

Usability Engineering

17-313 Spring 2025

Foundations of Software Engineering

<https://cmu-313.github.io>

Michael Hilton, Austin Henley, and Nadia Nahar

Administrivia

- P3B (Final Deliverables) due on Thursday

Smoking Section

- Last full row



Learning Goals

- Identify common examples of bad design, understand why they fail to meet user needs, and learn to critique design choices effectively
- View problems from the user's perspective, understanding their needs, behaviors, and challenges to inform better design decisions
- Learn different methods for assessing usability, including qualitative and quantitative metrics, to gauge the success of a system

What is usability?

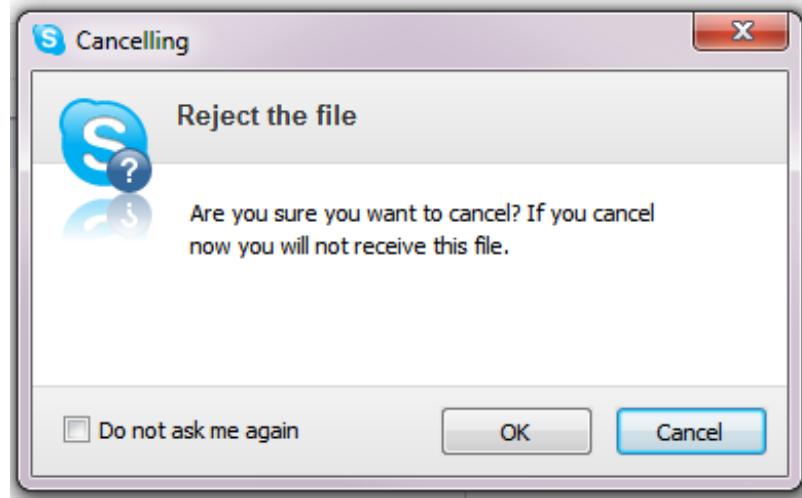
ISO 9241-11: “*the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.*”

*Learnability, Efficiency, Memorability,
Errors, Satisfaction*





- New York overhauled their signs
- Left is before, right is after
- Is it better? Is it good?







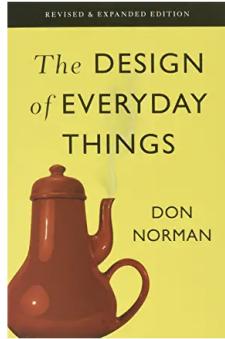




Same handle! Different action?



Best Seller



The Design of Everyday Things: Revised and Expanded Edition

by [Don Norman](#) | Nov 5, 2013

1,434

Paperback

\$14⁴⁹

Great On Kindle: A high quality digital reading

prime FREE Delivery Sat, Sep 12

More Buying Choices

\$9.47 ([59 used & new offers](#))

Audible Audiobook

\$0⁰⁰ ~~\$28.50~~

Free with Audible trial

Kindle

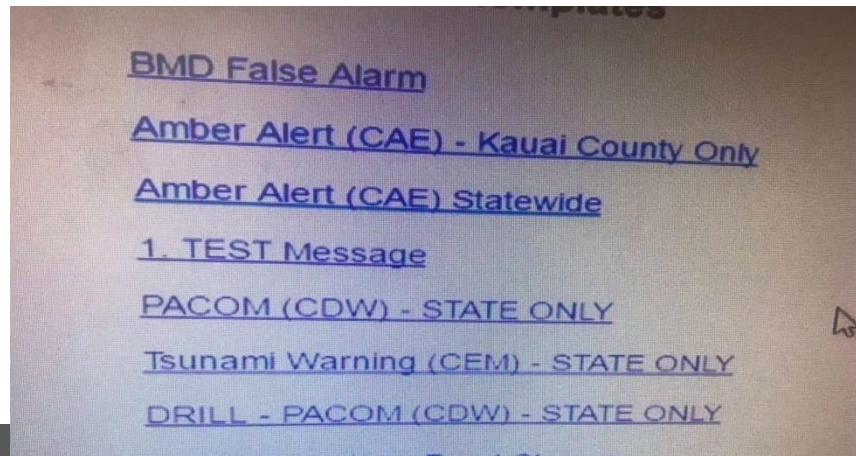
\$11⁹⁹ ~~\$18.99~~

Plus, earn a \$3.00 credit.

Available instantly

#1 book I would recommend to any student

Ok, innocent enough.
But can anything *bad* happen?





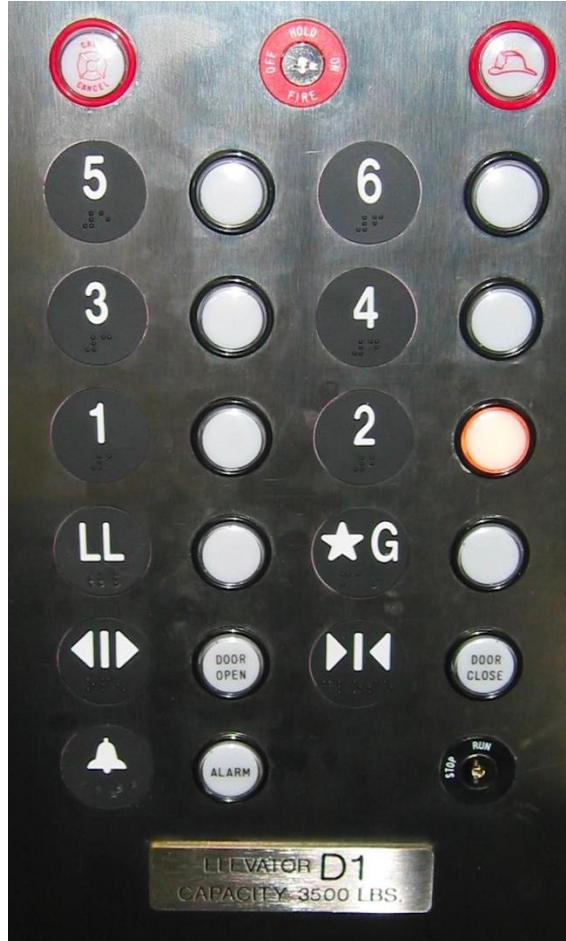
Therac-25

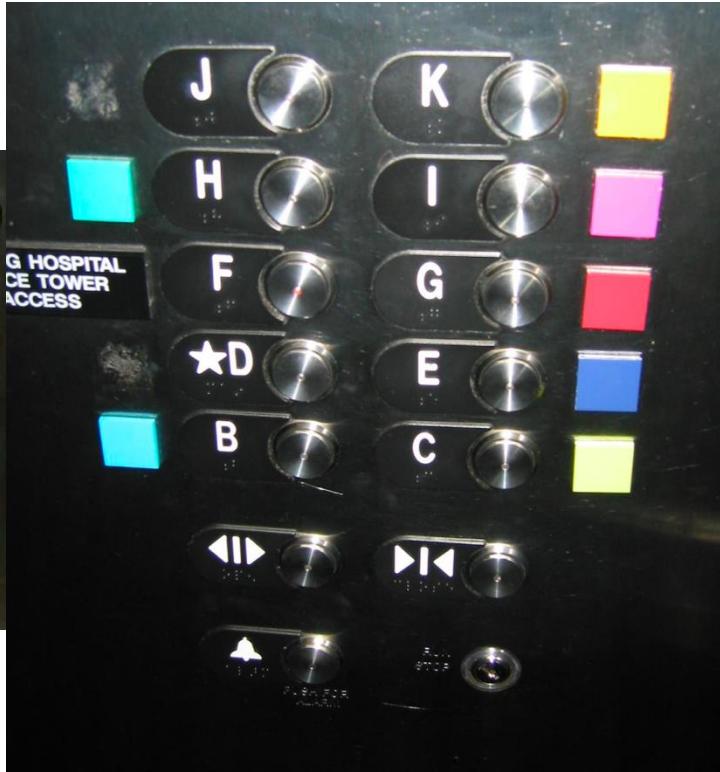
Trump announces all Boeing 737 Max jets are immediately grounded following its 2nd crash in 5 months

Graham Rapier and Benjamin Zhang Mar. 13, 2019, 2:33 PM

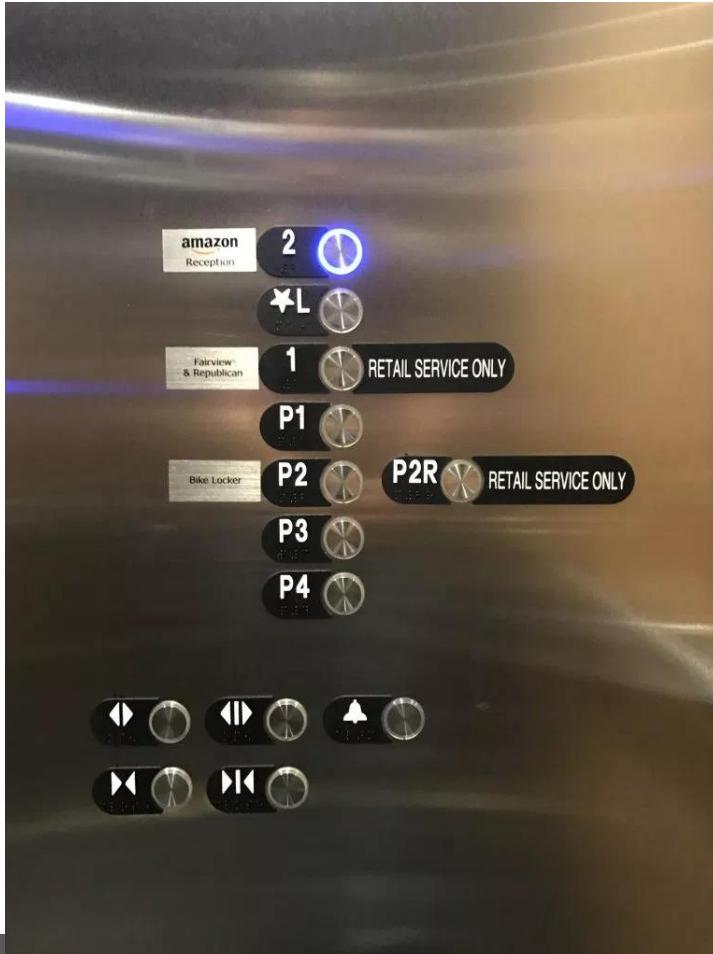


Ever look at elevator button panels?









Did you read the instructions?



Design for people





r/BadUIBattles

For more funny examples of bad design...

What usability problems have you seen with software?

How do we test usability?

Intuition?

Test on ourselves?

Copy interface from other apps?

Not worry about it until someone dies?

Activity: How do you test your feature?

- Who do you have test it?
- What do they do with it?
- What do you measure?
- What is the bar for “good”?

Test with actual USERS

- Put your app in front of actual people
 - Can they figure it out?
 - How long does it take them?
 - Do they need help?
-
- **How they behave != how you want them to behave**
 - Don't blame the user!
 - Blame the design and go fix it!



How do you measure usability?

- Imagine your group project...
- How do you measure whether it is usable?
- How do you test it with people?

Low-cost user testing

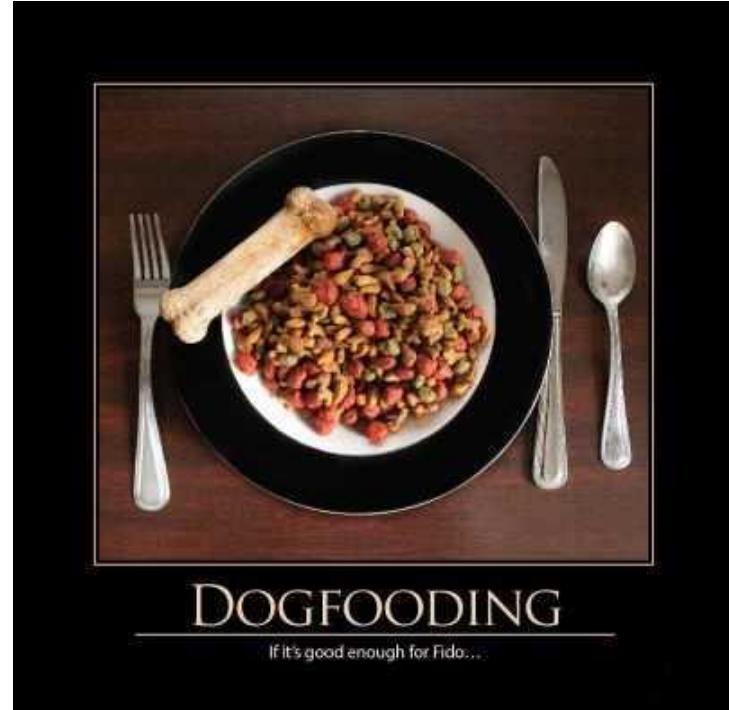
- Ask a few people to try your app every few weeks
- Can they figure it out without help?
- Give them a task with it... do they do it as you expected?



What problems might there be with this method?

“Dogfooding”

- Use your own product.
- Helps close the gap between user and developer.



User study methods

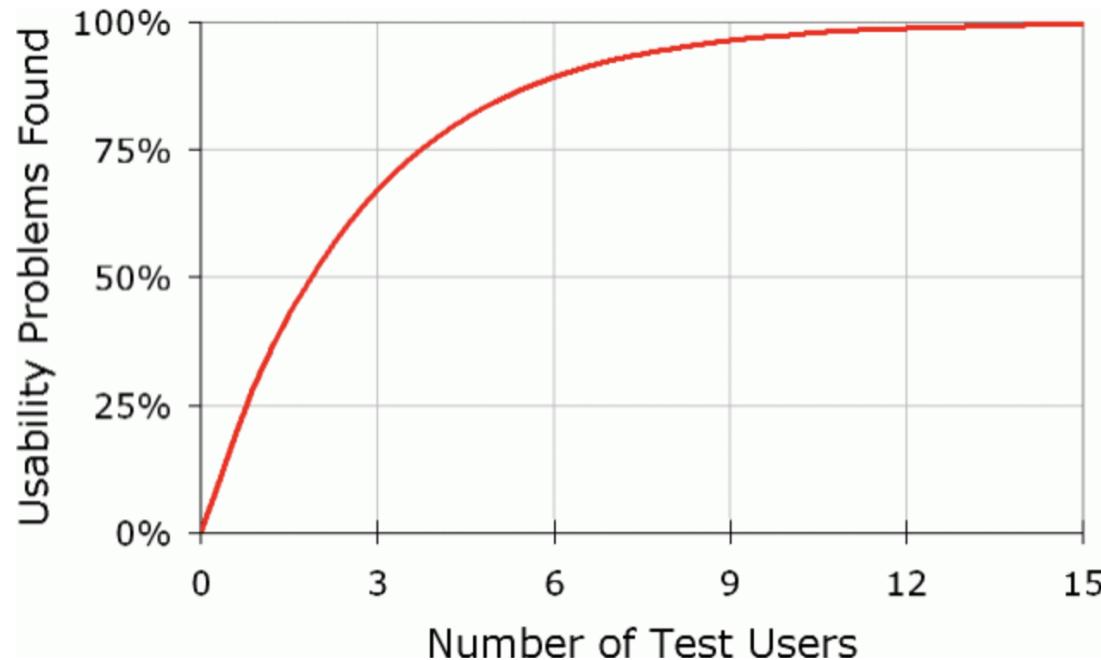
- Interviews
- Surveys
- Shadowing
- Lab study
- Field study
- Wizard of Oz
- Focus group
- Retrospective study
- Diary study
- A/B testing
- Pros/cons of each
- Each answers different questions
- Often use multiple

Lab study

- Controlled setting with complete control over tasks
- Common to record the participant and the screen
- Can be formal or informal
- Many possible tasks
 - Based on your user stories
 - Ask them to complete the same task using two different apps
- Metrics: task time, correctness, mistakes, opinions
- Beware of reactivity effects
- Do they use the app as you expected?



Revealing usability concerns



A/B testing

The screenshot shows the SIMCITY website homepage. At the top, the word "SIMCITY" is displayed in its signature font, followed by "AVAILABLE NOW!" in a large, bold, black font. Below this, there are links for "GAME INFO", "NEWS", "COMMUNITY", and "MEDIA". A prominent "BUY NOW >" button is located on the right. A large banner in the center features the Statue of Liberty against a city skyline with the text "PRE-ORDER AND GET \$20 OFF YOUR NEXT PURCHASE". Below this banner, two purchase options are shown: "SIMCITY™" for \$59.99 and "SIMCITY™ DIGITAL DELUXE EDITION" for \$79.99. Each option includes a "BUY NOW" button and a choice between "PC Download" and "PC Physical". A small note below the digital deluxe edition says "DIGITAL DELUXE EDITION INCLUDES HEROES AND VILLAINS SET".

This screenshot shows a modified version of the SIMCITY website. The main difference is the removal of the "PRE-ORDER AND GET \$20 OFF YOUR NEXT PURCHASE" banner. Instead, it displays two separate purchase options: "SIMCITY™" for \$59.99 and "SIMCITY™ DIGITAL DELUXE EDITION" for \$79.99. Both options include "PC Download" and "PC Physical" options and "BUY NOW" buttons. To the right of these options, a box titled "DIGITAL DELUXE EDITION INCLUDES" lists four items: "HEROES AND VILLAINS SET", "FRENCH CITY SET", "GERMAN CITY SET", and "BRITISH CITY SET". Below this, a section titled "Key Features" is visible.

A/B Testing

- Compare two variations of a design
- Often used for web pages with a call to action
- Randomly show a user design A or B
- Does design A or B lead to more sales/signups/clicks?
- Requires many users
- Is there a *statistically significant difference*?
- Doesn't tell you *why*

Shadowing

- “Fly on the wall”
- Observe people in their own setting... often work
- Rich qualitative data
- Expensive!
- Time consuming
- Difficult to get access



Interview & Survey

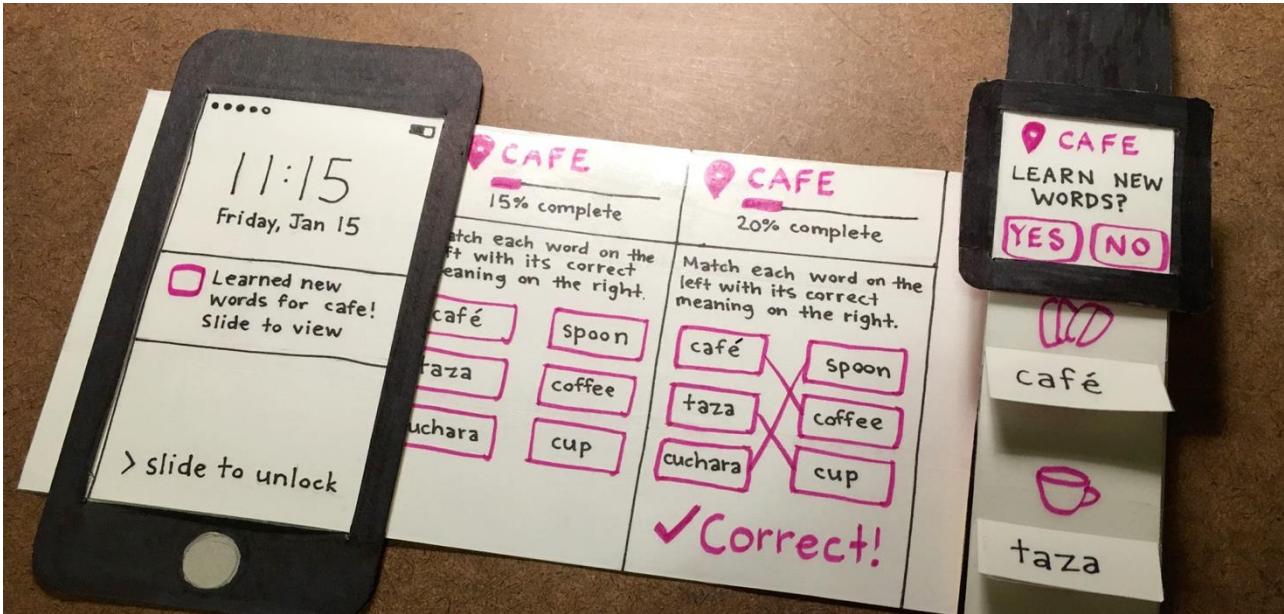
- Open-ended questions show the users perspective
 - Users will surprise you
 - Don't ask leading questions
 - *"How much do you love my app?"*
 - Responses don't always match behavior
 - Scalable
-
- *"Do you want this feature?" "YES!"*



Prototyping

- Implementing a full feature/product is expensive
 - Prototype to gather requirements first
-
- Mockups/wireframes
 - Paper prototype
 - Wizard of Oz

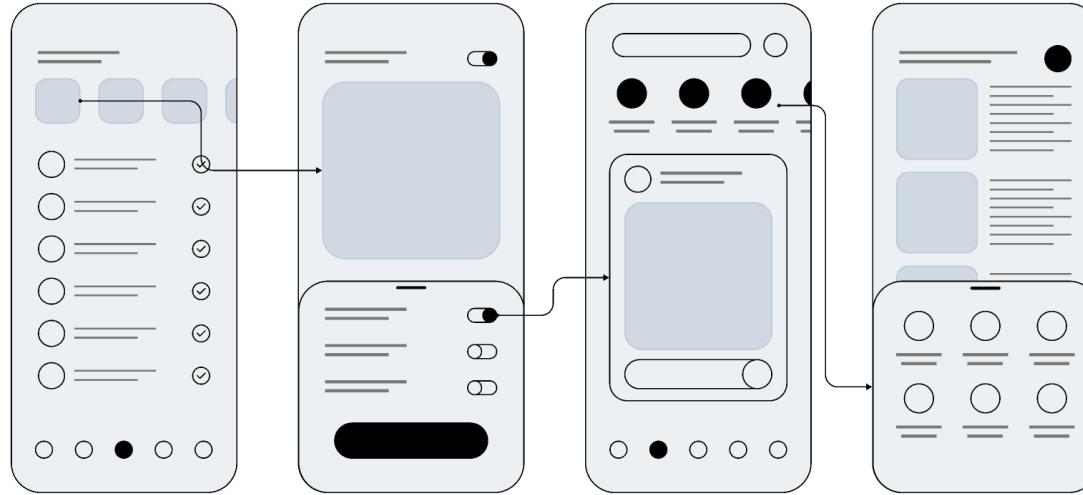
Paper prototype



Wizard of Oz

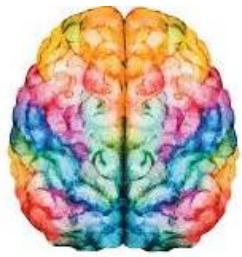


Wireframes



Aspects of Usability

Learnability



Efficiency



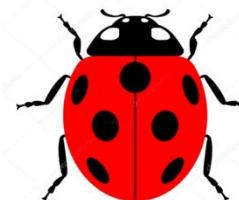
Satisfaction



Memorability



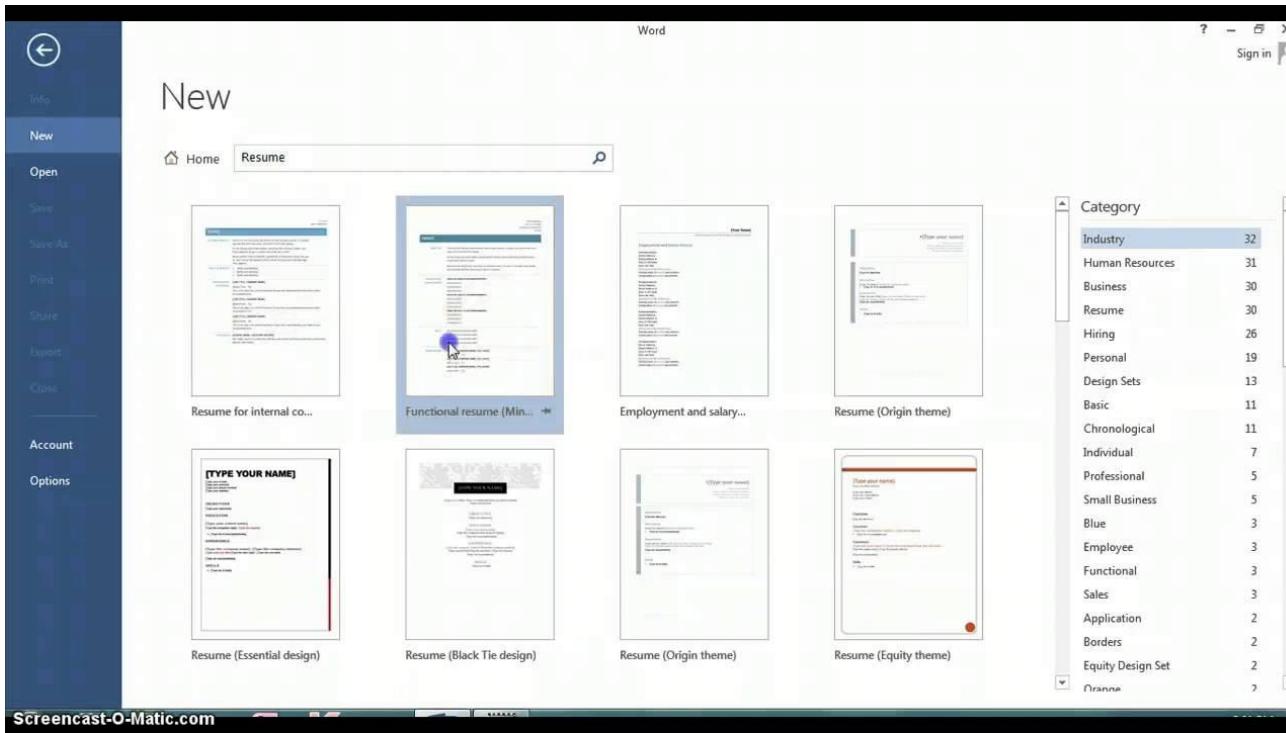
Errors



Learnability

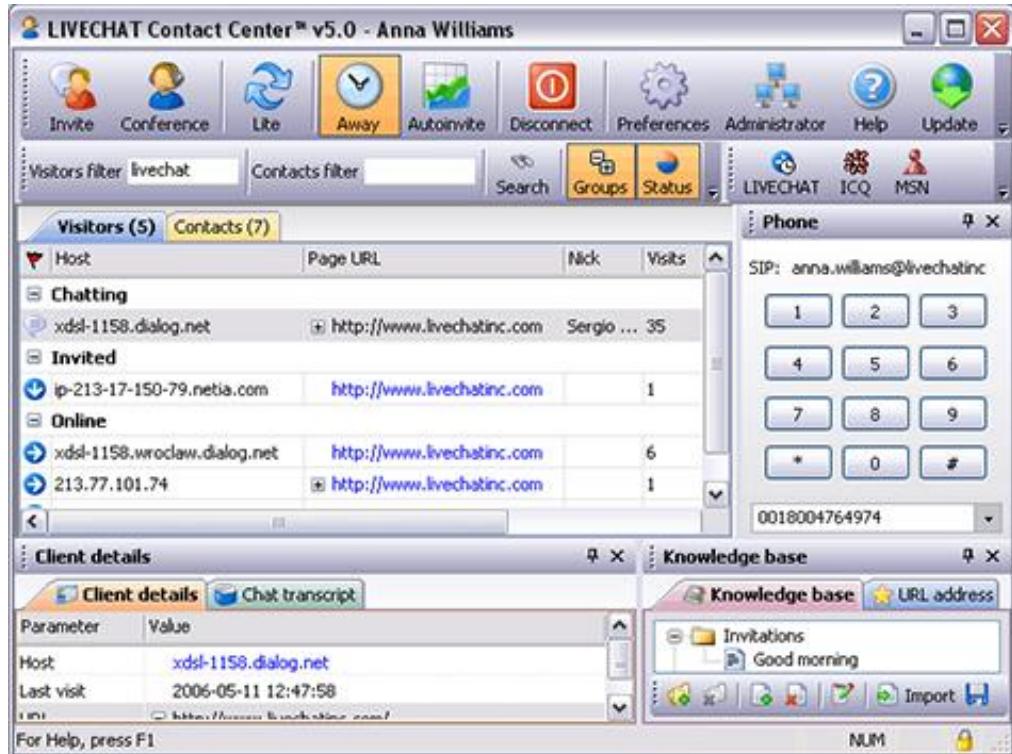
- Can users learn to correctly use it?
 - Can unfamiliar users pick it up?
 - Can power users learn to become experts with it?
- How to provide the right guidance to help users?
- They won't read instruction books!
- Wizards?
- Visible & labeled buttons





Design constraint:

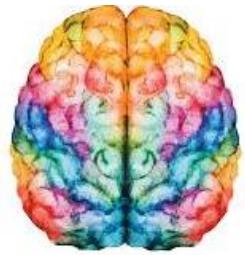
- Don't overwhelm the user
- Make it easy to find features



How could we measure learnability?

Aspects of Usability

Learnability



Efficiency



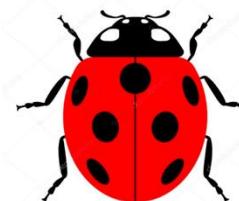
Satisfaction



Memorability

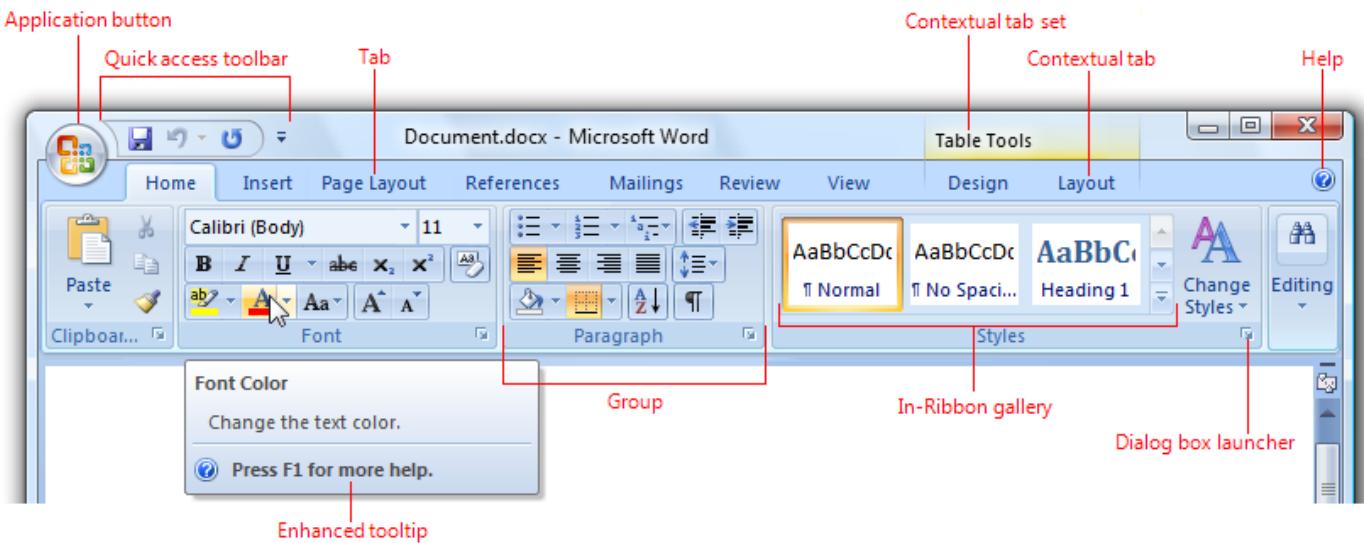


Errors



Efficiency

- How fast can users perform the correct interaction?
- How do you design interfaces for faster interactions?



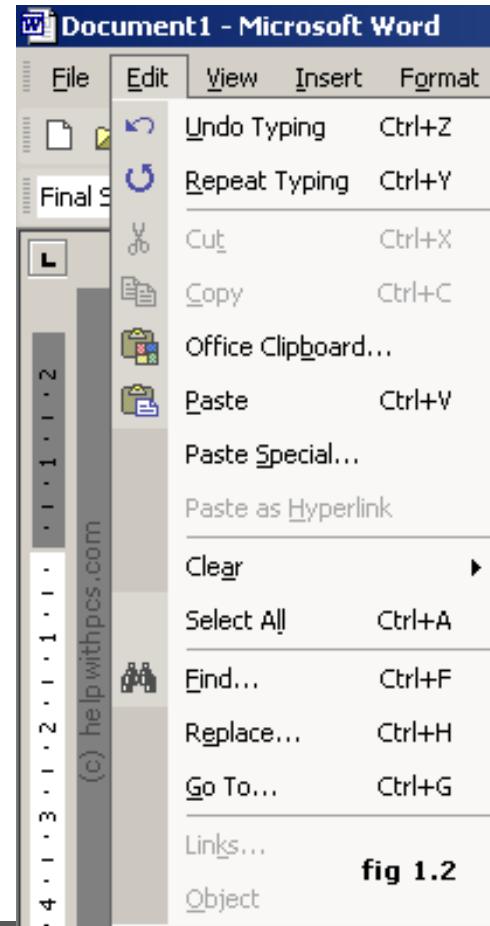


fig 1.2

```
write -- zsh -- 80x24
drwxr-xr-x  59 azh  staff   1.8K Jan 21 12:56 src
drwxr-xr-x  53 azh  staff   1.7K Jan 21 12:56 test
drwxr-xr-x   3 azh  staff    96B Jan 21 12:56 types
-rw-r--r--   1 azh  staff   1.9K Jan 21 12:56 webpack.common.js
-rw-r--r--   1 azh  staff  195B Jan 21 12:56 webpack.dev.js
-rw-r--r--   1 azh  staff  397B Jan 21 12:56 webpack.installer.js
-rw-r--r--   1 azh  staff  472B Jan 21 12:56 webpack.prod.js
[~/Coding/NodeBB] λ cd src/routes/write
[~/Coding/NodeBB/src/routes/write] λ ll
total 120
-rw-r--r--   1 azh  staff   1.5K Jan 21 12:56 admin.js
-rw-r--r--   1 azh  staff   2.2K Jan 21 12:56 categories.js
-rw-r--r--   1 azh  staff   4.4K Jan 21 12:56 chats.js
-rw-r--r--   1 azh  staff  993B Jan 21 12:56 files.js
-rw-r--r--   1 azh  staff   1.2K Jan 21 12:56 flags.js
-rw-r--r--   1 azh  staff   2.4K Jan 21 12:56 groups.js
-rw-r--r--   1 azh  staff   2.4K Jan 21 12:56 index.js
-rw-r--r--   1 azh  staff   2.8K Jan 21 12:56 posts.js
-rw-r--r--   1 azh  staff  967B Jan 21 12:56 search.js
-rw-r--r--   1 azh  staff   542B Jan 21 12:56 tags.js
-rw-r--r--   1 azh  staff   3.8K Jan 21 12:56 topics.js
-rw-r--r--   1 azh  staff   4.7K Jan 21 12:56 users.js
-rw-r--r--   1 azh  staff   557B Jan 21 12:56 utilities.js
[~/Coding/NodeBB/src/routes/write] λ
```

The screenshot shows a Microsoft Excel spreadsheet with data from row 1 to 21. The columns are labeled A, C, D, and E. The data includes various investment accounts like 401k, Roth IRA, and Brokerage, along with categories like Checking and HSA. The last two rows show totals for 'TOTAL' and 'LIQUID'.

	A	C	D	E
1	0 Actual Jan 2023	Projected July 2023	Actual July 2023	Projected Jan 2024
2	401k 1	\$33.00	\$44.00	\$71.00
3	401k 2	\$48.00	\$52.00	\$53.00
4	403b	\$0.00	\$0.00	\$0.00
5	Roth IRA	\$67.00	\$73.00	\$99.00
6	RSU/ESPP	\$89.00	\$105.00	\$123.00
7	Brokerage	\$42.00	\$54.00	\$75.00
8	HYSA	\$23.00	\$23.00	\$10.00
9	CD	\$0.00	\$0.00	\$10.00
10	Checking	\$34.00	\$34.00	\$48.00
11	Brokerage 2	\$79.00	\$90.00	\$0.00
12	HSA	\$32.00	\$35.00	\$79.00
13	0	0	0	\$10.00
14	TOTAL	\$830.00	\$950.00	\$980.00
15	LIQUID	\$83.00	\$110.00	\$130.00
16				\$999.00
17				
18				
19				
20				
21				\$163.00

Copilot

Can you add another projection column and calculate each row based on prior years data?

The headers in data range A1:L21 should be in columns, as a single row on top of the data. Ask me how to use the TRANSPOSE() function to switch the format from columns to rows.

AI-generated content may be incorrect

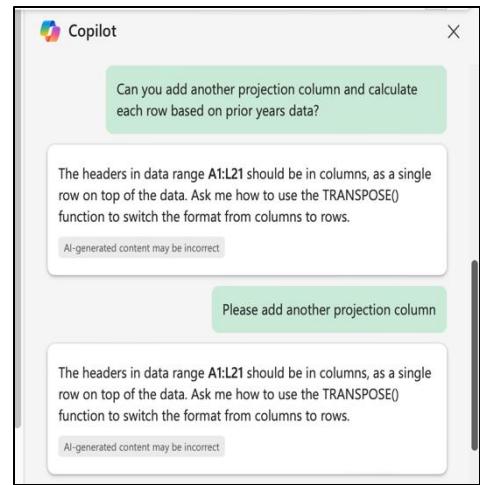
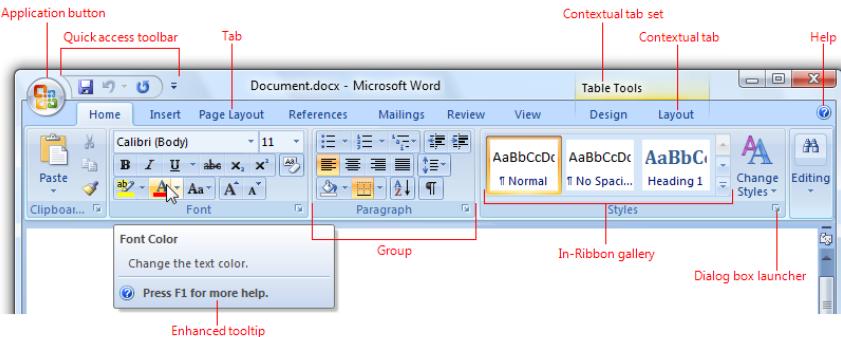
Please add another projection column

The headers in data range A1:L21 should be in columns, as a single row on top of the data. Ask me how to use the TRANSPOSE() function to switch the format from columns to rows.

AI-generated content may be incorrect



```
write --zsh --80x24
drwxr-xr-x 59 azh staff 1.8K Jan 21 12:56 src
drwxr-xr-x 53 azh staff 1.7K Jan 21 12:56 test
drwxr-xr-x 3 azh staff 96B Jan 21 12:56 types
-rw-r--r-- 1 azh staff 1.9K Jan 21 12:56 webpack.common.js
-rw-r--r-- 1 azh staff 195B Jan 21 12:56 webpack.dev.js
-rw-r--r-- 1 azh staff 397B Jan 21 12:56 webpack.installer.js
-rw-r--r-- 1 azh staff 472B Jan 21 12:56 webpack.prod.js
[~/Coding/NodeBB] $ cd src/routes/write
[~/Coding/NodeBB/src/routes/write] $ ll
total 120
-rw-r--r-- 1 azh staff 1.5K Jan 21 12:56 admin.js
-rw-r--r-- 1 azh staff 2.2K Jan 21 12:56 categories.js
-rw-r--r-- 1 azh staff 4.4K Jan 21 12:56 chats.js
-rw-r--r-- 1 azh staff 993B Jan 21 12:56 files.js
-rw-r--r-- 1 azh staff 1.2K Jan 21 12:56 flags.js
-rw-r--r-- 1 azh staff 2.4K Jan 21 12:56 groups.js
-rw-r--r-- 1 azh staff 2.4K Jan 21 12:56 index.js
-rw-r--r-- 1 azh staff 2.8K Jan 21 12:56 posts.js
-rw-r--r-- 1 azh staff 967B Jan 21 12:56 search.js
-rw-r--r-- 1 azh staff 542B Jan 21 12:56 tags.js
-rw-r--r-- 1 azh staff 3.8K Jan 21 12:56 topics.js
-rw-r--r-- 1 azh staff 4.7K Jan 21 12:56 users.js
-rw-r--r-- 1 azh staff 557B Jan 21 12:56 utilities.js
[~/Coding/NodeBB/src/routes/write] $
```



Design constraint:

- Let experts be fast
- Let novices learn



How could we measure efficiency?

Errors

- Is it unlikely that users will make errors?
- Is it easy to recover from errors?



Make it hard to make mistakes

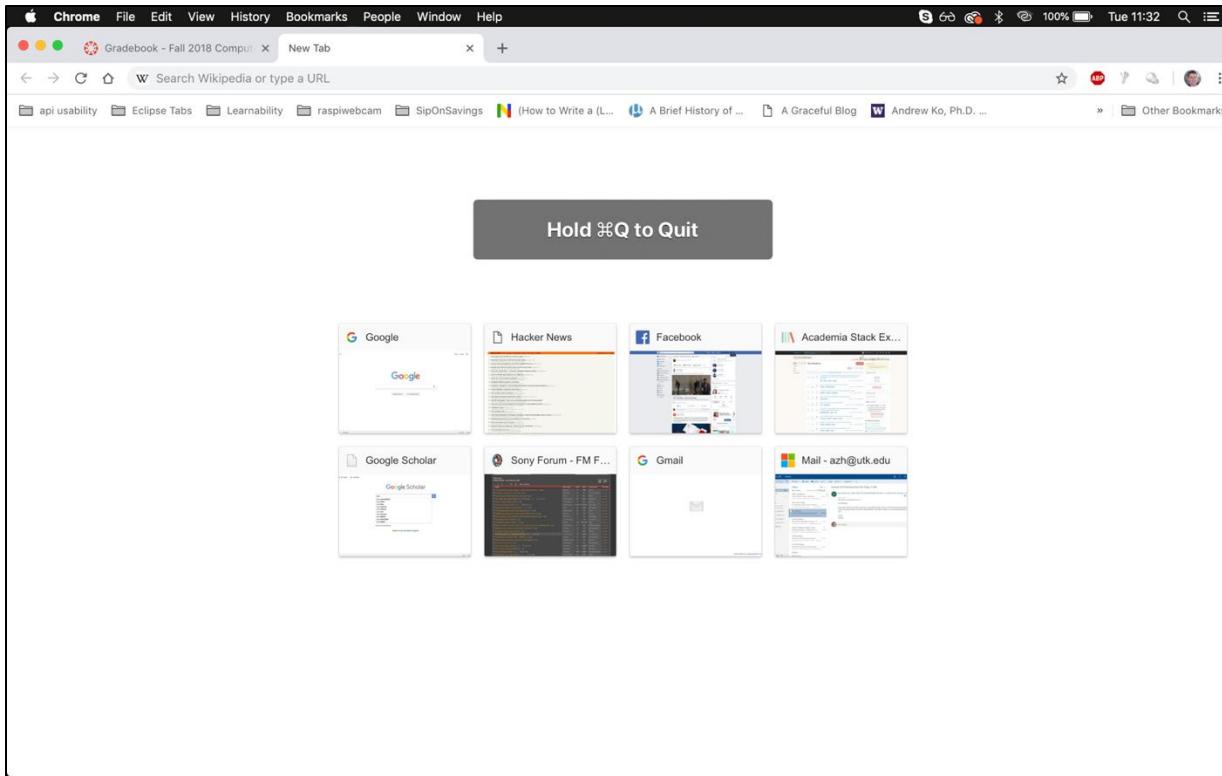
- Unambiguous buttons
- Clear descriptions
- No surprises!
- Every action should be revertible*

Reversible!

- Ever close a tab on accident?
 - Reopen tab
- Ever close all your tabs?
 - Reopen all
- Ever close Chrome?
 - Recent update made it harder
- Delete a file?
 - Trash
- Changed a file?
 - Undo
- Clicked the Next button?
 - Previous



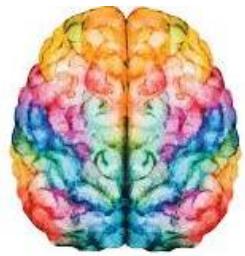
For permanent things...
Ask if they are sure.
Consider a delay.



How could we measure errors?

Aspects of Usability

Learnability



Efficiency



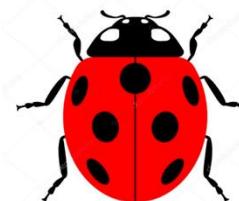
Satisfaction



Memorability



Errors



Activity: How do you test your feature?

- Who do you have test it?
- How do they test it?
- What do you measure?
- What is the bar for “good”?