

# How do you normally make a program look beautiful? [closed]

Asked 15 years, 1 month ago

Modified 15 years ago

Viewed 5k times



7



As it currently stands, this question is not a good fit for our Q&A format. We expect answers to be supported by facts, references, or expertise, but this question will likely solicit debate, arguments, polling, or extended discussion. If you feel that this question can be improved and possibly reopened, [visit the help center](#) for guidance.

Closed 11 years ago.

How can I make an Application look nice and not like an amateur pulled it together?

I mean graphic-wise.

Is there some sort of book you can read regarding beautiful program layouts, etc?

I put this in Community Wiki so please feel free to leave your opinions that way we can all learn. :D

Edit: Oh my god. I completely forgot to mention what **kind** of program. I mean desktop applications. Not web applications. :D

[layout](#)[wiki](#)[Share](#)[edited Nov 11, 2009 at 15:39](#)[Improve this question](#)[Follow](#)

community wiki

[5 revs, 4 users 95%](#)[Sergio Tapia](#)

- 
- 4    what kind of program? web based? client based? video game? what there are literally thousands of books you can read on this, in addition to millions of google results...  
– [Jason](#) Nov 10, 2009 at 21:30
- 
- 7    Yes, there is one single book that can magically imbue you with a complete design education, but I'm not allowed to tell you what it is because then all the art schools in the world would hunt me down and kill me for ruining their businesses.  
– [Azeem.Butt](#) Nov 10, 2009 at 21:36
- 
- 4    I think he means a rule of thumb. There are fundamental rule that apply to all applications like keep it simple, satisfy a business need etc. – [Phil](#) Nov 10, 2009 at 21:36
- 
- @Jason: A button is a button, be it web or desktop...  
– [OMG Ponies](#) Nov 10, 2009 at 21:39
- 

18 Answers

Sorted by:

Highest score (default)



Find a graphic designer.

29



I'm not trying to be snarky, I'm just sayin'. I fancy myself a pretty good developer, but I am at stick-figure level when it comes to graphic design.



The skills (and the tools) for good graphic design are fundamentally different than the skills of a developer.



Share Improve this answer

edited Nov 11, 2009 at 21:15

Follow

community wiki

3 revs

Robert Harvey

---

6 You are right. HOWEVER, Software Engineers must know a little about design themselves in order to be able to contribute in making the product good. Saying, "just hire a design" is like having someone build you a house without having considering whether or not you think it looks good. I learned this the hard way. – [Phil](#) Nov 10, 2009 at 21:35

---

1 Agreed. Or be a designer. I know people who are good at both programming and designing, but learning to be a designer requires almost as much effort as learning to be a good programmer (i.e. about a lifetime) ... – [Rasmus Kaj](#) Nov 10, 2009 at 21:35

---

1 +1: find a graphical designer. Programmers are code designers, not graphical designers. Joel has also said something nice about it:  
[joelonsoftware.com/items/2009/03/09.html](http://joelonsoftware.com/items/2009/03/09.html) – [BalusC](#) Nov 10, 2009 at 21:54

---

1 @Phil - There is a difference between a graphical designer and a usability engineer. Software engineers don't need to be a graphical designer. However, engineers working on a UI

should most definitely understand usability engineering.

– [JasCav](#) Nov 10, 2009 at 21:57

---

- 1 -1: Making a sharp distinction between development and GUI design is in my opinion a bad excuse used by developers for not having to learn something they're bad at or don't like. For a developer building applications with a UI, basic design skills is just as important as coding and other technical skills. That of course doesn't mean one cannot hire in UI experts as well.

– [jonsb](#) Nov 11, 2009 at 16:07

---



9



Three things:

1. Hire a designers that knows the business models and customers; unless you are good at it yourself
2. To help you provide meaningful feedback to the designer, read a book like I like [Don't Make Me Think: A Common Sense Approach to Web Usability, 2nd Edition](#) by Steve Krug
3. Study many good Apps that have had success in the business of interest to you. Be inspired (aka copy) good ideas from them and mix in your own ideas. A prime example is how [Google search](#) "inspired" Microsoft's Bing

Share Improve this answer

[edited Nov 10, 2009 at 22:07](#)

Follow

---

3 +1 for Don't Make me Think - an excellent book – [ChrisF](#) ♦  
Nov 10, 2009 at 22:46

---



The following terms are really important when it comes to the look and feel of an application

6



**User Interface Design** This talks about how the application has uniform soft colors, similarly sized controls, smooth fonts, appropriate groupings of data elements, etc.



**User Experience** This is a much more difficult thing to attain. This is how intuitive the application is "Are you sure?" and "Are you Really sure?" message boxes, doing away with pop-up's and dialog boxes, auto-save, the flow from one stage to another, accuracy of data saves, and a host of other things that make the user experience truly amazing.

*You cannot have good UX without good UI, but it is definitely possible to have a good UI without a good UX.*

community wiki

[2 revs](#)[Raj More](#)

---

This is almost worse than giving him Let Me Google That For You links. – [Garrett](#) Nov 10, 2009 at 21:41

---

@Garrett I don't think Raj was being sarcastic at all. Those are two terms that are definitely good to know and something a person asking this question may not have been aware of. – [DA.](#) Nov 10, 2009 at 22:10

---

@Garrett: I do not see how my answer merits your comment. The question starts with "How can I make an Application look nice and not like an amateur pulled it together?" and continues on about good software. Why is UI and UX a bad answer for that? – [Raj More](#) Nov 10, 2009 at 22:13

---



5



Just keep redoing it. Seriously, take all outside criticism. Ask them to be specific. If you're doing web work look at as many websites as you can. Sometimes, depending on the creator, you can email the website to ask why they did what they did and get a response. If you're doing desktop find a library you enjoy that can look good (not Swing, hate me if you want.) Then create, destroy, create, destroy, and so on until you're happy.

The biggest issue I think people have is learning to just throw away a component which isn't working for you.

Share Improve this answer

answered Nov 10, 2009 at 21:42

Follow

community wiki  
[wheaties](#)

---

I think that yeah, this kind of quick prototyping is a great way to learn design. If you're not going to have someone else do it, you need to get more familiar with usability yourself, and the best way to speed up that process is with many quick and dirty prototypes. – [Kzqai](#) Nov 10, 2009 at 23:45

---



I have found that going through the various desktop design guides can be pretty useful.

5



1. [Windows User Experience Interaction Guidelines](#)

2. [Apple Human Interface Guidelines](#)

3. [Gnome Human Interface Guides](#)



I don't think any of them can be considered as being "correct", but they will give you a good baseline idea as to what is expected in general as far as basic theory such as alignment, spacing and widget layout goes.

Share Improve this answer

answered Nov 10, 2009 at 22:20

Follow

community wiki  
[Soo Wei Tan](#)

---

I've used the Windows guidelines, and it's a great resource when building Windows UIs. Only problem is the big amount of information in there – [jonsb](#) Nov 11, 2009 at 16:26

---



I recommend reading the [AskTog site](#):

2

Bruce Tognazzini... ...founded the Apple Human Interface Group and acted as Apple's Human Interface Evangelist.



There's been some great stuff in the past, like how making buttons bigger is directly proportionate to their likelihood of being clicked, how mouse movement impacted design, etc.



But if you're not into learning - by all means, find a professional designer.

Share Improve this answer

answered [Nov 10, 2009 at 21:38](#)

Follow

community wiki

[OMG Ponies](#)



For Looks: Follow you platform's Human Interface Guidelines. Use the [Golden Mean](#) to please the eye.

2

[Simplify](#), [Simplify](#), [Simplify](#).



For Behavior: Watch the users use your application. Don't help them, don't correct them. Stand behind a two-way





mirror if you can. You'll quickly find lots of places where your UI could use a tune-up.



Share Improve this answer

answered [Nov 10, 2009 at 22:35](#)

Follow

community wiki  
[Sean McMillan](#)



I do try very hard to convince my users of the beauty of a command line interface.

1



Besides that, I think there is nothing better than "hallway testing". You can always hire an expert for GUI design though.



Share Improve this answer

answered [Nov 10, 2009 at 21:33](#)

Follow

community wiki  
[Johannes Rudolph](#)

---

Haha +1. My job would be over if they would embrace the beautiful command line version I use. – [chimpsarehungry](#) Jul 5, 2013 at 13:30

---



I really like the book [Designing Interfaces](#). It presents many UI design patterns and when they are appropriate to use. It won't teach you how to make things "pretty", but

1



it can help greatly in making your application intuitive and easy for a user to grasp. It also presents the patterns in a platform agnostic way.



Share Improve this answer

answered [Nov 10, 2009 at 22:01](#)

Follow

community wiki

[TheHurt](#)



There's lots of books out there are graphic design, interface design, web application design, css design, etc.

**1**

General rules:



- think in terms of grids
- don't fear white space
- keep it simple
- understand hierarchy
- understand basic color theory
- hire a user interface designer

Share Improve this answer

answered [Nov 10, 2009 at 22:06](#)

Follow

community wiki

[DA.](#)

---

white space - very important! – [Magnus Smith](#) Dec 2, 2009 at 15:56

---



1



Keep your coding work out of the design work, and do your mockups in photoshop or something similar before trying to implement them. Good coding and good visual UI design can be directly opposed to each other very often - what looks good on the outside takes some backend work that that most coders think is monstrous, and good UIs often require custom behavior where using standardized libraries might seem to make more sense from an engineering perspective. Having a target for the functional UI will help you to push yourself beyond the "good enough" results you might get if you let what your libraries do by default determine what the site or application will look like.

Share Improve this answer

[edited Nov 11, 2009 at 15:34](#)

Follow

community wiki

[2 revs](#)

[Dan Monego](#)



1

First of all, usability should always be priority #1 and should drive your design:

- <http://msdn.microsoft.com/en-us/library/ms997577.aspx>



- <http://www.bucketsoft.com/blog/post/is-usability-standard-changing.aspx>



However, that doesn't mean your application has to be ugly. And I disagree with those who say "you either have it or you don't." I think almost anyone can develop basic design skills if they put their mind to it. The very fact that you're asking about it proves that you're interested, which is a big step towards learning. And indeed even as a developer it helps to at least have a good sense for what looks good and how to make an engaging user interface. And if it's important you can always hire a professional designer later to take it to the next level.

There are a few fundamental details to design work that have helped me:

**1.) Use colors wisely.** Dull colors are not bad but they can make the design look boring and uninspired by themselves. If you add exciting, more saturated colors sparingly you can emphasize the more important elements. This will not only improve the look, but it can even be used to improve the usability by bringing attention to elements that you don't want your users to miss. Again, if you overdo this it's kind of like highlighting everything in your physics book. If you do it everywhere, it loses its meaning. I wrote more about [colors and contrast in my answer here](#).

**2.) Use [white space](#) wisely.** This is huge. It doesn't matter if you're talking about a website or a Windows app -- a cluttered application is an ugly one. There is a good

[article about this relating to web design](#), but the same basic rules would apply to a desktop app.

**3.) Fear not the [big fonts](#).** Another means for providing emphasis -- the big font. When combined with proper whitespace, a call-to-action written in great big words can be a powerful thing visually.

Come to think of it, all of these suggestions relate to creating emphasis.

Share Improve this answer

edited May 23, 2017 at 10:29

Follow

community wiki

10 revs

[Steve Wortham](#)



0



There are a lot of sites that offer tips on Graphical User Interface (GUI) design. Google for "GUI design guidelines" and you'll find some. While these tips might not make your application look 'beautiful' they can make it look professionally designed. If you want beautiful graphics you might need to get some help from a graphic artist.

Check [here](#) for a good list of books on GUI design.

Share Improve this answer

answered Nov 10, 2009 at 21:34

Follow



0

I really like this book, from Joel Spolsky:

[<http://www.amazon.com/o/ASIN/B001O9LB86/ref=nosim/betteraddons-20>]



Share Improve this answer

answered Nov 10, 2009 at 21:36

Follow



community wiki  
Luis Lobo Borobia

---



0

General guidelines we use:

1. Make getting around easy: use a toolbar or side menu to access different parts of the application.
2. Make viewing information as easy as possible, and in different formats.
3. Make changing information as consistent as possible. I.e., put New, Edit, and Save buttons in the same place.
4. Make the most common action the easiest.



Specific guidelines we use:

1. Use the same or similar font across the application. Vary by size or bold to catch attention.

2. Use color backgrounds to group and separate types of information.
3. Put the most relevant information together and make the most commonly used button larger than the rest.
4. Don't use a drop-down control where there are only only a few options. Use a list control big enough that one doesn't have to scroll or use a radio-button.
5. Adding some white space can actually make it easier to read the screen or report.

Share Improve this answer

answered Nov 10, 2009 at 22:37

Follow

community wiki  
[wagr](#)



0



Has anyone got any tips for the programmer who has no budget, or is working on a project at home for fun? We can't hire a designer, but we'd like to make a bit of an effort to get our applications looking less home-built.



I suggest showing a screenshot of your creation to family and friends. You'll hate their suggestions because they will differ from yours, of course. But it can help.

Research apps that are similar, AND some that are not, then pinch one small idea from each one!

Share Improve this answer

answered Dec 2, 2009 at 16:01

Follow

community wiki  
[Magnus Smith](#)



-1

Hire an expert. Design is something you either have or you don't. It sounds like you (just like me) are in the "don't" category :)



Share Improve this answer

answered [Nov 10, 2009 at 21:30](#)

Follow



community wiki  
[Andrew Hare](#)

As a graphic designer I can tell you that the "It" you are referring to is thousands of hours of study and practice. It is not magic. That said, if you don't want to put in at least 1000-2000 hours learning design at a pro level, hire someone who already did. – [willc2](#) Nov 11, 2009 at 1:25



-1

It'll give you an idea of where my head is at that my answer to "How do you normally make a program look beautiful?" starts with:



- Short methods
- Appropriate whitespace
- Consistent naming conventions
- Consistent formatting conventions





It's probably not a good idea to ask me what I think makes for a good UI.

Share Improve this answer

answered [Nov 10, 2009 at 22:56](#)

Follow

community wiki

[Robert Rossney](#)

---