Week 4, Part 1:

Principles of good design

Today we're going to talk about ...

- Guidelines for good design
- Principles of good design

Why learn this stuff?

- Understand the guidelines for design
- Be able to distinguish between bad design and good design
- Follow guidelines and principles derived from social sciences to create a good design

Is this design good or bad?

" I know bad design when I see it."

(But why is that?)

Questions to ask to determine if something is a good design:

How easily can you determine the function of the interface?

How easily can you tell what actions are possible?

How can you determine mapping from intent to physical movement?

How easy is it to perform the action?

How can you tell what state the system is in?

In human computer interaction
we are concerned with

usability.

Defining usability:

How easily can people who are using the interface learn to use it and to achieve their goals, and how satisfied are the with the process?

Two main reasons for dissatisfaction with interfaces:

Engineering aspects:

- The physical design is wrong
- The technology is wrong

Human user aspects:

- It doesn't look good/feel good
- It doesn't work well
- It's frustrating

Design guidelines & principles

What's the difference between guidelines and principles?

Guidelines provide a **set of rules** to follow.

Principles are **design values** that articulate what's important in the process.

Guideline: "Always do thorough stakeholder research and produce a stable set of requirements."

Principle: "The person who uses the thing should be at the centre of our design thinking."
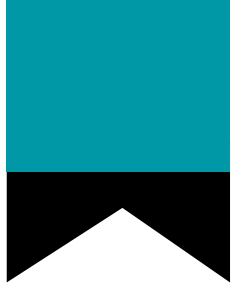
Design guidelines

Guidelines can:

- Contradict each other
- Not cover every possible aspect of development
- Reflect a holistic approach - consider the situation from a very broad perspective
- Are applicable at every stage of the process, from a wide perspective to a very specific one

Poor design is often unprincipled.

(This means that it's obvious the designers did not agree on or understand what's important.)

What are the **guidelines?**

What are the **principles?**

Schneiderman's Principles

1. Recognise diversity
2. Use the golden rules of interface design
3. Prevent errors
4. Follow guidelines for data display
5. Follow guidelines for data entry
6. Balance automated and human control

1. Recognise the diversity.

There are lots of users.

Some are experts, some are novices.

Some use iOS, some use Android.

Some use the thing every day, some use the thing once a year.

This means:

Take their context into account and understand their environment.

Are you asking them to input voice commands in a loud environment? Speak in a quiet environment?

Understand the constraints of the medium of communication.

Are you using a mobile phone? A computer screen? A telephone voice system?

Need to be able to interrupt and return

What do they have to remember next time? Can they come back later?

# How to do it:

Control should mirror real-world

Representations should be consistent with what your user already knows how to do

Express requirements clearly

Use common words - no jargon!

2. Use the Eight Golden Rules of Interface Design

1. Consistency
2. Universal Usability
3. Informative feedback
4. Dialogues with closure
5. Prevent errors
6. Reversal of action
7. Place user in control
8. Reduce short-term memory
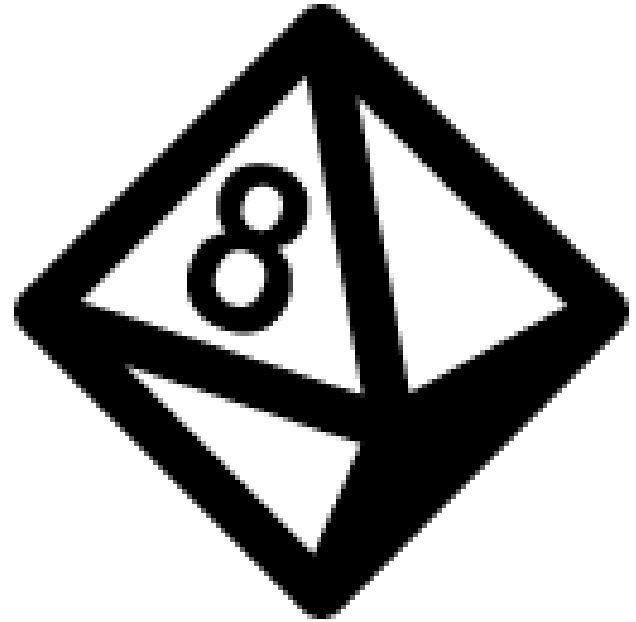
Rule 1: Consistency.

Consistency must exist in ...

Terminology: Identical words and terms for prompts, menus, help screens, interactions

Aesthetics: Colour codes, layouts, fonts

Symbols: Icons and symbols must be consistent

Response: The system must respond to input in the same way every time

Rule 2: Shortcuts.

Super users need shortcuts!

Allow frequent users to develop a clear idea of how the system works, and let them work faster.

Sometimes this takes the form of shortcuts, toolbars, and hotkeys.

Rule 3: Useful feedback

For every action there should be a reaction.

For frequent and minor actions, this can be modest.

For major actions, the response should be substantial!

Rule 4: Closure.

Design interactions to have a beginning, a middle, and an end.

Organise tasks into groups.

Offer feedback for every action ("Next", "Finished" buttons)

Rule 5: Errors happen.

As a designer it's your job to anticipate errors and handle them.

Design the system so massive errors aren't possible.

Build in recovery stages - "Do you really want to delete your entire drive? Type yes if so."

Rule 6: Reversal.

Everyone knows the value of undo.

Can be done with logs - history viewers, or recovering the last thing

Sometimes the thing we want recovered/undone isn't what we ever thought it would be - anticipate

No matter how many times you warn people, they will always manage to do something catastrophic.

Rule 7: Give control.

Users always want to feel in control.

This concept originates from the 1980s, when users went from **responding** to a computer to **initiating** actions.

The person should have control at every point: Stop a print job, stop connecting to a website, ctrl + alt + del

Rule 8: Reduce load.

Reduce the user's short-term memory load.

We can usually remember seven chnks of information. This means:

- Displays need to be simple
- Complexity should be reduced
- Sequences of actions need to be short

3. Prevent Errors

Correct actions: Understand the conditions and context so users don't make errors, and allow them to undo.

Correct commands: Offer command completion

Correct feedback: User should know the system state at all times.

Error #49371A8QMV67. Have a nice day!

OK

3. Prevent Errors

We need to provide good information about what the computer is doing and what it needs.

- Explain what happened and why
- Provide a means of recovery

3. Prevent Errors

Always, always, always provide good error messages.

4. Follow the guidelines for data display

Consistency: Terms, colours, abbreviations ...

Efficiency: Let user take in the information quickly. Line things up, visually organise.

Minimal memory load: Don't make the user remember things from screen to screen

Get their attention: Draw attention subtly

Are you already fed up with this form?

◯ Yes

◯ Yeah
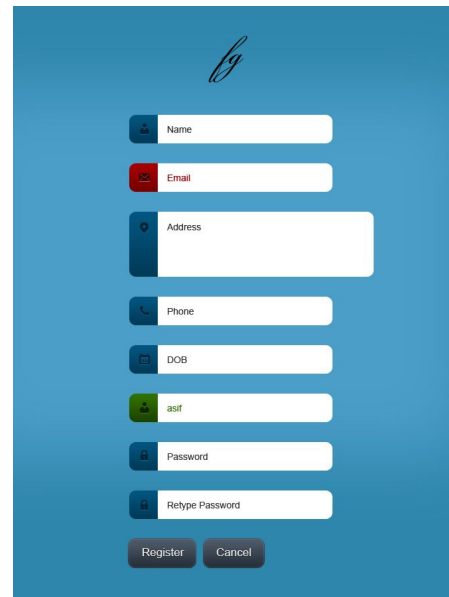
SUBMIT

5. Follow the guidelines for data entry.

Consistency: Similar style for similar data.

Minimal actions: Selectors save time!

Efficiency: Pre-fill forms if they're tedious info that needs to be filled in all the time (like address and postcode)

6. Balance automated/human control.

Tedious/routine tasks: Give it to a robot

Decision making/creative: Give it to a human
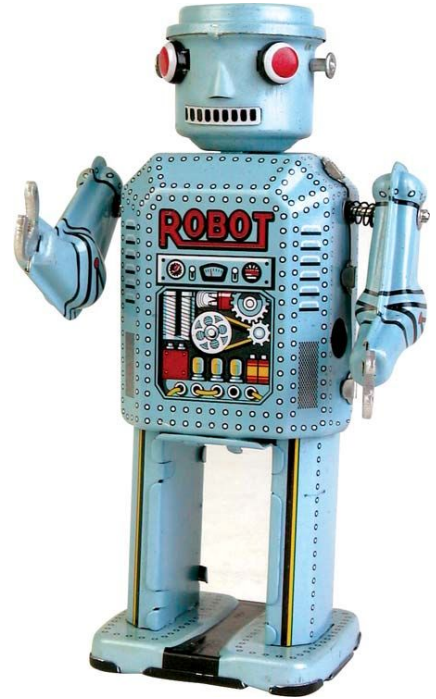
Good GUIs are:

Easy to learn.

Easy to remember.

Predictable.

Have few errors.

If they have errors, you can recover.

Efficient.

Engaging.

Week 4, Part 2: Multimodal interaction

The interaction style is determined by the experience supported by the interface.

Instructing (command based)

Conversing (dialogue based)

Manipulating (static interaction with the environment)

Exploring (dynamic interaction with the environment)

Instructing

Typing, pressing buttons, selecting from menus, talking, gestures …
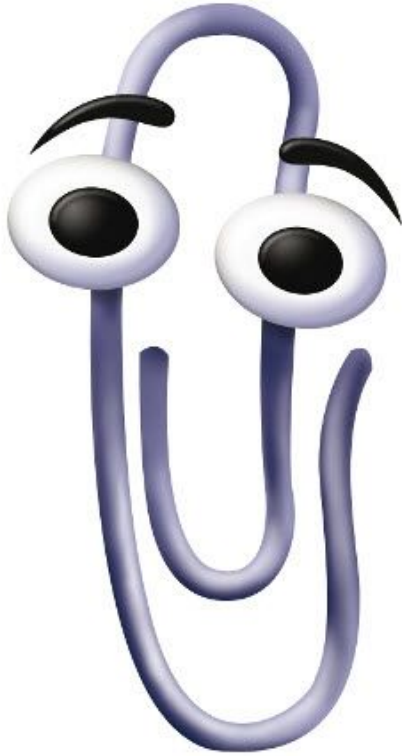
99% of interactions are instructing.

Conversing

Dialogues through question and answer

Speech based interfaces making headway

More natural, but tedious.

Sometimes I just popup for no reason at all. Like now.

Manipulating

Interacting with **objects**, physically or virtually

Current trend is **direct manipulation**

Easy to learn and remember, but slow for experts.

Exploring

Moving through physical or virtual environment

Fantasy/virtual worlds, games

Interactive art and installations

The world that we live in is multimodal.

We experience it through what we see, hear, touch, and taste.

We experience the world through multiple modes of communication, to which our senses are attuned.

Multimodal interaction

This means an interaction uses two or more natural communication modalities - speech, touch, gesture, and so on.
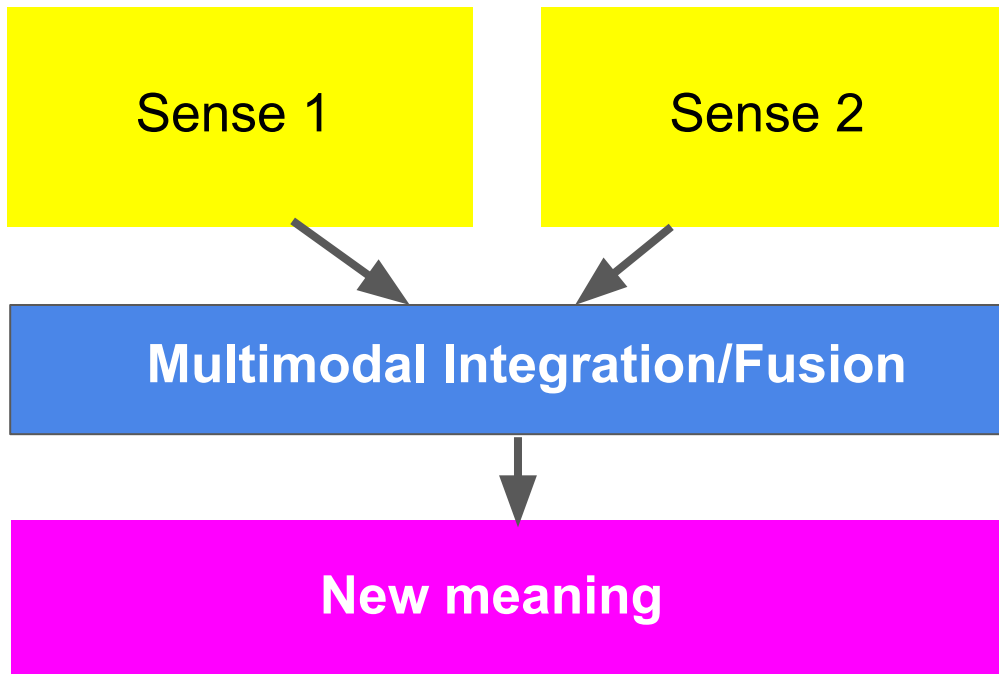
Advantages over GUIs:

- These interactions are flexible
- They fuse our natural multiple inputs
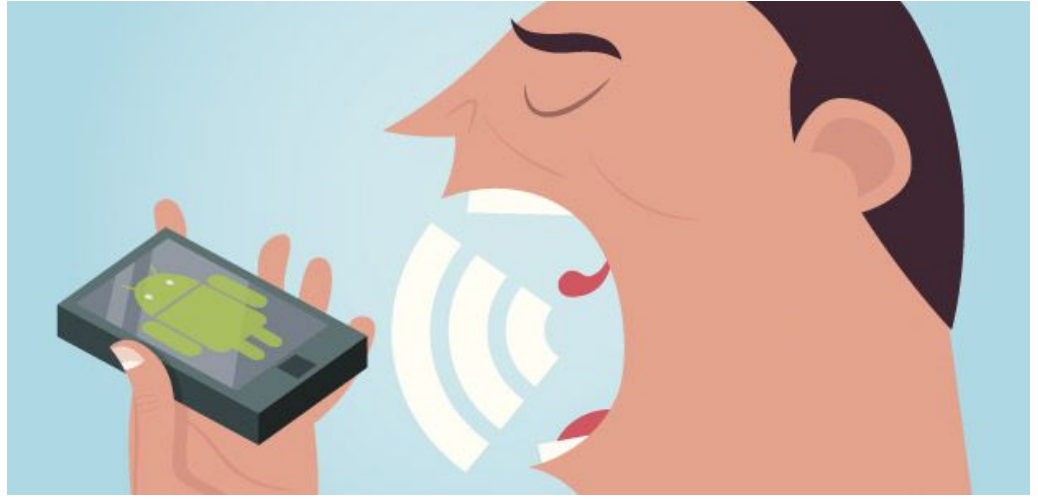- They more closely mirror how we experience the world

Mobile is making a lot of headway here!

Multimodal fusing

This means that the meaning made by two senses are combined into a new meaning.

Mutual disambiguation

Command recognition requires the system to process two or more simultaneous inputs on different input channels.
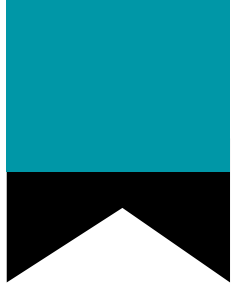
Mutual disambiguation

Command recognition requires the system to process two or more simultaneous inputs on different input channels.

Multimodal: Adapting to the environment

Multimodality allows users to adapt the interaction to their environment.

Is the room too loud for voice commands? Switch to an input screen.

Multimodal: Adapting to the user

If a person has a disability, they have choice. They can tailor an interaction to their ability.

Physical limitations preventing mouse use can be mitigated with voice commands.