

GUI Bloopers 2.0: Chapter 6 Interaction Bloopers

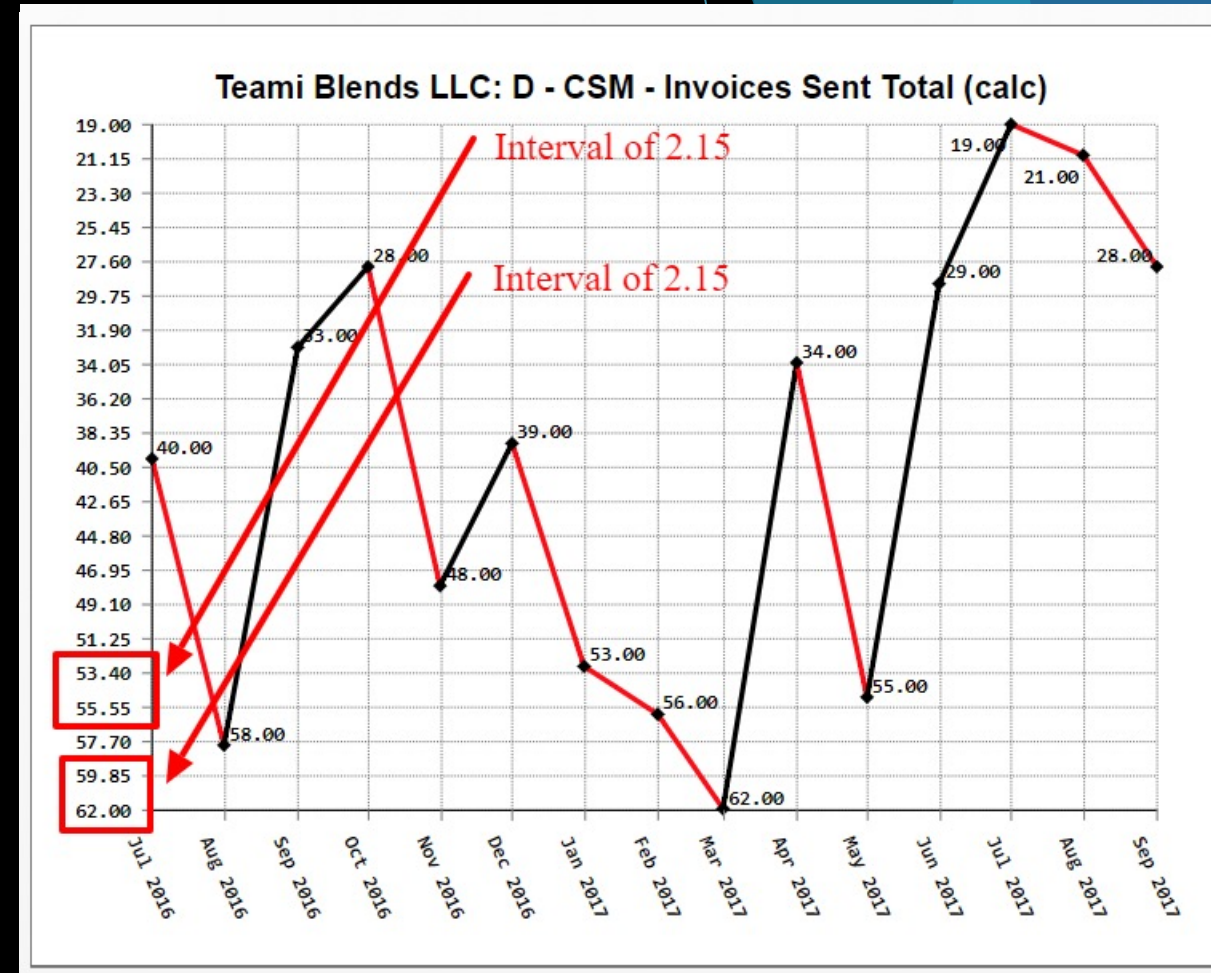
By: Morgan Kaufmann



Alex Delicer

Blooper 40: Exposing the implementation to users

- ▶ Implementation:
 - ▶ In graphs such as the intervals of an axis
 - ▶ Including words or exposing things involved in implementation
 - ▶ That designs an application the way the developer likes
- ▶ How to avoid this:
 - ▶ Focus user interface on a conceptual model that includes objects, actions, and attributes for target tasks
 - ▶ Enforce a product lexicon
 - ▶ Design for the convenience of users



Blooper 41: Needless restrictions

You can only choose 2 to keep

- ▶ Restrictions:
 - ▶ That are unnatural actions, hard to learn, easy to forget, or annoying
 - ▶ That limit text boxes (characters)
 - ▶ That limit username/password length
- ▶ How to avoid this:
 - ▶ Do not impose numerical limits
 - ▶ Allocate data storage dynamically
 - ▶ Use powers of 10 for limits (not 2)



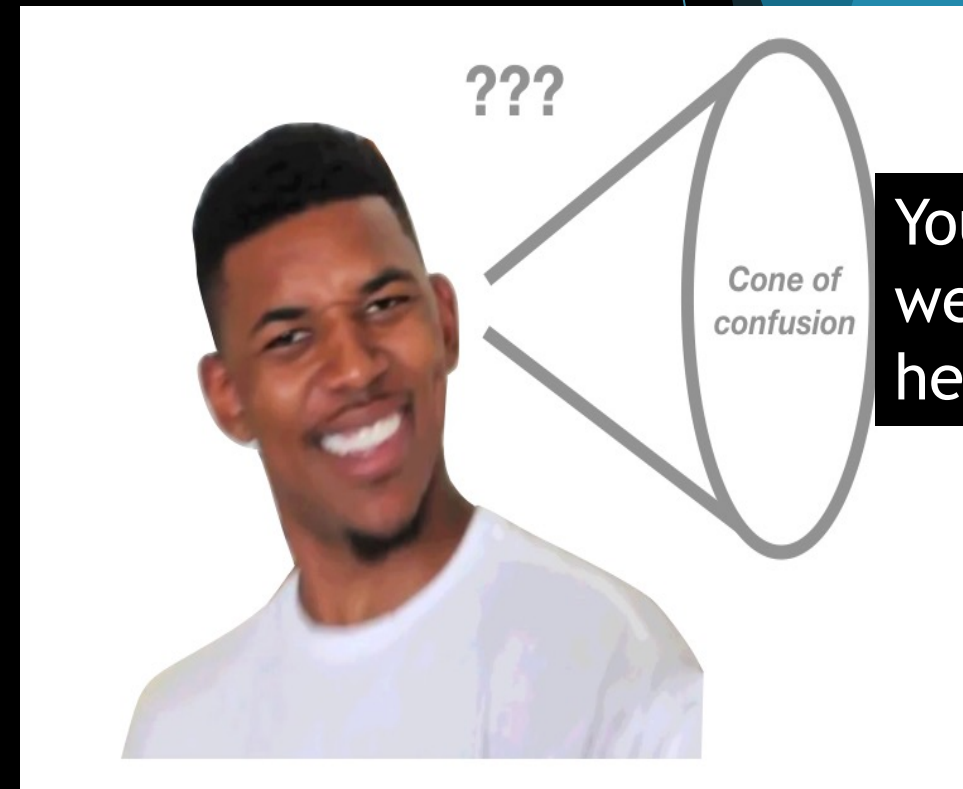
Max 20 char limit

Hi, there|

11 characters remaining

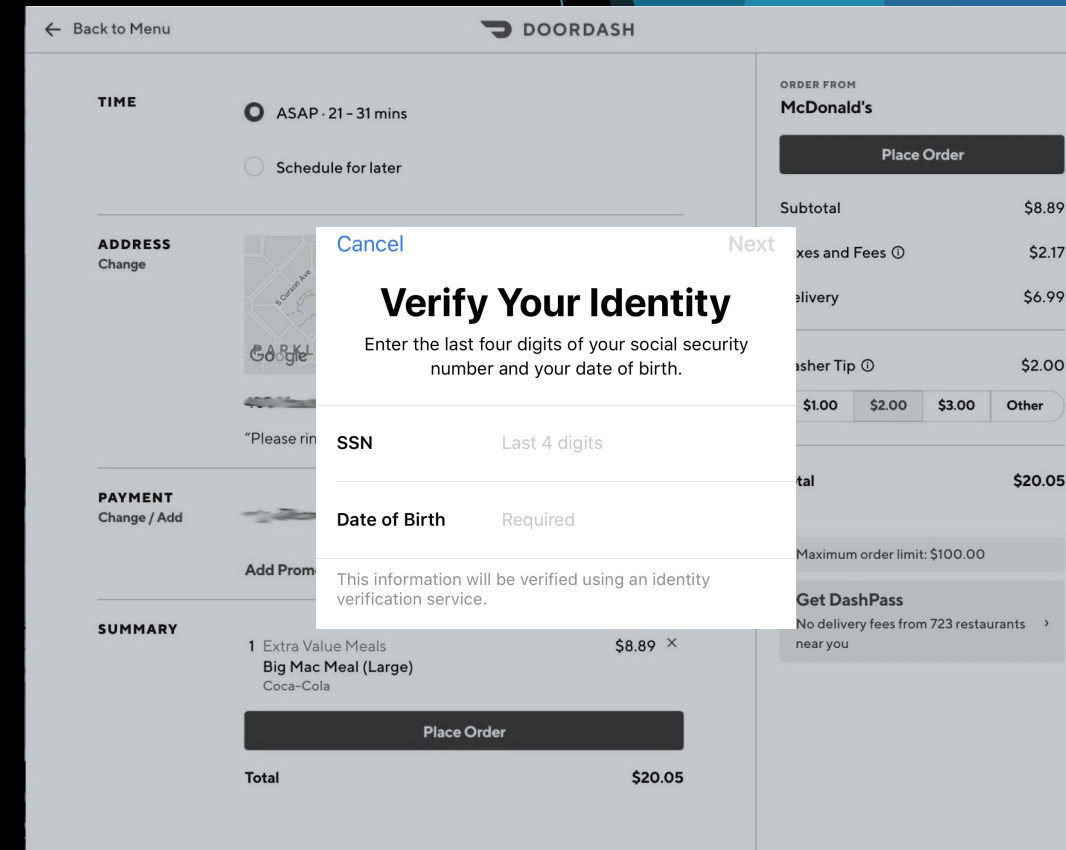
Blooper 42: Confusable concepts

- ▶ Concepts:
 - ▶ Such as Membership, Subscription, Access, and Entitlements
 - ▶ Such as asking users for a location by its information or on a map
 - ▶ Having two different home pages
- ▶ How to avoid this:
 - ▶ Avoid using overlapping concepts (synonyms like log in, sign in)
 - ▶ Combine similar concepts into one



Blooper 43: Asking users for unneeded data

- ▶ Asking:
 - ▶ Users for data after it has already been entered
 - ▶ Users unnecessary questions (data that could be gotten from other data entered)
 - ▶ Users to enter optional data (like comments or phone numbers)
 - ▶ Users to login repeated
- ▶ How to avoid this:
 - ▶ Ask users for necessary data only
 - ▶ Don't make data required unless it is needed
 - ▶ Don't require data that users will not have
 - ▶ Use the data given to you to fill out other data fields
- ▶ (THINK OF A RANDOM NUMBER 1-10)



Blooper 44: Asking users for random seeds

- ▶ Seeds:
 - ▶ That are required to sample data
 - ▶ That are required to run simulations
 - ▶ That create unique keys
 - ▶ That make characters move unpredictably
 - ▶ (Usually “random” numbers are not received)
- ▶ How to avoid this:
 - ▶ Use self seeding method (based on date/time or user-generated events)
 - ▶ Allow users to control only what they need to (repeating simulations or new simulation)
 - ▶ Don't require users to input seeds for in-game opponents



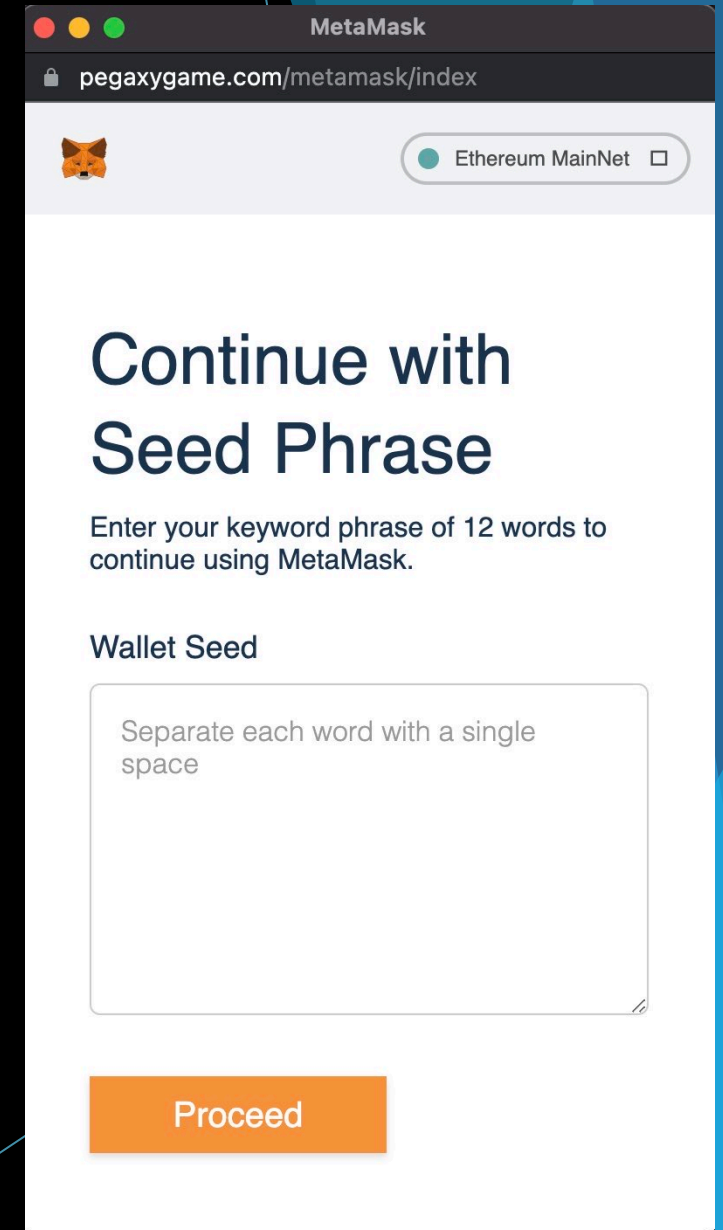
Blooper 45: Pointless choice

- ▶ Choices:
 - ▶ That have no difference from each other
 - ▶ That users do not know
 - ▶ That have an obvious answer
 - ▶ That are false (nonexistent choices, or contain nothing)
- ▶ How to avoid this:
 - ▶ Don't offer similar choices to users
 - ▶ Don't ask questions that users won't understand
 - ▶ Use obvious options instead of asking users
 - ▶ Don't offer choices that are unavailable



Blooper 46: Hard to remember ID

- ▶ ID's:
 - ▶ That are assigned and nonchangeable (passwords as well)
 - ▶ Unreasonable password restrictions
 - ▶ Security questions that don't work
- ▶ How to avoid this:
 - ▶ Allow users to create their own usernames, passwords, and pins
 - ▶ Don't create unnecessary restrictions on user created fields
 - ▶ Allow passwords and PINs (maybe usernames) to be changed
 - ▶ Allow users to get access to their password or PIN when it is forgotten or to reset it
 - ▶ Provide a good set of challenge questions



Blooper 47: Long instructions that go away too soon

- ▶ **Multistep Instructions:**
 - ▶ That are only given at the beginning of a process
 - ▶ That disappear after a certain time
- ▶ **How to avoid this:**
 - ▶ Provide a wizard (multipage dialog box) that includes a step and its instructions for each page
 - ▶ Keep instructions on the screen
 - ▶ Allow users to pull up instructions on every page

Safe Handling Instruction

This product was prepared from inspected and passed meat and/or poultry. Some food products may contain bacteria that could cause illness if the product is mishandled or cooked improperly. For your protection, follow these safe handling instructions.



Keep refrigerated or frozen.
Thaw in refrigerator or microwave.



Keep raw meat and poultry separate from other foods.
Wash working surfaces (including cutting boards),
utensils, and hands after touching raw meat or poultry.



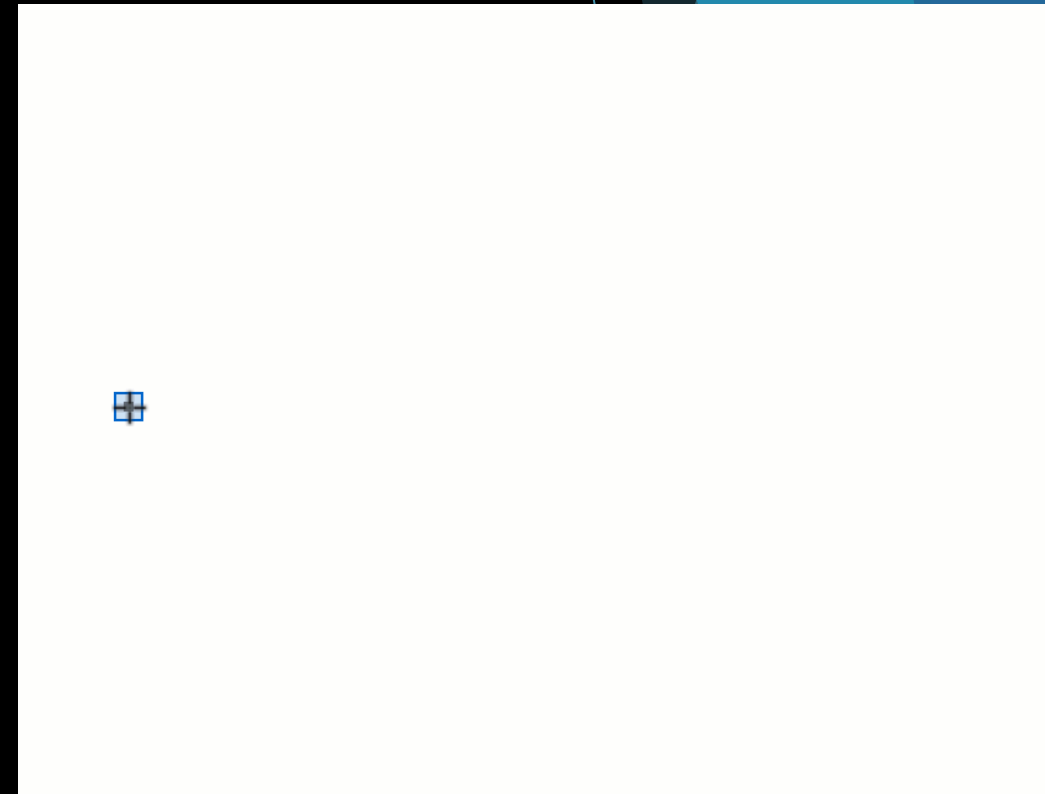
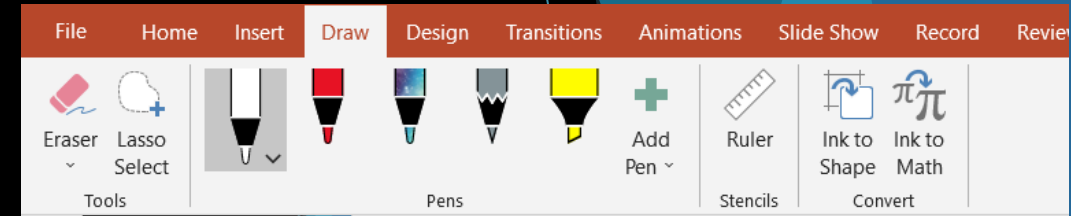
Cook thoroughly.



Keep hot foods hot. Refrigerate leftovers immediately
or discard.

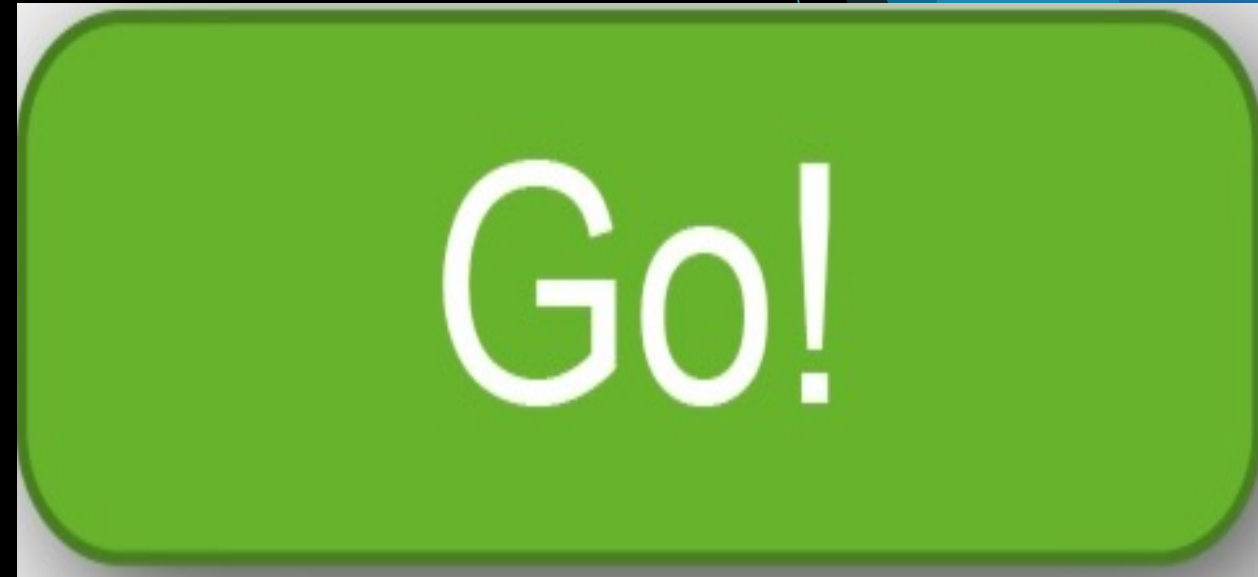
Blooper 48: Unnecessary or poorly marked modes

- ▶ Modes:
 - ▶ Such as a drawing program that locks in a drawing mode (POWERPOINT)
 - ▶ Printing format (landscape or portrait)
 - ▶ They allow users extra actions that have different effects
 - ▶ That are badly named
 - ▶ That cannot be reset
 - ▶ That disappear after they are activated
- ▶ How to avoid this:
 - ▶ Minimize the number of mode settings that an application has
 - ▶ Minimize use/impact of modal dialog boxes
 - ▶ Avoid forced sequences or restrictions on actions
 - ▶ Make mode indicators very visible



Blooper 49: Automatic rearrangement of display

- ▶ EASIEST way to annoy users:
 - ▶ Rearrange their data
 - ▶ Move their controls
 - ▶ Randomly resize windows
 - ▶ Warping their cursor position
 - ▶ Dynamic menus
 - ▶ Dancing tabs



What

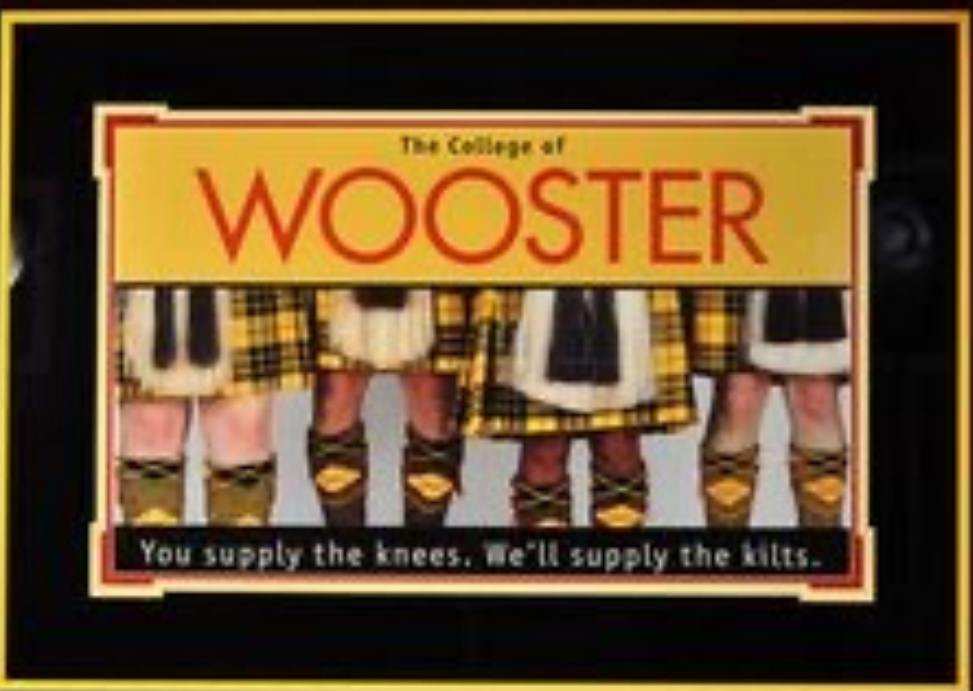
Automatic Rearrangement of

James Mearns

Blooper Reel



Save!



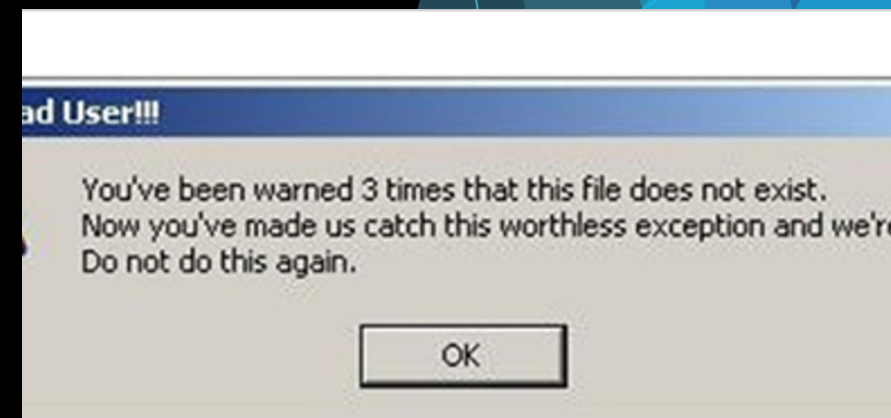
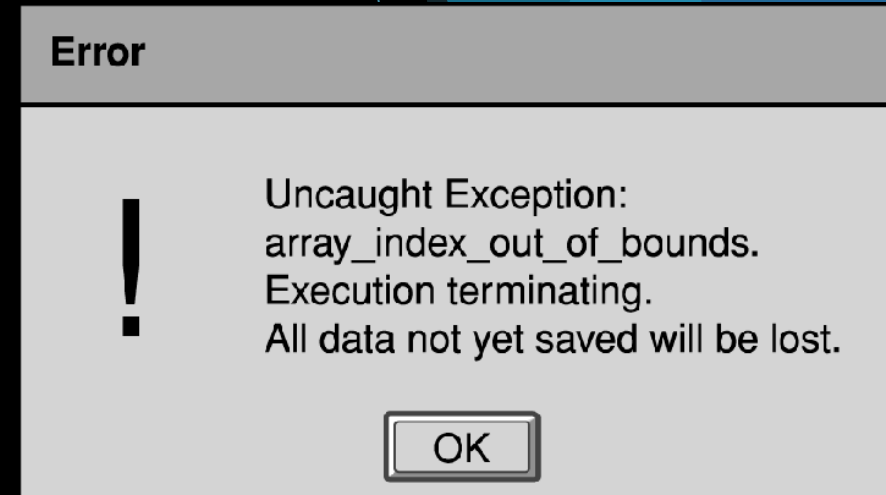
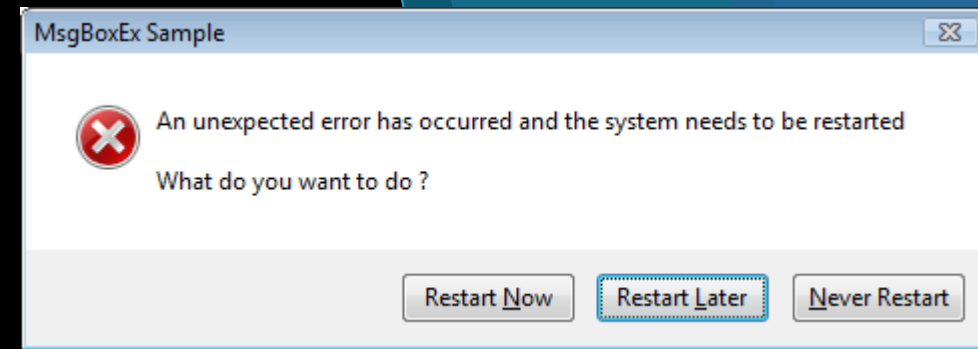
Blooper 49: Automatic rearrangement of display

- ▶ EASIEST way to annoy users:
 - ▶ Rearrange their data
 - ▶ Move their controls
 - ▶ Randomly resize windows
 - ▶ Warping their cursor position
 - ▶ Dynamic menus
 - ▶ Dancing tabs
- ▶ How to avoid this:
 - ▶ Allow the user to control the screen
 - ▶ Preserve display inertia (change as little as possible)



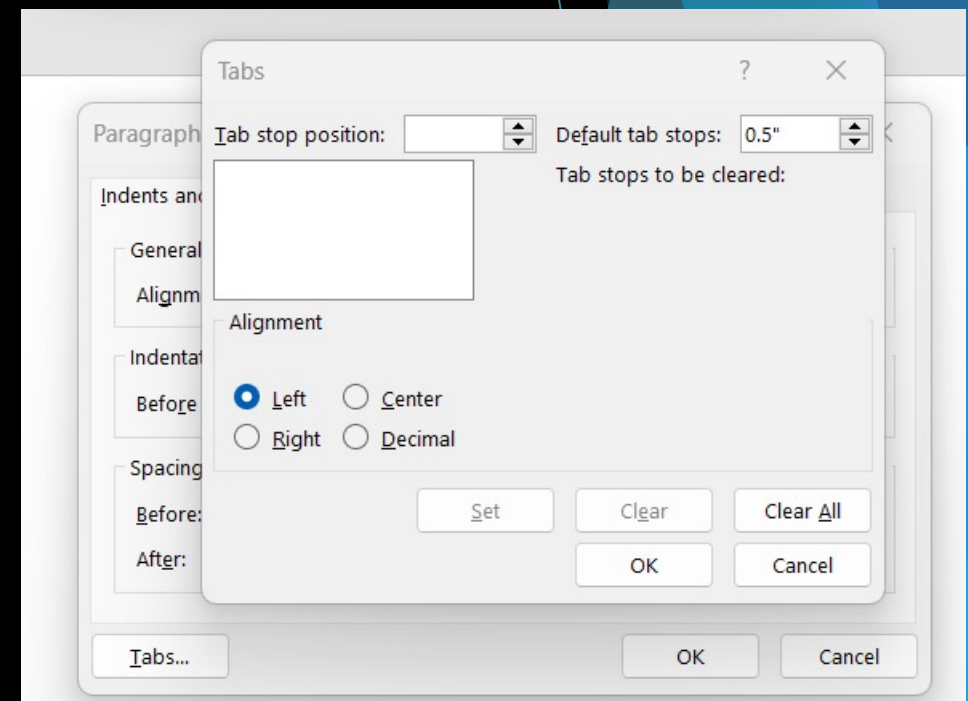
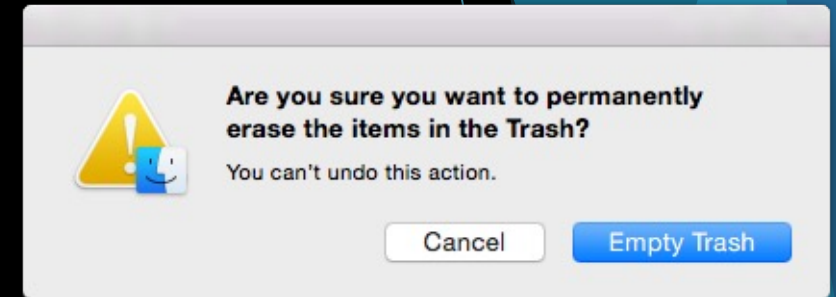
Blooper 50: Dialog boxes that trap users

- ▶ Dialog boxes:
 - ▶ With no “Cancel” button
 - ▶ With all button paths wrong
 - ▶ With required button inactive
 - ▶ With unclear button choices
 - ▶ With only “OK” button against user’s wishes
- ▶ How to avoid this:
 - ▶ Create buttons that would go with user’s goals
 - ▶ Avoid “OK” only situations
 - ▶ Remove inactive or unnecessary buttons



Blooper 51: “Cancel” doesn’t cancel

- ▶ Dialog boxes:
 - ▶ Directly edit application data instead of a copy of it
 - ▶ Changing between tabbed panels saves changes
 - ▶ Wizards (multipage) that don’t cancel everything
 - ▶ Multilevel dialog boxes (that initiate changes early)
- ▶ How to avoid this:
 - ▶ Prevent changes until “OK”/”Apply” is clicked
 - ▶ “Cancel” button cancels everything back to last “OK”/”Apply” click
 - ▶ Do not apply any changes from multilevel dialog boxes until parent dialog box is dismissed



Thank you

ALEX DELICER