



Human Error

Learning Goals

- Understand ...
 - When and how errors should be communicated
 - How human error and design are not independent
 - The difference between mistakes and slips
 - The concept of constraints and how they can help to reduce errors
- Be able to ...
 - explain the assumptions that are made about what errors users make
 - discuss different types of slips and give examples
 - Discuss how a user interface designs can be improved to prevent errors

Communicating Systems Errors

- What to do, if an error in the system occurs?
 - Will the user benefit from knowing about the error?
 - Can the user do something about the error?
 - What other solutions are available?
-
- If the error is provided to the user it must be
 - Understandable (the user gets what the problem is)
 - Actionable (the user gets options to do things)



```
public final int STARTSTOP = 05;  
public final int BACK = 05;  
public final int FORWARD = 06;  
public final int DECREASE = 08;  
public final int DECREASE = 08;
```

integer number too large: 08

Communicating Systems Errors

Swiss International Air Lines AG [CH] https://checkin.swiss.com/ck.fly?locale=de&first_name1=Albrecht&last_name1=Schmidt&departure_port=LUG&departure_date=2016/06/09&flight_number=2903&carrier=LX

BLPFabrizio Montesi

DEUTSCH ▼



Passagierdaten

2

Check-in Angaben

3

Dokumente



Missing: error/CKI_SENDFQTV_BADHOSTRESP
(Ref: 88B1366386 09060001)

Wählen Sie "Ändern" um Ihren Sitz zu wechseln. Falls gewünscht, können Sie hier Ihre Vielfliegernummer hinzufügen/ändern.

Bei dieser Route ist es nicht möglich, die Gepäcketiketten schon zu Hause auszudrucken.

Flug	Von	Nach	Abflug	Klasse
LX 2903	Lugano (LUG)	Zürich (ZRH)	Jun 9, 2016 - 07:15 / pünktlich	Economy

Passagier	Gepäcketikette i	Sitzplatz	Vielfliegernummer
SCHMIDT / ALBRECHT PROF	0	16C	Miles & More 992002087763286
<div>Ändern ✎</div>			

Flug	Von	Nach	Abflug	Klasse
LX 1176	Zürich (ZRH)	Stuttgart (STR)	Jun 9, 2016 - 09:35 / pünktlich	Economy

Communicating Systems Errors

Swiss International Air Lines AG [CH] https://checkin.swiss.com/ck.fly?locale=de&first_name1=Albrecht&last_name1=Schmidt&departure_port=LUG&departure_date=2016/06/09&flight_number=2903&carrier=LX

BLPFabrizio Montesi

DEUTSCH ▼

 **SWISS** | Online Check-in



Passagierdaten



Check-in Angaben



Dokumente



Missing: error/CKI_SENDFQTV_BADHOSTRESP
(Ref: 88B1366386 09060001)



Missing: error/CKI_SENDFQTV_BADHOSTRESP
(Ref: 88B1366386 09060001)

Flug	Von	Nach	Abflug	Klasse
LX 1176	Zürich (ZRH)	Stuttgart (STR)	Jun 9, 2016 - 09:35 / pünktlich	Economy

**Who's fault is it,
if an accident happens**



By Hellerick

Human Error as the Ultimate Explanation?

Deadly crash on German monorail

Twenty-three people died and 10 were injured when an elevated magnetic train ploughed into a maintenance vehicle in north-western Germany.

The train, which floats on a monorail via a magnetic levitation system called maglev, was going at nearly 200km/h (120 mph) when it crashed near Lathen.

[...]

'Human error'

The maintenance vehicle hit by the train had two crew members.

A spokesman for IABG, the company which operates the train, said the accident had been caused by **human error, rather than a technical fault.**



Rescuers had to use ladders and cranes to reach the train

Bei der Analyse der Unfallursachen stützt sich der Bericht laut «Nordwest-Zeitung» auf zwei Gutachten zu dem Unglück:

Nach Ansicht der Gutachter verstieß der Fahrdienstleiter gegen die Betriebsvorschriften, weil er die elektronische Streckensperre nicht setzte.

Als weitere Ursache wird die Missachtung des Vier-Augen-Prinzips im Leitstand der Teststrecke genannt.

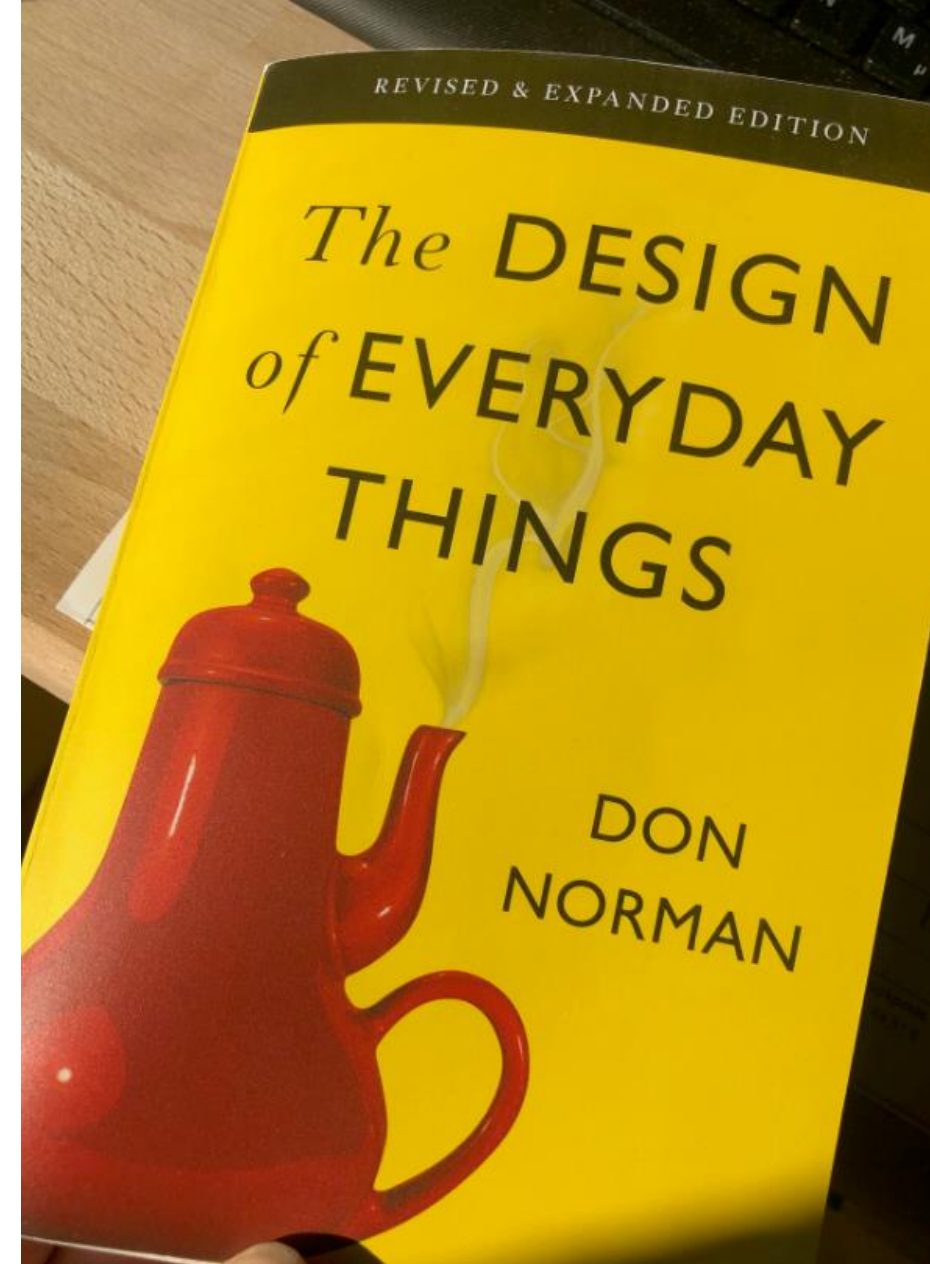
<http://www.netzeitung.de/politik/deutschland/720674.html>

<http://news.bbc.co.uk/1/hi/world/europe/5370564.stm>

About (Human) Errors...

... and implications for user interface design

- “If an error is possible, someone will make it” (Norman)
- “Human Error” are a starting point to look for design problems.



Norman, D. A. (2013). The design of everyday things: Revised and expanded edition. New York: Doubleday.

About (Human) Errors...

... and implications for user interface design

- Design implications
 - Assume **all possible errors will be made**
 - **Minimize the chance** to make errors (constraints)
 - **Minimize the effect** that errors have (is difficult!)
 - Include mechanism to **detect errors**
 - Attempt to make actions **reversible**
- Prevent that users make errors in the first place
 - Make it **impossible to enter wrong commands**
 - Ensure that users **can always recover**

Shneiderman, B., Plaisant, C., Cohen, M., Jacobs, S., Elmqvist, N., & Diakopoulos, N. (2016). Designing the user interface: strategies for effective human-computer interaction. Pearson. <http://www.cs.umd.edu/hcil/DTUI6/>

Norman, D. A. (2013). The design of everyday things: Revised and expanded edition. New York: Doubleday.

Understanding Errors

- Errors are routinely made
 - Communication and language is used between people to clarify – more often than one imagines
 - Common understanding of goals and intentions between people helps to overcome errors
- Two fundamental categories
 - Mistakes = wrong goal
 - overgeneralization
 - wrong conclusions
 - Slips = right goal but wrong action
 - Result of “automatic” behaviour
 - Appropriate goal but performance/action is wrong

Norman, D. A. (2013). The design of everyday things: Revised and expanded edition. New York: Doubleday.

Understanding Errors

- **Errors are routinely made**
 - Communication and **language** is used between people to **clarify** – more often than one imagines
 - **Common understanding** of goals and intentions between people helps to overcome errors
- Two fundamental categories
 - **Mistakes = wrong goal**
 - overgeneralization
 - wrong conclusions
 - **Slips = right goal but wrong action**
 - Result of “automatic” behaviour
 - Appropriate goal but performance/action is wrong

Norman, D. A. (2013). The design of everyday things: Revised and expanded edition. New York: Doubleday.

Understanding the types of Slips Users Make

- Capture errors
- Description errors
- Data driven errors
- Associate action errors
- Loss-of-Activation error ~ forgetting
- Mode error

Norman, D. A. (2013). The design of everyday things: Revised and expanded edition. New York: Doubleday.

Capture errors

Understanding the types of Slips Users Make

- **Capture errors**
 - Two actions with common start point, the more familiar one captures the unusual (driving to work on Saturday instead of the supermarket)
- Description errors
- Data driven errors
- Associate action errors
- Loss-of-Activation error ~ forgetting
- Mode error

Norman, D. A. (2013). The design of everyday things: Revised and expanded edition. New York: Doubleday.

Description errors

Understanding the types of Slips Users Make

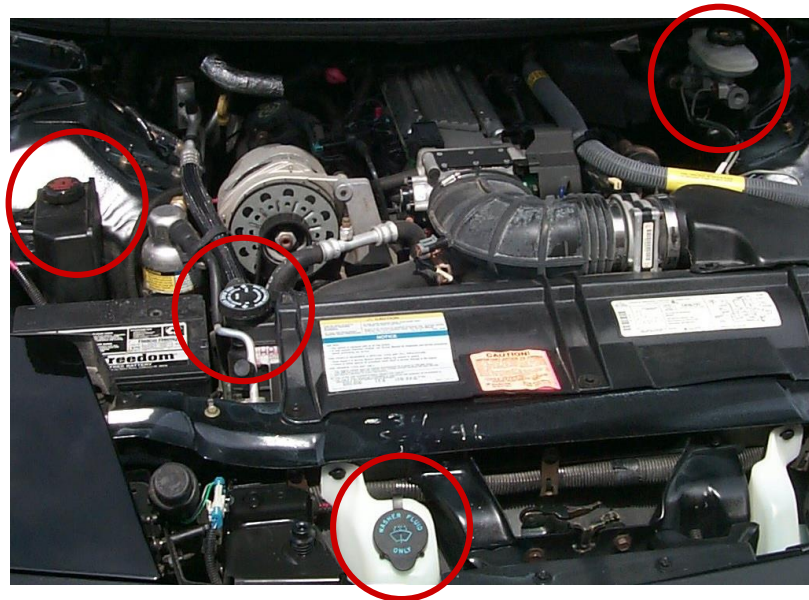
- Capture errors
- **Description errors**
 - Performing an action that is close to the action that one wanted to perform (putting the cutlery in the bin instead of the sink)
- Data driven errors
- Associate action errors
- Loss-of-Activation error ~ forgetting
- Mode error

Norman, D. A. (2013). The design of everyday things: Revised and expanded edition. New York: Doubleday.

Description errors - Example

Understanding the types of Slips Users Make

- Related to Gestalt theory
- Example Car
 - Different openings for fluids, e.g. oil, water, break, ...
 - Openings differ in
 - Size
 - Position
 - Mechanism to open
 - Color
- Design recommendations
 - Make controls for different actions look different



print

save

send

off

print

save

send

off

Norman, D. A. (2013). The design of everyday things: Revised and expanded edition. New York: Doubleday.

Data driven errors

Understanding the types of Slips Users Make

- Capture errors
- Description errors
- **Data driven errors**
 - Using data that is visible in a particular moment instead of the data that is well-known (calling the room number you see instead of the phone number you know by heart)
- Associate action errors
- Loss-of-Activation error ~ forgetting
- Mode error

Norman, D. A. (2013). The design of everyday things: Revised and expanded edition. New York: Doubleday.

Associate action errors

Understanding the types of Slips Users Make

- Capture errors
- Description errors
- Data driven errors
- **Associate action errors**
 - You think of something and that influences your action. (e.g. saying come in after picking up the phone)
- Loss-of-Activation error ~ forgetting
- Mode error

Norman, D. A. (2013). The design of everyday things: Revised and expanded edition. New York: Doubleday.

Loss-of-Activation error ~ forgetting

Understanding the types of Slips Users Make

- Capture errors
- Description errors
- Data driven errors
- Associate action errors
- **Loss-of-Activation error ~ forgetting**
 - In a given environment you decided to do something but when leaving then you forgot what you wanted to do. Going back to the start place you remember.
- Mode error

Norman, D. A. (2013). The design of everyday things: Revised and expanded edition. New York: Doubleday.

Mode error

Understanding the types of Slips Users Make

- Capture errors
- Description errors
- Data driven errors
- Associate action errors
- Loss-of-Activation error ~ forgetting
- **Mode error**
 - You forget that you are in a mode that does not allow a certain action or where a action has a different effect

Norman, D. A. (2013). The design of everyday things: Revised and expanded edition. New York: Doubleday.

Mode error - Example

Understanding the types of Slips Users Make

- Why use modes in the first place?
 - User interface trade-off, e.g.
 - number of buttons needed can be reduced
- Design recommendations
 - Minimize number of modes
 - Make modes always visible
- Example alarm clock
 - Mode vs. mode free
 - Visualization of mode
- What is your solution?
 - Draw the control elements
 - Provide labels



Setting time and alarm
with mode?



Setting time and alarm
without mode?

Correcting Errors

Actions on different level

- If something goes wrong, we attempt corrections on the lowest level
- A task includes action on different levels
 - Drive to University
 - Get into the car
 - Open the car door
 - Insert car key and turn
 - Apply pressure to the key
 - ...

Norman, D. A. (2013). The design of everyday things: Revised and expanded edition. New York: Doubleday.

Preventing Errors

Confirmation is unlikely to prevent Errors

- Example
 - User: “remove the file ‘most-important-work.txt’”
 - computer: “are you sure that you want to remove the file ‘most-important-work.txt’?”
 - User: “yes”
 - Computer: “are you certain?”
 - User: “yes of course”
 - Computer: “the file ‘most-important-work.txt’ has been removed”
 - User: Oops, damm
- The user is not reconsidering the overall action – it only prompts to think about the immediate action (clicking)
- A solution is to make the action reversible

Norman, D. A. (2013). The design of everyday things: Revised and expanded edition. New York: Doubleday.

Detecting Errors

- When “human” errors are detected get into a understandable dialog with the user

Forcing Function

- **Interlock** (e.g. functions can only be done in a certain order)
- **Lock-Ins** (e.g. you can not leave, before you have not done something)
- **Lock-Outs** (e.g. you can get in, before you have not done something)

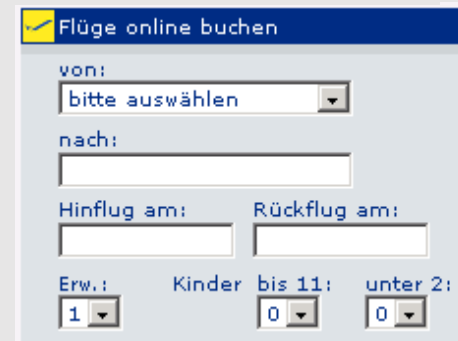
Norman, D. A. (2013). The design of everyday things: Revised and expanded edition. New York: Doubleday.

Constraints to prevent errors

- Physical constraints
 - Basic physical limitations
- Semantic constraints
 - Assumption to create something meaningful
- Cultural constraints
 - Borders and context provided by cultural conventions
- Logical constraints
 - Restrictions due to reasoning
- Applying constraints is a design decision!
 - Practical way to realize the principle “prevent errors”

GUI Example

Date unconstrained



Date constrained



Norman, D. A. (2013). The design of everyday things: Revised and expanded edition. New York: Doubleday.

Sketching a Form


Mini-Exercise: Preventing Errors

- Design a Webform for inputting the following information:
 - Family Name, First Name
 - Country
 - Town, post code, street name and number
 - Email address
 - Gender
 - Birthday incl. year
 - Phone number
- What typical errors do you expect when people fill in the form?
- How to minimize the possibility of errors?
- How to minimize the effect of errors?

What errors do you expect?

Mini-Exercise: How would you prevent them?

Create your Windows Live ID

It gets you into all Windows Live services—and other places you see 
All information is required.

If you use **Hotmail**, **Messenger**, or **Xbox LIVE**, you already have a Windows Live ID.
[Sign in](#)

Use your email address:
[Or get a Windows Live email address](#)

Create a password:
6-character minimum; case sensitive

Retype password:

Phone number:

First name:

Last name:

Country/region:

ZIP code:

Gender: ☐ Male ☐ Female

Birth date:

YAHOO!

Yahoo! | [Help](#)

With a Yahoo! Account, get free email and other leading web services.

Name

Gender

Birthday

Country

Language

Postal Code

Select an ID and password

Yahoo! ID and Email @

Password Password Strength

Re-type Password

In case you forget your ID or password...

Alternate Email (optional)

Secret Question 1

Your Answer

Secret Question 2

Your Answer

What errors do you expect?

Mini-Exercise: How would you prevent them?

```
albrecht@albrecht-VirtualBox: ~/Desktop
GNU Wget 1.16.1, a non-interactive network retriever.
Usage: wget [OPTION]... [URL]...

Mandatory arguments to long options are mandatory for short options too.

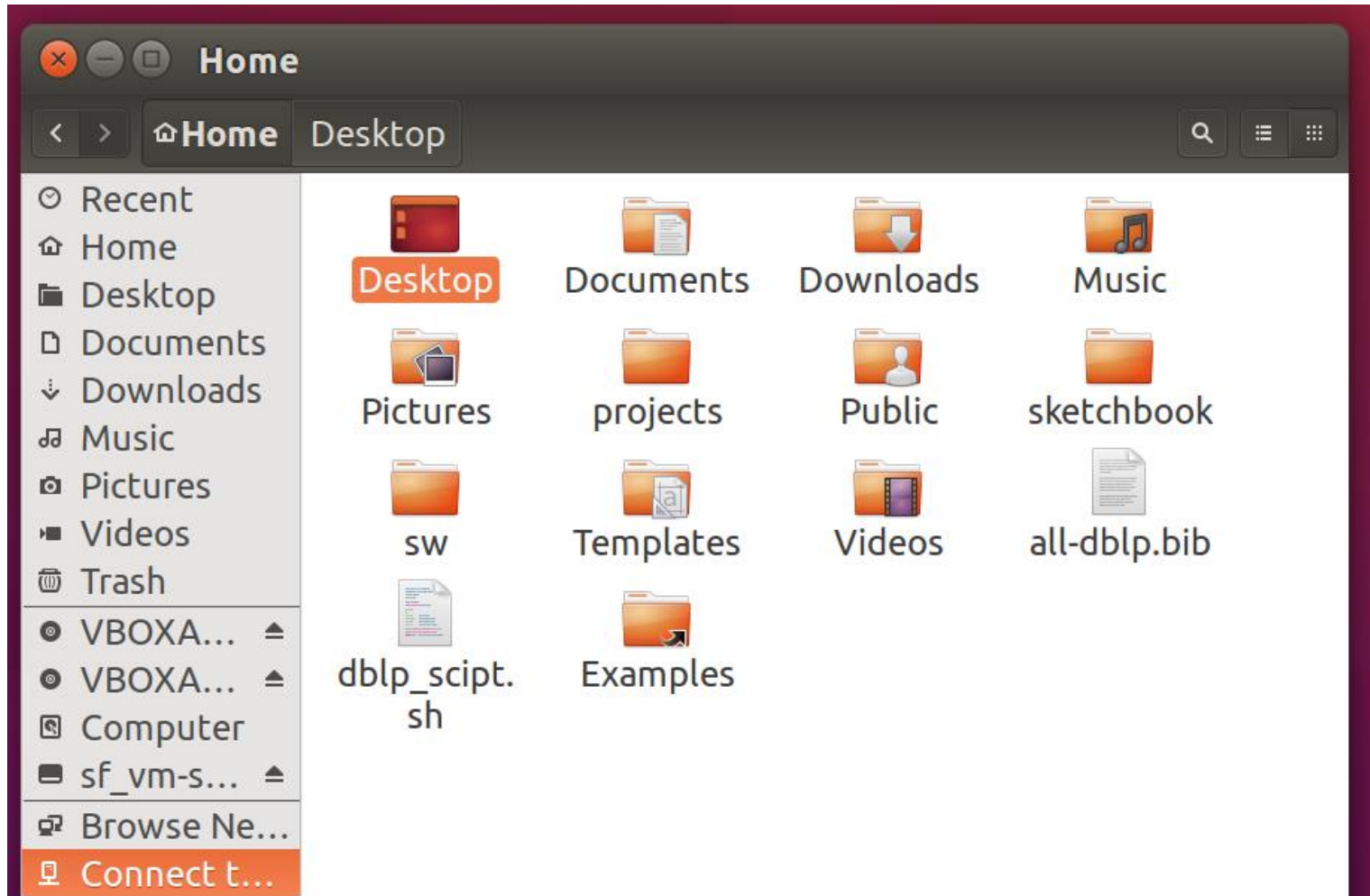
Startup:
  -V, --version           display the version of Wget and exit.
  -h, --help             print this help.
  -b, --background       go to background after startup.
  -e, --execute=COMMAND  execute a '.wgetrc'-style command.

Logging and input file:
  -o, --output-file=FILE  log messages to FILE.
  -a, --append-output=FILE append messages to FILE.
  -d, --debug            print lots of debugging information.
  -q, --quiet            quiet (no output).
  -v, --verbose          be verbose (this is the default).
  -nv, --no-verbose      turn off verbosity, without being quiet.
  --report-speed=TYPE    Output bandwidth as TYPE.  TYPE can be bits.
  -i, --input-file=FILE  download URLs found in local or external FILE
  .
  -F, --force-html       treat input file as HTML.
  -B, --base=URL         resolves HTML input-file links (-i -F)
                        relative to URL.
  --config=FILE          Specify config file to use.
  --no-config            Do not read any config file.

Download:
  -t, --tries=NUMBER     set number of retries to NUMBER (0 unlimits).
                        --retry-connrefused  retry even if connection is refused.
  -O, --output-document=FILE write documents to FILE.
  -nc, --no-clobber      skip downloads that would download to
                        existing files (overwriting them).
```


What errors do you expect?

Mini-Exercise: How would you prevent them?



Learning Goals

- Understand ...
 - When and how errors should be communicated
 - How human error and design are not independent
 - The difference between mistakes and slips
 - The concept of constraints and how they can help to reduce errors
- Be able to ...
 - explain the assumptions that are made about what errors users make
 - discuss different types of slips and give examples
 - Discuss how a user interface designs can be improved to prevent errors



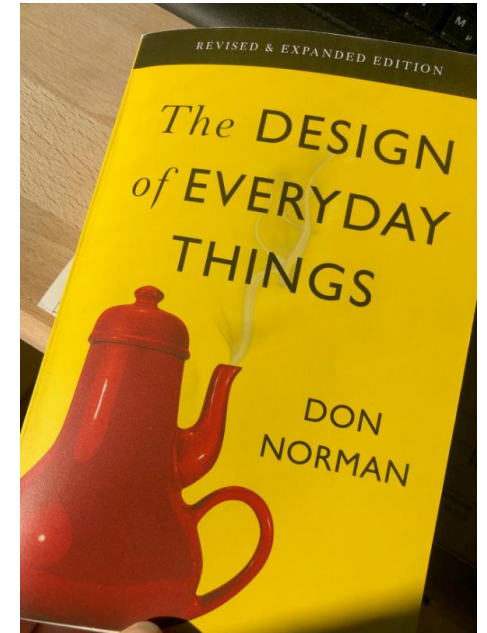
Did you understand this block?

Can you answer these questions?

- When should you not communicate a system error to the user?
- Given a Webform – discuss the statement “All possible errors will be made.”
- Explain the difference between mistakes and slips
- What is a capture error? Give an example.
- What is a data driven error? Give an example.
- Explain physical constraints on the example of a Micro-USB and USB-C connector
- Explain the concept of constraints using the example of a Date-Picker
- Discuss how a user interface designs can be improved to prevent errors

Reference

- Norman, D. A. (2013). The design of everyday things: Revised and expanded edition. New York: Doubleday.
- Shneiderman, B., Plaisant, C., Cohen, M., Jacobs, S., Elmqvist, N., & Diakopoulos, N. (2016). Designing the user interface: strategies for effective human-computer interaction. Pearson. <http://www.cs.umd.edu/hcil/DTUI6/>



This file is licensed under the Creative Commons Attribution-Share Alike 4.0 (CC BY-SA) license:

<https://creativecommons.org/licenses/by-sa/4.0>

Attribution: Albrecht Schmidt

For more content see: <https://hci-lecture.de>

