhone and iPad). It also supports **agile software development** and **UX design tools**, which includes **user story tool**, **sprint development**, **storyboard**, etc.

Try now for FREE

References
1.   Learn about Visual Paradigm's agile development tools

Visual Paradigm

Visual Paradigm home page
(https://www.visual-paradigm.com/)

Visual Paradigm handbooks
(https://www.visual-paradigm.com/learning/handbooks/)