

# 界面测试

界面是软件与用户交互的最直接的层，界面的好坏决定用户对软件的第一印象。而且设计良好的界面能够引导用户自己完成相应的操作，起到向导的作用。同时界面如同人的面孔，具有吸引用户的直接优势。设计合理的界面能给用户带来轻松愉悦的感受和成功的感觉，相反由于界面设计的失败，让用户有挫败感，再实用强大的功能都可能在用户的畏惧与放弃中付诸东流。目前界面的设计引起软件设计人员的重视的程度还远远不够，直到最近网页制作的兴起，才受到专家的青睐。而且设计良好的界面由于需要具有艺术美的天赋而遭拒绝。

目前流行的界面风格有三种方式：多窗体、单窗体以及资源管理器风格，无论那种风格，以下规则是应该被重视的。

## 1: 易用性:

按钮名称应该易懂，用词准确，屏弃没楞两可的字眼，要与同一界面上的其他按钮易于区分，能望文知意最好。理想的情况是用户不用查阅帮助就能知道该界面的功能并进行相关的正确操作。

易用性细则:

- 1):完成相同或相近功能的按钮用 Frame 框起来，常用按钮要支持快捷方式。
- 2):完成同一功能或任务的元素放在集中位置，减少鼠标移动的距离。
- 3):按功能将界面划分区域块，用 Frame 框括起来，并要有功能说明或标题。
- 4):界面要支持键盘自动浏览按钮功能，即按 Tab 键、回车键的自动切换功能。
- 5):界面上首先要输入的和重要信息的控件在 Tab 顺序中应当靠前，位置也应放在窗口上较醒目的位置。
- 6):同一界面上的控件数最好不要超过 10 个，多于 10 个时可以考虑使用分页界面显示。
- 7):分页界面要支持在页面间的快捷切换，常用组合快捷键 Ctrl+Tab
- 8):默认按钮要支持 Enter 及选操作，即按 Enter 后自动执行默认按钮对应操作。
- 9):可寫控制項檢測到非法輸入後應給出說明並能自動獲得焦點。
- 10):Tab 键的顺序与控件排列顺序要一致，目前流行总体从上到下，同时行间从左到右的方式。
- 11):核取方塊和選項框按選擇幾率的高底而先後排列。
- 12):核取方塊和選項框要有默認選項，並支援 Tab 選擇。
- 13):選項數相同時多用選項框而不用下拉清單框。

14): 界面空间较小时使用下拉框而不用选项框。

15): 选项数较少时使用选项框, 相反使用下拉列表框。

16): 专业性强的软件要使用相关的专业术语, 通用性界面则提倡使用通用性词语。

## **2: 规范性:**

通常界面设计都按 Windows 界面的规范来设计, 可以说: 界面遵循规范化的程度越高, 则易用性相应的就越好。小型软件一般不提供工具厢。

规范性细则:

1): 常用菜单要有命令快捷方式。

2): 完成相同或相近功能的菜单用横线隔开放在同一位置。

3): 菜单前的图标能直观的代表要完成的操作。

4): 菜单深度一般要求最多控制在三层以内。

5): 工具栏要求可以根据用户的要求自己选择定制。

6): 相同或相近功能的工具栏放在一起。

7): 工具栏中的每一个按钮要有及时提示信息。

8): 一条工具栏的长度最长不能超出屏幕宽度。

9): 工具栏的图标能直观的代表要完成的操作。

10): 系统常用的工具栏设置默认放置位置。

11): 工具栏太多时可以考虑使用工具箱。

12): 工具箱要具有可增减性, 由用户自己根据需求定制。

13): 工具箱的默认总宽度不要超过屏幕宽度的 1/5。

14): 状态条要能显示用户切实需要的信息, 常用的有:

目前的操作、系统状态、用户位置、用户信息、提示信息、错误信息等, 如果某一操作需要的时间较长, 还应该显示进度条和进程提示。

15): 滚动条的长度要根据显示信息的长度或宽度能及时变换, 以利于用户了解显示信息的位置和百分比。

16): 状态条的高度以放置五好字为宜, 滚动条的宽度比状态条的略窄。

17): 菜单和工具条要有清楚的界限; 菜单要求凸出显示, 这样在移走工具条时仍有立体感。

18): 菜单和状态条中通常使用 5 号字体。工具条一般比菜单要宽, 但不要宽的太多, 否则看起来很不协调。

19): 右键快捷菜单采用与菜单相同的准则。

### **3: 帮助设施:**

系统应该提供详尽而可靠的帮助文档, 在用户使用产生迷惑时可以自己寻求解决方法。

帮助设施细则:

- 1): 帮助文档中的性能介绍与说明要与系统性能配套一致。(我们的系统帮助文档都是系统的祖先时期的说明, 让人困惑)。
- 2): 打包新系统时, 对作了修改的地方在帮助文档中要做相应的修改。
- 3): 操作时要提供及时调用系统帮助的功能。常用 F1。
- 4): 在界面上调用帮助时应该能够及时定位到与该操作相对的帮助位置。也就是说帮助要有即时针对性。
- 5): 最好提供目前流行的联机帮助格式或 HTML 帮助格式。
- 6): 用户可以用关键词在帮助索引中搜索所要的帮助, 当然也应该提供帮助主题词。
- 7): 如果没有提供书面的帮助文档的话, 最好有打印帮助的功能。
- 8): 在帮助中应该提供我们的技术支持方式, 一旦用户难以自己解决可以方便的寻求新的帮助方式。

### **4: 合理性:**

屏幕对角线相交的位置是用户直视的地方, 正上方四分之一处为易吸引用户注意力的位置, 在放置窗体时要注意利用这两个位置。

合理性细则:

- 1): 父窗体或主窗体的中心位置应该在对角线焦点附近。
- 2): 子窗体位置应该在主窗体的左上角或正中。
- 3): 多个子窗体弹出时应该依次向右下方偏移, 以显示窗体出标题为宜。
- 4): 重要的命令按钮与使用较频繁的按钮要放在界面上注目的位置。
- 5): 错误使用容易引起界面退出或关闭的按钮不应该放在易点击的位置。横排开头或最后与竖排最后为易点位置。
- 6): 与正在进行的操作无关的按钮应该加以屏蔽(Windows 中用灰色显示, 没法使用该按钮)。
- 7): 对可能造成数据无法恢复的操作必须提供确认信息, 给用户放弃选择的机会。
- 8): 非法的输入或操作应有足够的提示说明。
- 9): 对运行过程中出现问题而引起错误的地方要有提示, 让用户明白错误出处, 避免形成无

限期的等待。

10)：提示、警告、或错误说明应该清楚、明了、恰当。

## 5: 美观与协调性:

界面应该大小适合美学观点，感觉协调舒适，能在有效的范围内吸引用户的注意力。

美观与协调性细则：

- 1)：长宽接近黄金点比例，切忌长宽比例失调、或宽度超过长度。
- 2)：布局要合理,不宜过于密集，也不能过于空旷，合理的利用空间。
- 3)：按钮大小基本相近，忌用太长的名称，免得占用过多的界面位置。
- 4)：按钮的大小要与界面的大小和空间要协调。
- 5)：避免空旷的界面上放置很大的按钮。
- 6)：放置完控件后界面不应有很大的空缺位置。
- 7)：字体的大小要与界面的大小比例协调，通常使用的字体中宋体 9-12 较为美观，很少使用超过 12 号的字体。
- 8)：前景与背景色搭配合理协调，反差不宜太大，最好少用深色，如大红、大绿等。常用色考虑使用 Windows 界面色调。
- 9)：如果使用其他颜色，主色调要柔和，具有亲和力与磁力，坚决杜绝刺目的颜色。
- 10)：大型系统常用的主色有“#E1E1E1”、“#EFEFEF”、“#C0C0C0”等。
- 11)：界面风格要保持一致，字的大小、颜色、字体要相同，除非是需要艺术处理或有特殊要求的地方。
- 12)：如果窗体支持最小化和最大化或放大时，窗体上的控件也要随着窗体而缩放；切忌只放大窗体而忽略控件的缩放。
- 13)：对于含有按钮的界面一般不应该支持缩放，即右上角只有关闭功能。
- 14)：通常父窗体支持缩放时，子窗体没有必要缩放。
- 15)：如果能给用户自定义界面风格则更好，由用户自己选择颜色、字体等。

## 6: 菜单位置:

菜单是界面上最重要的元素，菜单位置按照按功能来组织。

菜单测试细则：

- 1)：菜单通常采用“常用--主要--次要--工具--帮助”的位置排列，符合流行的 Windows 风格。

- 2): 常用的有“文件”、“編輯”,“查看”等,幾乎每個系統都有這些選項,當然要根據不同的系統有所取捨。
- 3): 下拉菜单要根据菜单选项的含义进行分组,並且按照一定的规则进行排列,用横线隔开。
- 4): 一组菜单的使用有先后要求或有向导作用时,应该按先后次序排列。
- 5): 没有顺序要求的菜单项按使用频率和重要性排列,常用的放在开头,不常用的靠后放置;重要的放在开头,次要的放在后边。
- 6): 如果菜单选项较多,应该采用加长菜单的长度而减少深度的原则排列。
- 7): 菜单深度一般要求最多控制在三层以内。
- 8): 对常用的菜单要有快捷命令方式,组合原则见 8。
- 9): 对与进行的操作无关的菜单要用屏蔽的方式加以处理,如果采用动态加载方式——即只有需要的菜单才显示——最好。
- 10): 菜单前的图标不宜太大,与字高保持一直最好。
- 11): 主菜单的宽度要接近,字数不应多于四个,每个菜单的字数能相同最好。
- 12): 主菜单数目不应太多,最好为单排布置。
- 13): 菜单条是否显示在合适的语境中?
- 14): 应用程序的菜单条是否显示系统相关的特性(如时钟显示)?
- 15): 下拉式操作能正确工作吗?
- 16): 菜单、调色板和工具条是否工作正确?
- 17): 是否适当地列出了所有的菜单功能和下拉式子功能?
- 18): 是否可能通过鼠标访问所有的菜单功能?
- 19): 相同功能按钮的图标和文字是否一致?
- 20): 是否能够用其他的文本命令激活每个菜单功能?
- 21): 菜单功能是否随当前的窗口操作加亮或变灰?
- 22): 菜单功能是否正确执行?
- 23): 菜单功能的名字是否具有自解释性?
- 24): 菜单项是否有帮助,是否语境相关?
- 25): 在整个交互式语境中,是否可以识别鼠标操作?
- 26): 如果要求多次点击鼠标,是否能够在语境正确识别?
- 27): 如果鼠标有多个按钮,是否能够在语境中正确识别?
- 28): 光标、处理指示器和识别指针是否随操作恰当地改变?

## 7:独特性:

如果一味的遵循业界的界面标准,则会丧失自己的个性.在框架符合以上规范的情况下,设计具有自己独特风格的界面尤为重要.尤其在商业软件流通中有着很好的潜移默化的广告效用。

测试细则:

- 1): 安装界面上应有单位介绍或产品介绍,并有自己的图标。
- 2): 主界面,最好是大多数界面上要有公司图标。
- 3): 登录界面上要有本产品的标志,同时包含公司图标。
- 4): 帮助菜单的“关于”中应有版权和产品信息。
- 5): 公司的系列产品要保持一直的界面风格,如背景色、字体、菜单排列方式、图标、安装过程、按钮用语等应该大体一致。

## 8: 快捷方式的组合

在菜单及按钮中使用快捷键可以让喜欢使用键盘的用户操作得更快一些 在西文Windows及其应用软件中快捷键的使用大多是一致的。

菜单中:

1):面向事务的组合有:

Ctrl-D 删除 ; Ctrl-F 寻找 ; Ctrl -H替换; Ctrl-I 插入 ; Ctrl-N 新记录 ; Ctrl-S 保存 Ctrl-O 打开。

2): 列表:

Ctrl-R , Ctrl-G定位; Ctrl-Tab下一分页窗口或反序浏览同一页面控件;。

3):编辑:

Ctrl-A全选; Ctrl-C 拷贝; Ctrl-V 粘贴; Ctrl-X 剪切; Ctrl-Z撤消操作; Ctrl-Y恢复操作。

4)文件操作:

Ctrl-P 打印; Ctrl-W 关闭。

5):系统菜单

Alt-A文件; Alt-E编辑; Alt-T工具; Alt-W窗口; Alt-H帮助。

6):MS Windows保留键:

Ctrl-Esc 任务列表 ; Ctrl-F4 关闭窗口; Alt-F4 结束应用; Alt-Tab 下一应用 ; Enter

缺省按钮/确认操作 ; Esc 取消按钮/取消操作 ; Shift-F1 上下文相关帮助。

按钮中:

可以根据系统需要而调节, 以下只是常用的组合。

Alt-Y确定(是); Alt-C取消; Alt-N 否; Alt-D删除; Alt-Q退出; Alt-A添加; Alt-E编辑;  
Alt-B浏览; Alt-R读; Alt-W写。

这些快捷键也可以作为开发中文应用软件的标准, 但亦可使用汉语拼音的开头字母。

## 9:安全性考虑:

在界面上通过下列方式来控制出错几率, 会大大减少系统因用户人为的错误引起的破坏。开发者应当尽量周全地考虑到各种可能发生的问题, 使出错的可能降至最小。如应用出现保护性错误而退出系统, 这种错误最容易使用户对软件失去信心。因为这意味着用户要中断思路, 并费时费力地重新登录, 而且已进行的操作也会因没有存盘而全部丢失。

安全性细则:

- 1): 最重要的是排除可能会使应用非正常中止的错误。
- 2): 应当注意尽可能避免用户无意录入无效的数据。
- 3): 采用相关控件限制用户输入值的种类。
- 4): 当用户作出选择的可能性只有两个时, 可以采用单选框。
- 5): 当选择的可能再多一些时, 可以采用复选框, 每一种选择都是有效的, 用户不可能输入任何一种无效的选择。
- 6): 当选项特别多时, 可以采用列表框, 下拉式列表框。
- 7): 在一个应用系统中, 开发者应当避免用户作出未经授权或没有意义的操作。
- 8): 对可能引起致命错误或系统出错的输入字符或动作要加限制或屏蔽。
- 9): 对可能发生严重后果的操作要有补救措施。通过补救措施用户可以回到原来的正确状态。
- 10): 对一些特殊符号的输入、与系统使用的符号相冲突的字符等进行判断并阻止用户输入该字符。
- 11): 对错误操作最好支持可逆性处理, 如取消系列操作。
- 12): 在输入有效性字符之前应该阻止用户进行只有输入之后才可进行的操作。
- 13): 对可能造成等待时间较长的操作应该提供取消功能。
- 14): 特殊字符常有: ; ' " < , \ ' : " [ " { 、 \ | } ] +=) -( \_ \* & & ^ % \$ # @ ! , . . ? / 还有空格。

- 15): 与系统采用的保留字符冲突的要加以限制。
- 16): 在读入用户所输入的信息时, 根据需要选择是否去掉前后空格。
- 17): 有些读入数据库的字段不支持中间有空格, 但用户切实需要输入中间空格, 这时要在程序中加以处理。

## **10:多窗口的应用与系统资源:**

设计良好的软件不仅要有完备的功能, 而且要尽可能的占用最底限度的资源。

- 1): 在多窗口系统中, 有些界面要求必须保持在最顶层, 避免用户在打开多个窗口时, 不停的切换甚至最小化其他窗口来显示该窗口。
- 2): 在主界面载入完毕后自动卸出内存, 让出所占用的WINDOWS系统资源。
- 3): 关闭所有窗体, 系统退出后要释放所占的所有系统资源, 除非是需要后台运行的系统。
- 4): 尽量防止对系统的独占使用。
- 5): 窗口能否基于相关的输入或菜单命令适当地打开?
- 6): 窗口能否改变大小、移动和滚动?
- 7): 窗口中的数据内容能否使用鼠标、功能键、方向箭头和键盘访问?
- 8): 当被覆盖并重调用后, 窗口能否正确地再生?
- 9): 需要时能否使用所有窗口相关的功能?
- 10): 所有窗口相关的功能是可操作的吗?
- 11): 是否有相关的下拉式菜单、工具条、滚动条、对话框、按钮、图标和其他控制可为窗口可用, 并适当地显示?
- 12): 显示多个窗口时, 窗口的名称是否被适当地表示?
- 13): 活动窗口是否被适当地加亮?
- 14): 如果使用多任务, 是否所有的窗口被实时更新?
- 15): 多次或不正确按鼠标是否会导致无法预料的副作用?
- 16): 窗口的声音和颜色提示和窗口的操作顺序是否符合需求?
- 17): 窗口是否正确地关闭?