

# *Human Error? No, Bad Design*

---

The Design of Everyday Things

Author: Don Norman

Presented by: Naod Fisseha

# *Understanding Why there is Error*

---

- Most common reason is poor design
- Design should accommodate to how people behave instead other way around
- Error can occur because of physical or mental limitations

# *Root Cause Analysis*

---

- Root cause is something that sets motion a set of events
- Aim is to identify single cause to an incident
- Limitations are engineers always look ways to sacapegoat human beings for bad design and many things can go wrong instead of one

# *The Five Ways*

---

- A technique that can help adapt to the limitations of root cause analysis
- Five is an arbitrary number, can change depending on the incident

# Example of Five Whys

## Five Whys

Question	Answer
Q1: Why did the plane crash?	Because it was in an uncontrolled dive.
Q2: Why didn't the pilot recover from the dive?	Because the pilot failed to initiate a timely recovery.
Q3: Why was that?	Because he might have been unconscious (or oxygen deprived).
Q4: Why was that?	We don't know. We need to find out.
Etc.	

# *What can cause errors?*

- Tinkering and Experimentation
- Distraction and Pressure



# *Deliberate Violations*

---

- Violation of guidelines made for sake of efficiency or effectiveness
- Sometime can be done during special circumstances
- Out of the scope of the design of everyday things

# *Two Types of Errors*

---

Slips - occur when user performs right actions badly

Mistakes - occur when user fails to perform the right action



# *Slips*

---

Capture Slips: occurs when switching from one repetitive activity to another

Description Similarity Slips: When affordances look alike to a person



# *Slips*

---

Memory-lapse Slips: caused due to lapses in short term memory

Mode Error Slips: occurs when there are different states of operation, but user does not know



# *Mistakes*

---

Can be classified into:

1. Rule based mistakes
2. Knowledge based mistakes
3. Memory-lapse mistakes

# *Mistakes*

---

- **Rule based mistakes** usually happens when a person with little knowledge or experience follows rules written by someone else
- User assumes that the rules should work in all instances
- Can be caused by poor communication, faulty instructions, user disregarding the rule
- Can be avoided by simplifying task, simplifying instruction, consider primary use, handle contingencies, usability tests

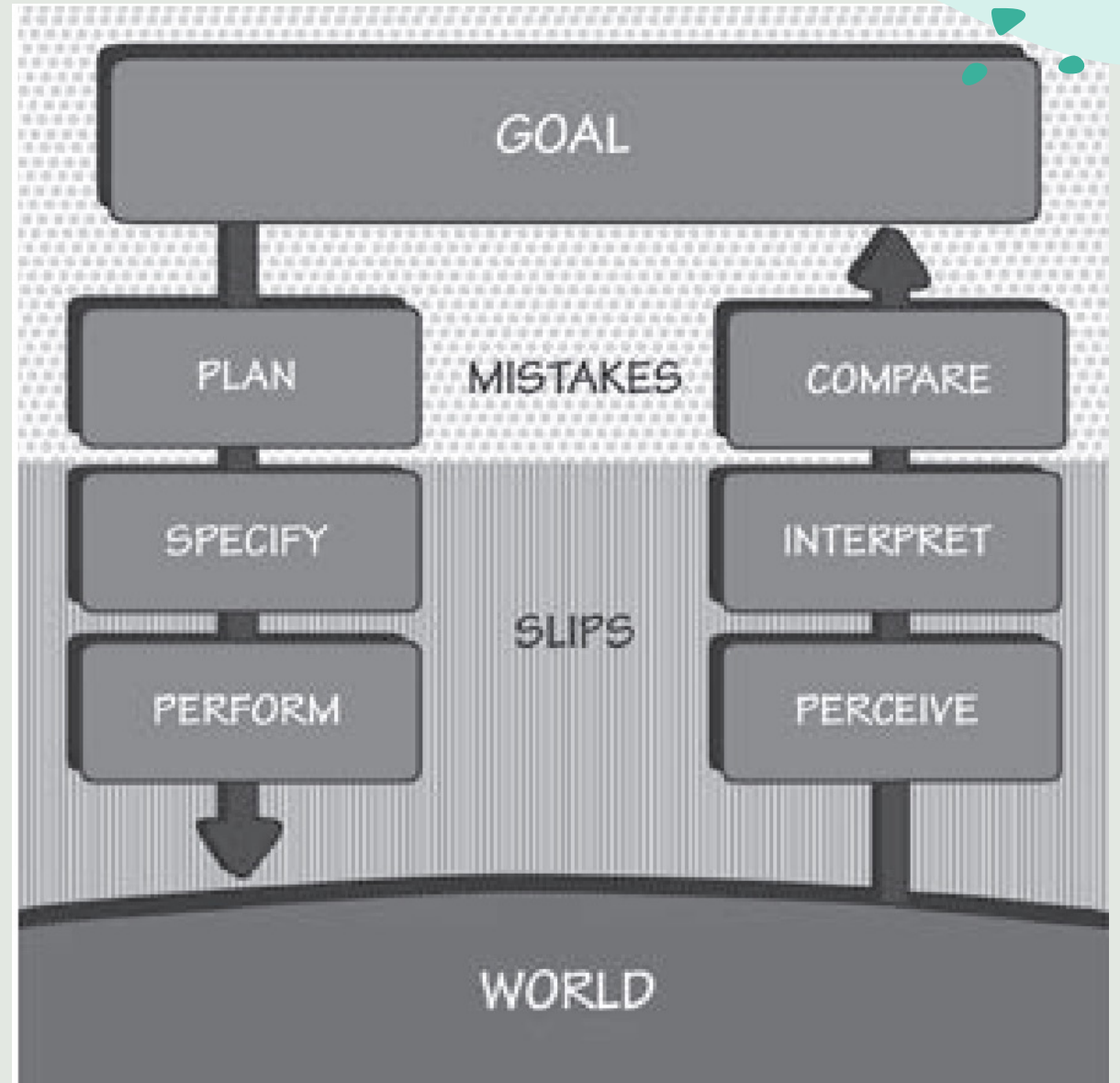
# *Mistakes*

---

- **Knowledge based mistakes** happens when an individual faces a problem they have never encountered
- Therefore, user will use trial and error
- Can be countered by giving knowledge or using knowledge of similar tasks
- **Memory lapse mistakes** occur because lapse of memory during planning
- Similar to memory lapse slips except error happens before action instead of during action

# *Errors and The Seven Stages of Action*

---



# *What to do to avoid errors?*

---

- Reporting Errors
- Detecting Errors
- Designing for Errors

# *Reporting Error*

---

- When errors are recognized, implementations can be made to make changes that decrease errors in the future
- Shift in Cultural Attitude is needed
- Example: "Jidoka"
- Workers can stop assembly line when they think something is wrong



# *Detecting Error*

---

- When errors are detected quickly, it can be fixed to minimize harm
- Immediate feedback is very important

# *Designing for Error*

---

- A good approach is to accommodate for choice a user might make that do not lead to success
- Designing for error should consider the following:
  - Warning and error messages
  - Addressing interruptions
  - Adding constraints to block errors
  - Undo functions
  - Confirmations
  - Sensibility checks

# *When Good Design Is Not Enough*

---

When can we say that error is not the fault of design?