Frontend Design

CSC 326

© NC State Software Engineering Faculty

1

Why Study Design Notation?

- Drawing and diagramming is essential task in software development

Understanding existing code

Design review

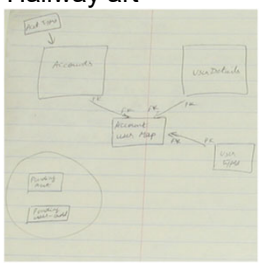
Explain to external members



Ad-hoc meetings (whiteboard)

Onboarding

Hallway art



Refactoring plans

Explain to secondary members

Documentation

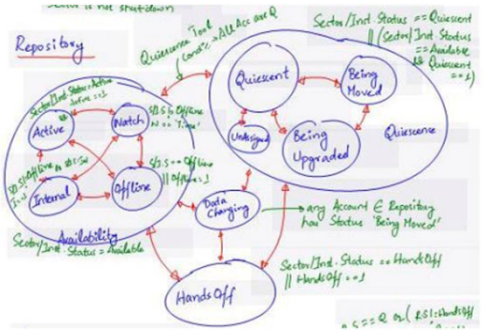


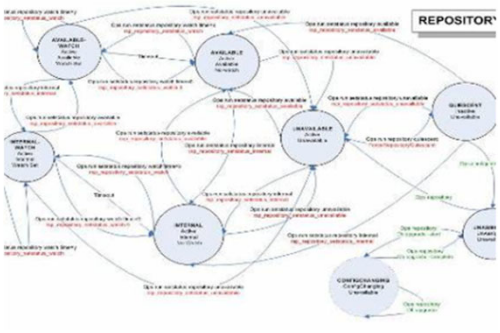
"Let's go to the whiteboard: how and why software developers use drawings" **CHI'07**
http://research.microsoft.com/pubs/74243/p557-cherubini.pdf

2

Lifecycle of a Diagram

1. Transient sketches and reverse-engineering visualizations
2. Reiterated sketches (recurring need to recreate same sketch).
3. Rendered drawings
4. Archival drawings





3

Types of Diagrams

Interface Design

- Wireframes
- Flow Maps/Navigation Maps

Software Design

- Class Diagrams
- Sequence Diagrams

…

Many more specialized types



Specific diagram types [edit]

A
- Activity diagram used in UML 6/9 and SysML

B
- Bachman diagram
- Booch – used in software engineering
- Block diagram
- Block Definition Diagram (BDD) used in SysML
- Business & IT Diagram (B&IT) – used in business and IT modelling

C
- Carroll diagram
- Cartogram
- Category theory diagrams
- Cause-and-effect diagram
- Circuit diagram
- Class diagram – from UML 1/9
- Collaboration diagram – from UML 2.0
- Communication diagram – from UML 2.0
- Commutative diagram
- Component diagram – from UML 3/9
- Composite structure diagram – from UML 2.0
- Concept map
- Constellation diagram
- Context diagram
- Control flow diagram
- Contour diagram
- Cordier diagram
- Cross functional flowchart

D
- Data model diagram
- Data flow diagram
- Data structure diagram
- Dendrogram
- Dependency diagram
- Deployment diagram – from UML 9/9
- Dot and cross diagram
- Double bubble map – used in education
- Drakon-chart

E
- Entity-Relationship diagram (ERD)
- Event-driven process chain
- Euler diagram
- Eye diagram – a diagram of a received telecommunications signal
- Express-G
- Extended Functional Flow Block Diagram (EFFBD)

F
- Family tree
- Feynman diagram
- Flow chart
- Flow process chart
- Fusion diagram

L
- Ladder diagram
- Line of balance
- Link grammar diagram

M
- Martin ERD
- Message Sequence Chart
- Mind map – used for learning, brainstorming, memory, visual thinking and problem solving

N
- N2
- Nassi–Shneiderman diagram or structogram – a representation for structured programming
- Nomogram
- [Network connection]

O
- Object diagram – from UML 2/9

P
- Package diagram from UML 4/9 and SysML
- Parametric diagram from SysML
- PERT
- Petri net – shows the structure of a distributed system as a directed bipartite graph with annotations
- Phylogenetic tree - represents a phylogeny (evolutionary relationships among groups of organisms)
- Piping and instrumentation diagram (P&ID)
- Phase diagram used to present solid/liquid/gas information
- Plant Diagram
- Pressure volume diagram used to analyse engines
- Pourbaix diagram
- Process flow diagram or PFD – used in chemical engineering
- Program structure diagram

R
- Radar chart
- Radial Diagram
- Requirement Diagram Used in SysML
- Rich Picture
- R-diagram

S
- Sankey diagram – represents material, energy or cost flows with quantity proportional arrows in a process network.
- Sentence diagram – represents the grammatical structure of a natural language sentence.
- Sequence diagram from UML 8/9 and SysML
- SDL/GR diagram – Specification and Description Language. SDL is a formal language used in computer science.
- Smith chart

4

UI v.s. UX

UX vs. UI designers

UX designer	UI designer
Interaction designer	Visual designer
Charts the user pathway	Chooses color and typography
Plans information architecture	Plans visual aesthetic
Expert in wireframes, prototypes, and research	Expert in mockups, graphics, and layouts



5

UX Design Methods

18F

https://methods.18f.gov/

Discover

- Bodystorming
- Cognitive walkthrough
- Contextual inquiry
- Design studio
- Dot voting
- Heuristic analysis
- KJ method
- Metrics definition
- Stakeholder and user interviews

Decide

- Affinity diagramming
- Comparative analysis
- Content audit
- Design principles
- Journey mapping
- Mental modeling
- Personas
- Site mapping
- Storyboarding
- Style tiles
- Task flow analysis
- User scenarios

Make

- Design pattern library
- Prototyping
- Wireframing

Validate

- Card sorting
- Multivariate testing
- Usability testing
- Visual preference testing

6

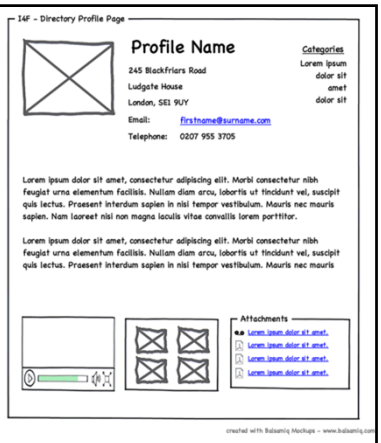
Wireframes

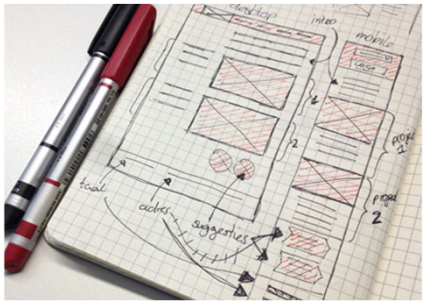
A **wireframe** is a view schematic that captures all layout and content decisions of that view.

- How will you allocate space for particular content?
- Where will content live?
- What actions can you do?

Can vary from low-fidelity to high-fidelity.

Tool: https://balsamiq.com/





7

Wireframes – How To

Look at the requirements

1. Identify items that require physical space on the screen

8

Wireframes – How To

Uploa health care practitioner racker data

- The HCP can select a date on the calendar and enter the following fitness data:
  - Calories Burned, Steps, Distance, Floors, Minutes Sedentary, Minutes Lightly Active, Minutes Fairly Active, Minutes Very Active, Activity Calories.
- The HCP can click on a calendar day and view the patient's fitness information
- The HCP can create the following summary reports:
  - A line graph showing step counts over a user-specified time frame. The time frame must be at least one day in length.
  - A chart or graph of weekly step count averages

9

Wireframes – How To

Look at the requirements

1. Identify items that require physical space on the screen
2. Group together the items based on various views/pages

10

Wireframes – How To

Upload health tracker data

- The HCP can select a date on the calendar and enter the following fitness data:
  - Calories Burned, Steps, Distance, Floors, Minutes Sedentary, Minutes Lightly Active, Minutes Fairly Active, Minutes Very Active, Activity Calories.
- The HCP can click on a calendar day and view the patient's fitness information
- The HCP can create the following summary reports:
  - A line graph showing step counts over a user-specified time frame. The time frame must be at least one day in length.
  - A chart or graph of weekly step count averages

11

Wireframes – How To

Upload health tracker data

- The HCP can select a date on the calendar and enter the following fitness data:
  - Calories Burned, Steps, Distance, Floors, Minutes Sedentary, Minutes Lightly Active, Minutes Fairly Active, Minutes Very Active, Activity Calories.
- The HCP can click on a calendar day and view the patient's fitness information
- The HCP can create the following summary reports:
  - A line graph showing step counts over a user-specified time frame. The time frame must be at least one day in length.
  - A chart or graph of weekly step count averages

12

Wireframes – How To

Related to fitness data!

Upload health tracker data

- The HCP can select a date on the calendar and enter the following fitness data:
  - Calories Burned, Steps, Distance, Floors, Minutes Sedentary, Minutes Lightly Active, Minutes Fairly Active, Minutes Very Active, Activity Calories.
- The HCP can click on a calendar day and view the patient's fitness information
- The HCP can create the following summary reports:
  - A line graph showing step counts over a user-specified time frame. The time frame must be at least one day in length.
  - A chart or graph of weekly step count averages

13

Wireframes – How To

Related to calendar!

Upload health tracker data

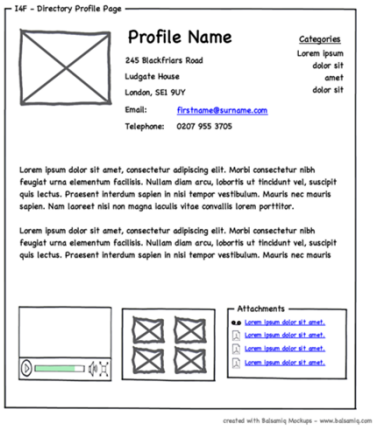
- The HCP can select a date on the calendar and enter the following fitness data:
  - Calories Burned, Steps, Distance, Floors, Minutes Sedentary, Minutes Lightly Active, Minutes Fairly Active, Minutes Very Active, Activity Calories.
- The HCP can click on a calendar day and view the patient's fitness information
- The HCP can create the following summary reports:
  - A line graph showing step counts over a user-specified time frame. The time frame must be at least one day in length.
  - A chart or graph of weekly step count averages

14

Wireframes – How To

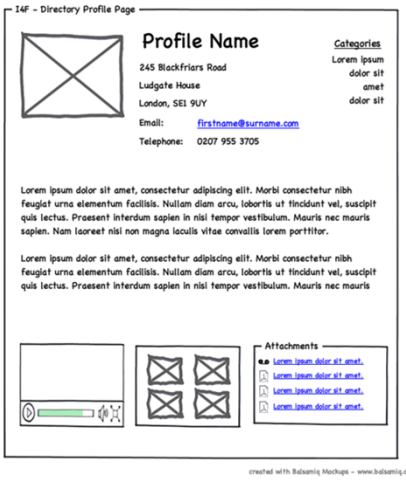
Look at the requirements

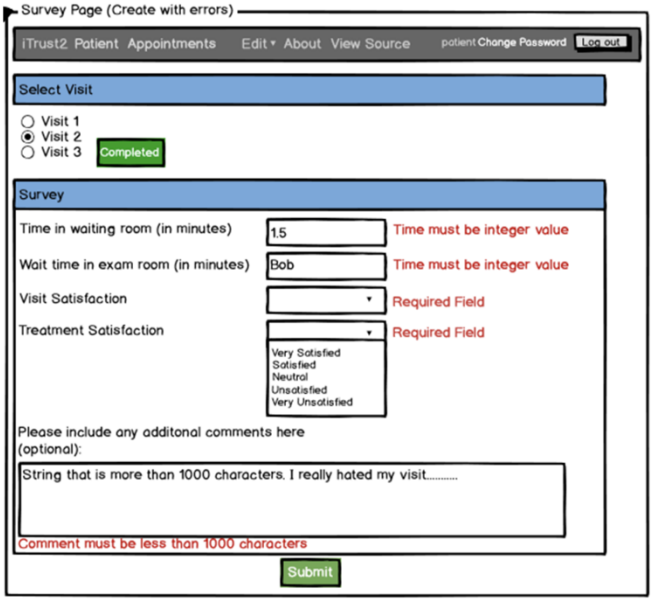
1. Identify items that require physical space on the screen
2. Group together the items based on various views/pages
3. Draw out the pages



15

Design Documents: **Frontend Static** *Wireframes*





16

Traceability

Elements in a wireframe should be traceable to the requirements

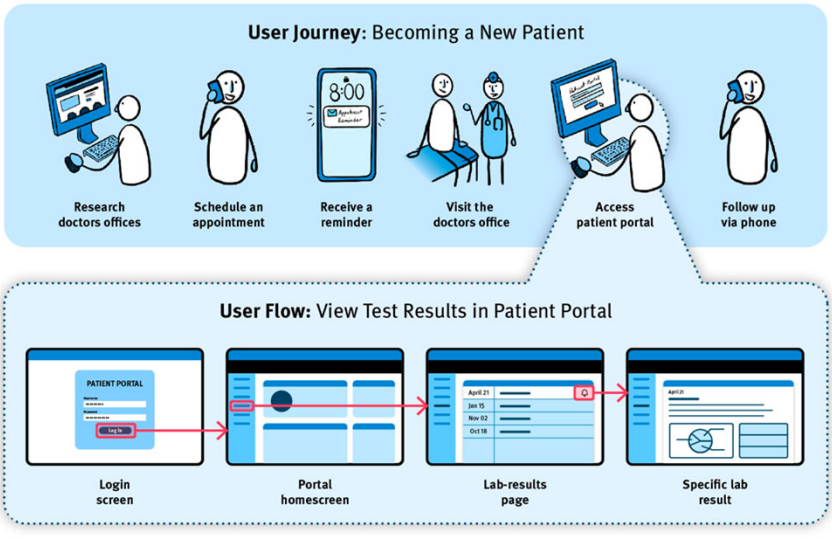
Wireframe elements that have *behavior* (e.g., a clickable button) should be traceable to the next wireframe that response to the click.

17

17

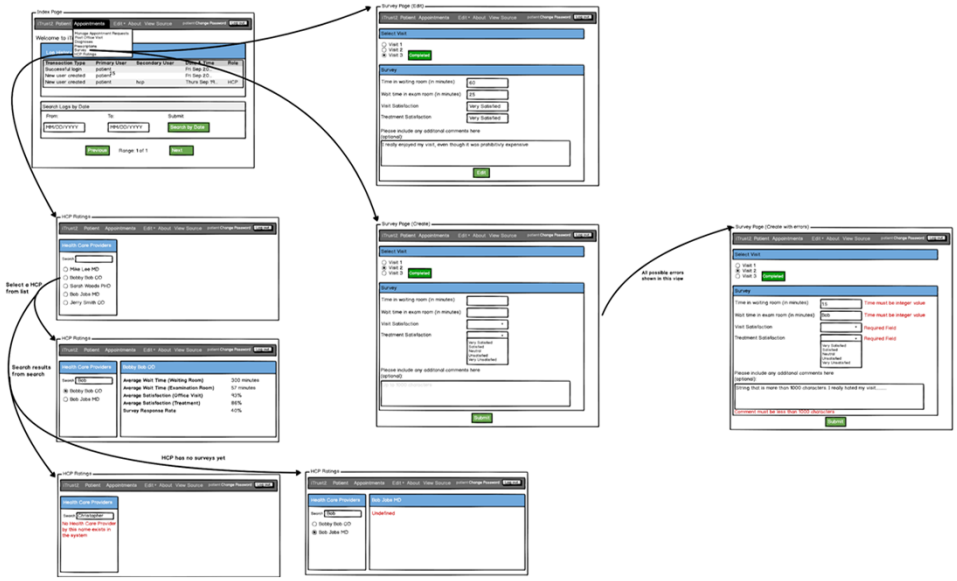
Flow Map / Navigation Map

- A **flow map** describes the *wayfinding* activity of a user and transitions between UI states.



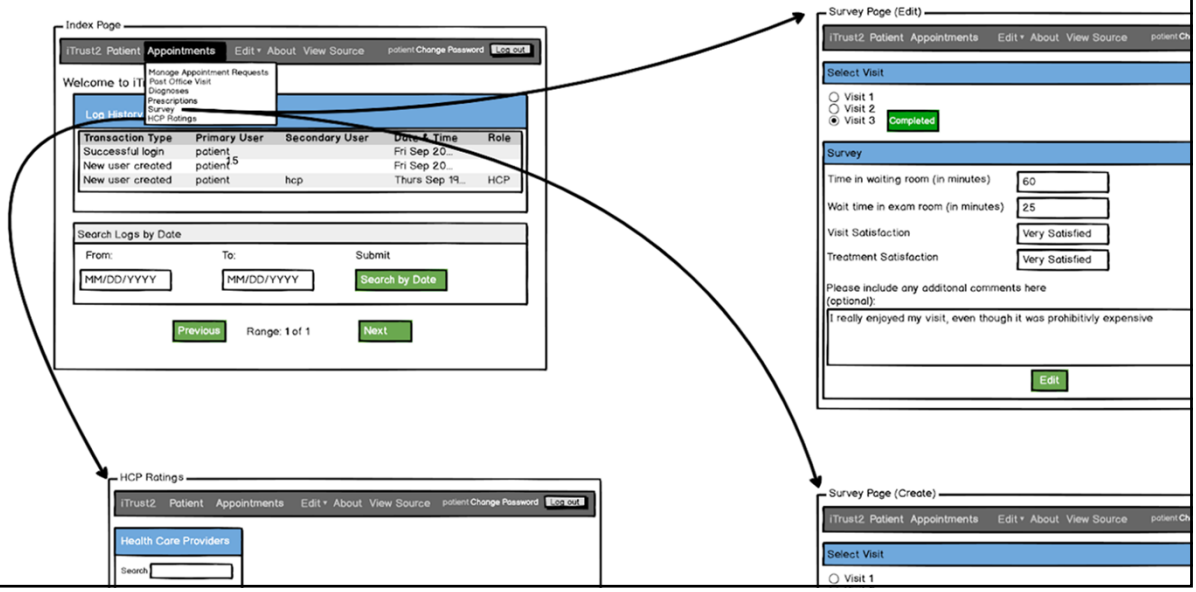
18

Design Documents: **Frontend Dynamic**
Multiple Wireframes



19

Design Documents: **Frontend *Dynamic***
Multiple Wireframes (zoomed in)



20

Exercise

You are asked to design a mobile app for tracking heart rate while performing activities such as running (location data). App allows you to share data with others, and compare to previous heart rates based on location or time.

In teams, create a wireframe for viewing heart rate over time and location and other features of app. (5 minutes)

Describe the UI transitions that you can make using a flow map. (5 minutes)

21

Exercise Part 2

This is paper prototyping!
A usability validation technique.

Now, trade your design with another group.

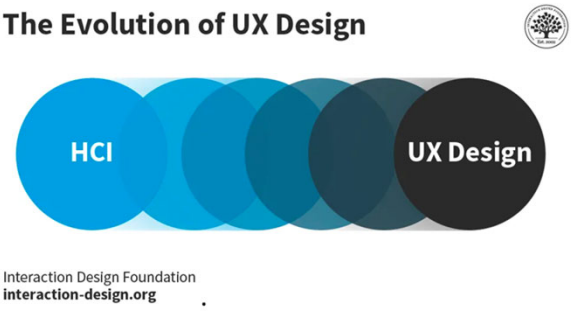
Study the design in front of you. Think about how you would perform the following tasks:

1. View past heart rates based on your current location
2. Share your data with a friend

Using the designs, describe to the designer how you would perform tasks 1 and 2. Is it possible? Is the design missing something?

Exercise 09.01: Paper Prototyping Feedback

22



HCI: Human-computer interaction

Based on content from Dr. Kenya Oduor

23

What is HCI?

Human-computer interaction is a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them

In comparison:

Computer science is the study of the phenomena surrounding computers

24

24

Pragmatic reasons to care

- Everyone likes well-designed tools and environments
- What makes something “well designed”?
  - It solves your problem
  - It reduces your effort
  - It is easy to learn
  - You enjoy your experience using it

25

25

Pragmatic reasons to care

What do people say about bad design?
How can you tell if something is poorly designed?

- It’s confusing or frustrating
- It’s inefficient
- It doesn’t tell you when you’ve made mistakes
- It doesn’t tell you what you’ve done
- It doesn’t tell you what it’s doing
- It’s dangerous

26

26

Why should we care about HCI?

Benefit 1: More Usable Products

Understanding the people you're designing for can help you make systems that are more usable, more learnable, and more efficient.

27

[Ritter et al., 2014]

27

Why should we care about HCI?

Benefit 2: Financial Savings

Designing to support users can save companies money, even of the order of millions of dollars.

28

[Ritter et al., 2014]

28

Why should we care about HCI?

Benefit 3: Safer Systems

Much of the work that has made airplanes the safest transportation per passenger mile has gone into supporting pilots and air traffic controllers to avoid and, more importantly, catch and recover from errors.

29

[Ritter et al., 2014]

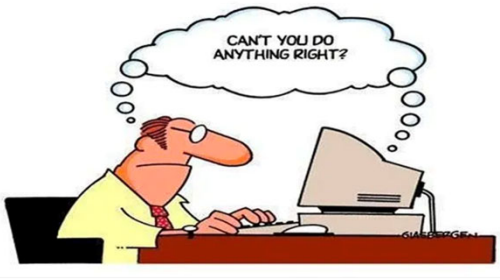
29

Maybe it's just about making everyday life a little better instead of a little worse

30

30

What makes HCI hard?



Pragmatics
- Discovering what people want and need
- Designing for people as they are
- Integrating technology into people's activities

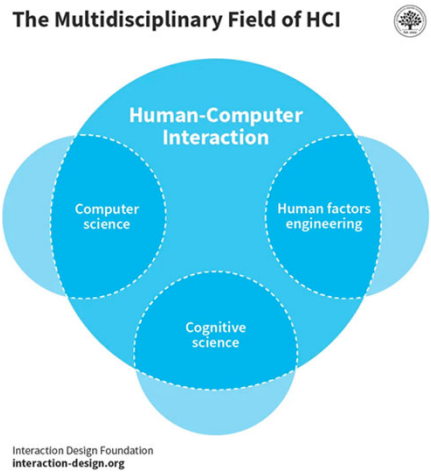
Theory
- Understanding people
- Understanding technology
- Understanding subtleties in their interaction

31

31

Computing perspectives

1. The **physical computing** perspect
2. The **cognitive computing** perspec
3. The **social computing** perspective



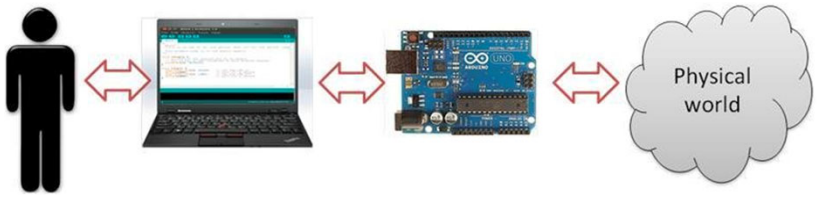
32

32

Computing perspectives

The **physical computing** perspective

- Safety
- User space, work space
- Distractions
- Lighting
- Ambient sound
- Pollution



33

33

Computing perspectives

The **cognitive computing** perspective

- Age
- Disabilities
- Degree of technical knowledge
- Degree of focus
- Cognitive stress

34

34

Computing perspectives

The **social computing** perspective

- Presence/absence of others
  - Physical
  - Virtual
- Social relationships
- Organizational relationships
- Other people as resources

35

35

The **physical computing** perspective

36

36

Physical computing perspective

Anthropometrics

Anthropometrics:
The shape of the body and how it influences what is designed

- consideration of the physical characteristics of intended users such as what size they are, what muscle strength they have, and so on
- analysis of what users can do, including movement

37

37

Physical computing perspective

Anthropometrics

Example usability issues:

- Differences in limb sizes, body weight, height, …
- How sensitive touch is for input and output
- Muscle strain
- Posture

38

38

The **cognitive computing** perspective

39

39

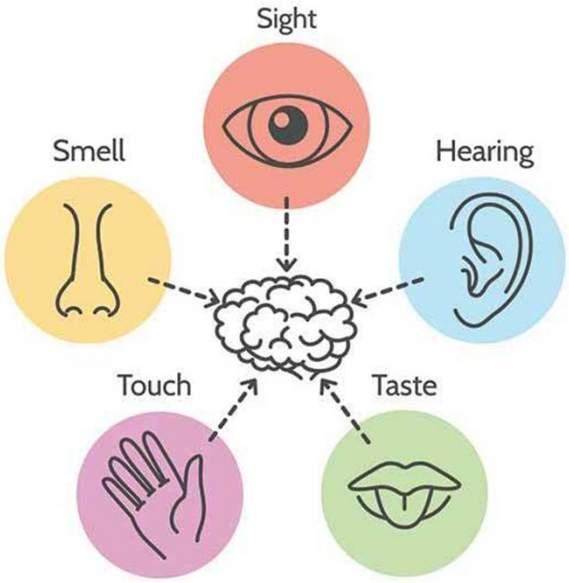
Perception is the
sensory experience of the world

40

40

Cognitive computing perspective

Perception



https://marcus-jackson.com/2018/07/31/the-five-senses-and-the-first-day-of-school/

41

41

Cognitive computing perspective

Cognition

Cognition includes:

Learning, attention, and other aspects of cognition and how these processes influence design

- Defined by how users think and what they know and what knowledge they can acquire

42

42

Cognitive computing perspective

Cognition

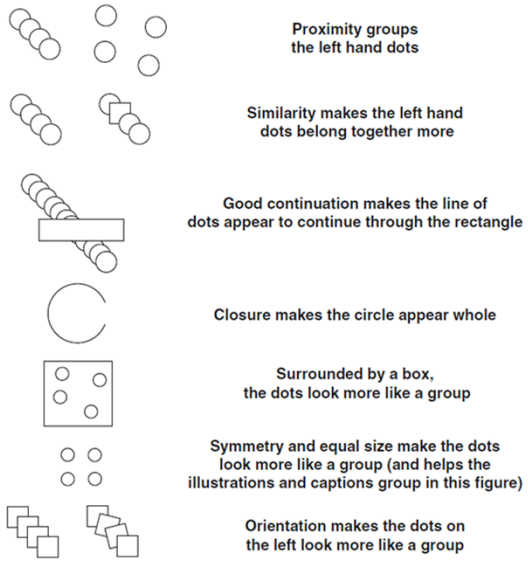
Example usability issues:

- How do users decide where to look for information?
- What information do users need to develop a strategy for performing a particular task?
- What is the user's mental model of the interface and task?
- Is there so much noise and interruption that users cannot process information?

43

43

Gestalt principles of visual grouping



Explains how groups of objects should be interpreted based on their relationship to other objects

44

Cognitive computing perspective

Cognition - Motivation

MOTIVATION DRIVES BEHAVIOR

Useful to understand & apply to gaming software design.

Want to keep your user engaged? You have to know the appropriate motivation for them to carry on playing

Why we behave in a certain way:

1. We act automatically as a result of the central nervous system. We have no control over these sorts of actions
   - Someone blows a puff of air in your eye and you automatically blink

2. We make a deliberate decision to perform a particular act
   - One of the influencing factors that determine not only the action, but how well we perform is *motivation*

45

45

Cognitive computing perspective

Cognition - Behavior

Behavior:

Perceptual and motivational characteristics, looking at what people can perceive and why they do what they do

46

46

Cognitive computing perspective

Cognition - Behavior

Example usability issues:

- Car/mobile interfaces: legibility of characters, clarity of instructions during navigation
- Looking at simple errors when using an application, to see how they can be mitigated or prevented

47

47

The **social computing** perspective

48

48

Social computing perspective

Social

Social factors:

How groups of users behave, and how to support them through design

- Users defined by where they are—their context broadly defined including their relationships to other people.

49

49

Social computing perspective

Social

Example usability issues:

- A crew distracted by interruptions that failed to complete a safety checklist did not confirm that the airplane's flaps were extended, causing the plane to crash on take-off
- A communications breakdown between captain, co-pilot, and air traffic control on the amount of fuel in the plane caused a crash when the fuel ran out

50

50

How do we use this information?

51

51

Concept Design

Concept design involves

- Structuring the information space
- Creating of alternative solutions
- Determining which design concept to pursue

52

52

Concept Design – Create task flow

Create a new contact				
User	1. Gather new contacts details	2. Locate Contact area in the application	3. Enter Contact details	4. Save New Contact details
System		Create button on Home page	Textfields to enter Contact details	Save button and way to show user new contact has been created and saved
Considerations for design		Primary action in the app so button should be front and center of Home page	Required fields: Name Email Phone Optional fields: Address Birthday	Creation success = Info is read only and presented in the user interface (UI)

53

53

Concept Design – Create task flow

Scenario: Make a new post in Vistas app					
User	1. Take awe-inspiring picture	2. Locate the "new post" area in the application	3. Upload photo	4. Write text with emojis	5. Upload post
System		Create new post button on Home screen	Icon that opens the photo app	Phone keyboard appears when user clicks on text field.	Upload button is easy to find. After upload, the user sees a confirmation that the post was created.
Considerations for design		A common action in the app, so button should be front and center on the Home page	The app has permissions to access the photo library	Keyboard includes emoji menu	Creation success is presented in the UI using both color and shape to indicate success

54

54

Concept Design

- Identify the user steps, system task, and design considerations for the following user story. Discuss the physical, cognitive, and social computing perspectives

A user wants to share their latest run and heart rate information with a friend.

55

Want to learn more?

CSC 454/554!

56

56