# COMPSCI 345 EXAM NOTES

# Section 1: Lecture 2: Usability Evaluations

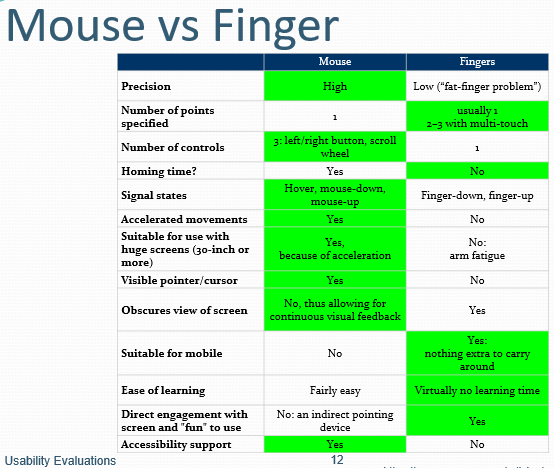
**Learning objectives**

* Be able to define usability
* Be able to describe aspects and measure of usability
* Appreciate that usability depends on context
* Be able to compare and contrast the different usability evaluation techniques
* Be able to select an appropriate usability evaluation technique for a problem

**Usability**

* Usability is the measure of the quality of a user’s experience when interacting with a product or system.
* Usability is a quality attribute that assess how easy user interfaces are to use.

**Usability Factors**

* Fit for use
  + Can the system support the tasks that the user wants to perform?
  + Does the system function as expected?
  + What are the system goals?
  + Average (measures)
* Ease of learning
  + How fast can a user who has never seen the user interface before learning it sufficiently well to accomplish basic tasks?
  + Do you have to read a manual for help?
  + How much time are you prepared to invest in it and what are your usability expectations.
* Efficiency of use
  + Once an experienced user has learned to use the system, how can he or she accomplish tasks?
* Memorability
  + If a user has used the system before, can he or she remember enough to use it effectively the next time or does the user have to start over again learning everything?
  + Shouldn’t have problems for users to remember important tools they will frequently be using.
  + A good interface is one where you remember or can, with prompts, recall what to do.
* Error frequency and severity
  + How often do users make errors while using the system, how serious are these errors, and how do users recover from these errors?
  + What is the cost of these errors.
* Subjective satisfaction
  + How much does the user like using the system?
  + If users like the interface
    - They will make less errors
    - They will persist longer when they are having problems
  + Aesthetics – how nice does it look is very important
    - This can include font size, clean layout, good grouping

**Performance measurements**

* Are directly observable by watching a user complete a task within a specific time. This includes monitoring the number of errors and time needed to accomplish the task. These types are “quantifiable measures’ which means that they can be communicated with numbers.
  + Fitt’s law the classic performance measure for time to complete the task of pointing at an object
    - Time to target depends on target width (W) and distance to move pointer (D)
    - T = a + b \log_2 \Bigg(1+\frac{D}{W}\Bigg)
    - It is very valuable measure for designing
      * Control size and location
  + Hick-Hyman Law time taken to make a decision
    - The time it takes for a person to make a decision as a result of the n possible choices
    - T = b \cdot \log_{2}(n + 1)
    - Particularly important for menus
    - Other factors
      * Recognition time: for icon or word
      * Consistency is good: spatial memory is very powerful

# Lecture 3: Heuristic evaluations and early prototype evaluations

**Learning objectives**

* To be aware of a range of heuristic evaluation options appropriate to the analysis and design phase
  + In particular, well-known heuristics of usable systems
* To understand the evaluations challenges of early prototypes with limited functionality

**Evaluation – Heuristic Evaluation**

* Heuristic evaluations are performed by usability experts using a predetermined set of criteria designed to measure the usability of a proposed design
* The evaluator follows a scenario through the design and tests each step against the heuristic criteria
* Carrying out heuristic evaluation is an excellent way to get an understanding of a system that you are going to usability test.
* Or a heuristic evaluation can be stand-alone with the evaluator making observations and recommendations based on their experience.

**Nielsen’s Heuristics**

* Visibility of system status
  + The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.
* Match between system and the real world
  + The system should speak the user’s language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order
* User control and freedom
  + Users often choose system functions by mistake and will need a clearly marked “emergency exit” to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.
* Consistency and standards
  + Users should not have to wonder whether different words, situations, or actions mean the same thing.
* Error prevention
  + Even better than good error messages is careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.
* Recognition rather than recall
  + Minimize the user’s memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.
* Flexibility and efficiency of use
  + System should cater to both inexperienced and experienced users. Allow users to tailor frequent actions.
* Aesthetic and minimalist design
  + Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information
* Help users recognize, diagnose, and recover from errors
  + Error messages should be expressed in plain language, precisely indicate the problem, and constructively suggest a solution.
* Help and documentation
  + Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user’s task, list concrete steps to be carried out, and not too large.

**Evaluating prototypes**

* Evaluating early prototypes is a bit different to evaluating fully functional systems because of the lack or limited functionality
* Or could be a semi functional prototype in a prototyping environment or UI design tool.

**Functional prototypes**

* Functional prototypes are interactive prototypes that represent various degrees of functionality
* Functioning prototypes can be created using
  + Prototyping tools
  + Or RAD environments, such as:
    - Microsoft
    - Adobe

**Prototyping tools**

* Advantage
  + Closer to the real interface and can explore the functionality a bit more
* Disadvantage
  + Lock down the design therefore inhibit creativity

# Lecture 4: Usability Testing

**Learning objectives**

* Be able to articulate why and when to do usability testing
* Be able to develop usability testing plans
* Be able to write usability test reports

**Usability studies**

* Empirical user test
  + Get real users to try to perform specified tasks
  + Observe and record
  + Ask their opinion
* Analyse results
  + What is causing problems?
  + How can you fix them?
* Specific tasks
  + Observed
  + Recorded
  + Measured
  + Think-aloud

**Usability Testing**

* Testing it with representative users
  + Users will try to complete typical tasks while observers watch, listen and take notes.
* Goal is to identify any usability problems
  + Collect quantitative data on participants performance
  + Determine participant’s satisfaction with the product
* You should test as early as possible; usability testing lets the design and development teams identify any problems before they get coded.

**How to test**

* Know what your goal is (actually this is true for all usability evaluation methods – heuristic, performance measure-based or participant based)
  + Focus on whatever you believe are the key aspects
* Set the tasks accordingly
* Recruit participants
* Test
* Observe
* Record

**Key points**

* About completing routine tasks successfully
  + How long It takes to do that
  + Where people run into trouble
  + What sort of errors they make
* How satisfied participants are with your interface
* Helps to identify changes required to improve user performance
  + Finding a problem doesn’t hand you the answer, but at least gives you a focus for re-design / iterative refinement.
* Measures the performance to see if it meets your usability objectives

**Test Planning**

* A good plan is absolutely essential for a good test and defendable results
* The higher the stakes, the better the plan needs to be
  + In early iteration for design it might be quite informal
  + As software moves from design to prototype to pre-market product the formality picks up
  + Can also do formal testing as part of product selection, too.
    - It’s much more common to be selecting a product to get a job done than to be perfecting a product for market.

# Lecture 5: Usability Testing Planning and Reporting

**Learning objectives**

* Be able to develop usability testing plans
* Be able to write usability test reports
* Understand the nature of human research ethics requirements when conducting studies on humans.

**Planning**

* Think that you are planning the test for 5 other people to do, each in a different part of the world so it must be useful to all the participants.
* Product under test
  + Exactly what and how it is going to be tested include
* Test objectives
  + What’s the goal? What are you planning to measure?
* Participants required
* Equipment needed
* Tasks to undertake
* Data to be collected
* Test procedure
* Data analysis plan

**Planning**

* Generally, it is better if people can be themselves
* Try to match age, gender and things like first language to the target audience
* How many?
  + 10-12 will generally give you good results for a single product
  + 30+ if comparing products and you want statistically valid results

**Task Design**

* Record completion Paths
  + Step through the task yourself
  + Record different routes to successfully complete the task
  + Time yourself
* Note things you think are difficult or confusing
  + You are, in effect doing a heuristic evaluation
  + Remember these are inaccurate
* Do not show these to participants

**Procedures**

* Don’t under-estimate the practical problems
  + If you get something wrong, you can lose a lot of time having to reschedule
  + Have you figured out
    - How to pay participants
    - Any catering for longer sessions
    - Do you have enough power points?
    - The time and ability to rest in between participants

**Analysis plan**

* How are you going to turn the raw observations into assessment against your usability requirements, and into recommendations?
* You must define exactly which elements of the task you are timing, and the protocol for marking the task time.
  + Is the method practical and accurate? Will it support the overall purpose of your usability test?
* What is the plan for how to report the findings?
* Errors – will need to count them and form categories
  + Wrong navigation
  + Problem finding particular features etc.
* Questionnaire analysis
  + If around 10 people or less, show raw data, mean and standard deviation
  + If more than 10 people, box plots or frequency distribution graphs might be appropriate.
* Pilot Test
  + Try the whole thing out on one or two people
  + After the first person fix the obvious problems
    - If very few corrections are needed in test plan, then you can go straight to testing
    - But it is much better to do a second pilot than discover major problems half way through
* Summarize information into tables
* Use numbers where you can
* Classify comments into groups

**Ethics**

* If you are doing a study with living (human or animal) participants in a university you will probably need ethics approval
  + Can be quite a lot of paperwork, and takes a while to get authorization
  + You will need such approval for a study to be part of your dissertation or thesis
  + Many journals require such approval to publish
* Quite a few companies have similar requirements
* This is why for your assignment you are not testing on others.

**Research ethics basics**

* Informed consent
  + Participant knows what they are “in for”
    - Task time why you’re doing it
    - Confidentiality of their data
    - Compensation
  + Participant is clear that they are not compelled to participate
    - They need to know that they can refuse, or withdraw without jeopardizing the key service
  + Anonymous questionnaires, esp. in public, are probably the easiest from an ethics perspective.

**Ethics application**

* Explains protocol and goals: essentially like a test plan
  + And so it’s helpful to complete one because it acts as a check on your plan
* Particular focus on issues such as who has access to the data and the risk to participants
* Research organizations have standing committees to review applications
  + Have representatives from a range of perspectives: clinical, legal, statistical

# Lecture 6: User requirements

**Learning objectives**

* To describe when user requirements are gathered
* To develop a set of skills for systematic analysis of a problem domain to discover HCI issues and requirements
* To understand perspectives on data collection including types of stakeholders
* To be able to collect data by interviews, focus groups and questionnaires
* To be able to create user profiles

**When are user requirements gathered?**

* At the beginning
  + Of the software development lifecycle – regardless of life cycle
  + When writing new software
  + Or when evaluating software products

**Requirements collection processes**

* During the collection phase you will formally identify:
  + People involved
  + Things they use
  + Processes involved
  + Information required
  + Constraints imposed
  + Inputs required
  + Outputs created
* You will then model the information by:
  + Creating descriptions of the people who do the work
  + Documenting the main use-cases
  + Creating different stories about how the various aspects of the work are done
  + Creating formal diagrams of the interaction

**Exploring the Domain**

* Identify all stakeholders
* There are four types of stakeholders
  + Primary – A person who uses the design directly (must have the most impact on the eventual design)
  + Secondary – a person who either supplies input or receives output from the design
  + Facilitator – A person who maintains or develops the design
  + Indirect – A person who is affected by the use of the design but has no contact with it, such as the user’s superior or coworkers and the client who is paying for the project (the client may or may not also be the primary stakeholder)

**Collection**

* Methods of collection
  + Observation:
    - Watching people perform their activities in the context of the work environment
    - Your presence will affect the people you observe
    - Hawthorne effect – the alteration of behavior by the subjects of a study due to their awareness of being observed.
  + Elicitation:
    - Involves direct and indirect methods of investigation, such as direct and indirect approach:
      * Direct
        + Interview – on-side interviews (may help people remembers aspects of the job) or away from job site interviews (not interrupted by normal work-related events)
        + focus groups

Require moderator/facilitator to keep discussion on track

Maintain spontaneity

Provide participants with a context for the project

Should have peer relationship so aren’t feeling too shy to speak in front of everyone

* + - * Indirect
        + Questionnaires – contain open and closed questions
        + Corporate documentation – policies and procedures
        + Logs and notes – log activities, collect notes people make (sticky notes)

**Advantages of focus groups**

The advantages of focus groups:

* They are relatively inexpensive and easy to set up.
* They can be used early in the design process to help to identify and prioritize features.
* Gain inside into people’s attitudes and motivations
* Help sell a new solution

Disadvantages of focus group:

* They only represent the views of one particular group
* A strong voice can capture the group
* They are not statistically significant
* They do not provide information about usability

**Advantages of questionnaires**

* They do not involve face-to-face contact and can be administered remotely.
* They can be used to supply information for primary stakeholder profiles
* They can be used to ascertain whether proposed solutions will meet with acceptance as well as to elicit new ideas.
* They can reach a large audience with relatively little expense.

**Disadvantages of questionnaires**

* Vague questions will return ambiguous responses that will serve no useful purpose for the design
* People do not like to fill out long questionnaires.
* Closed-ended questions can restrict responses.

**Documentation**

* Requirements
  + Users
    - Who are the users of the system?
    - What are they like?
    - What tasks will they perform?
  + Requirements
    - Functional – what features must be present?
    - Information – what information is needed to carry out the functions? And what outputs are required by the stakeholders?
  + Input/output mediums – desktop, mobile, special, environments?
  + Constraints – physical, financial, time, data storage, networking

# Lecture 7: Modelling and Documenting Requirements

**Learning objectives**

* To learn structured methodologies for describing:
  + Stakeholders of a system
  + Tasks
  + Pathways between states
  + Major uses

**Primary stakeholders**

* Used to define the target user
* Cognitive
  + Education level
  + Computer literacy
  + Typing ability etc.
* Physical ability
  + Visual acuity, colour vision
  + Motor ability
* Individual profile
  + Age, gender, occupation, interests, country, language
  + These profile elements move us toward personas

**Task Analysis**

* Task analysis is a way of documenting how people perform tasks
  + It includes all aspects of the work flow
* It is used to:
  + Explore the requirements of the proposed system and structure the results of the data collection phase
* Two types:
  + Hierarchical task analysis(HTA)
    - HTA provides a top-down, structured approach to documenting processes
      * Goal - being analyzed
      * Plans – the order and conditions for proceeding with the sub-tasks
      * Information – all the information needed to undertake the task
      * Objects – all the physical objects involved
      * Methods – the various ways of doing the sub-tasks
  + State transition network (STN)
    - Provides a description what actions/events are available at what point and what state the system will be in after each action.

**State transition network (STN)**

* Circles – states
* Arcs – actions/events
* Arc labels a bit cramped because:
  + Notation is “state heavy”
  + The events require most detail
* Labels in circles a bit uninformative:
  + Sometimes states are hard to name
  + But easier to visualize

# Lecture 8: Personas and Scenarios

**Learning objectives**

* To appreciate the role of conceptual design prior to physical design
* To equipped with a set of methods for conceptually design, particularly
  + Persons and scenarios
  + Brainstorming, card sort and mind mapping
  + Process diagrams

**Conceptual Design**

* Conceptual design involves
  + Structuring the information space
  + Creating alternative solutions
* Determining which design concept to pursue
* The tools involved in conceptual design:
  + Personas
  + Scenarios
  + Brainstorming
  + Card sort
  + Mind maps
  + Process diagrams

**Personas**

* Personas are archetypes of actual users, defined by the user’s goals and attributes.
* Should be a reflection of information derived from the data collection activities
* Direct one-to-one relation with an observed user behaviour or statement.
* Identify them as if they were a real person
* Can be a synthesis of characteristics observed over multiple users

**Conceptual Design – Personas**

Advantages of personas:

* They are quick and easy to create
* They provide a consistent model for all team members
* They are easy to use with other design methods
* They make the user “real” in the mind of the designer

Disadvantages of personas

* They can be difficult to create if the target audience is international
* Having too many personas will make the work difficult
* There is a risk of incorporating unsupported designer assumptions

**Scenarios**

* Scenarios
  + A description in plain English of a typical task
  + It describes
    - The basic goal
    - The conditions that exist at the beginning of the task
    - The activities in which the persona will engage
    - The outcomes of those activities
  + Paired with a persona
    - We pair with a PACT structure
      * People
      * Activities
      * Context
      * Technology
* Scenario should be situated with one or more personas
* Main difference between a use-case and a scenario is that the scenario uses a person’s name while the use-case is abstracted to the role

**Brainstorming / Mind maps**

* Team activity
* Can work in parallel by breaking into subgroups
* Brainstorming sessions generate a lot of material that must then be **filtered and organized.**
* Mind maps are similar to brainstorming – semi structured
  + Allow easy way to explore problem space
  + Provide a way to create clusters of related elements
  + Graphical view
  + Require knowledge of problem space
  + Can lead beyond the problem space
  + No formal semantics for defining symbol meaning.

**Conceptual Design – Card sort**

* Card sorting can be used to discover user-centered groupings
* Organize the information collected in the discovery phase
* Used to define groupings for menu, controls and web page
* Generate labels for menus, buttons and navigation
* Advantages
  + Easy to perform
  + Can be done before any preliminary designs have been made
  + Let you know how people organize information
* Disadvantages
  + They suggest solutions that imply structures
  + Become difficult to navigate/interpret with more categories

# Lecture 9: Design principles

**Learning objectives**

* To be able to apply design principles in the context of user interface design tasks
* To be able to characterize key design principles for efficacy (ability to produce a desired result)
* To be able to conceptualize design principles in terms of an interaction framework

**Principles of Interaction Design**

* Clear, minimal, to-the-point and insanely successful
* Design principles can be used to guide design decisions
  + Do not prescribe specific outcomes; they function within the context of a particular design project
  + Guide interaction designers to help them make decisions that are based on established criteria

**Framework for design principles**

* Compressibility barrier – if the presentation is comprehensible, then the user can get past the comprehensibility barrier. The user needs to understand how they access the functionality.
* Learnability – if the interface is comprehensible it will be learnable; there is a direct relationship. Learnability and comprehensibility are recursive: we start with comprehensibility which affects learnability, which will in turn increase comprehensibility.
* Usability – if the user can complete tasks then the interface is usable
* Usefulness – if the user can complete the tasks efficiently and enjoyably then the interface is useful.

**Efficacy**

Efficacy is effectiveness + efficiency

* Effectiveness
  + Goal stipulates that a design must fulfill the user’s needs by providing the required functionality
  + Can the user complete the task?
    - Utility – The principle of utility relates to what the user can do with the system
    - Safety – If a design has a high degree of safety, it will prove more useful than a design that involves a high degree of risk.
    - Flexibility – A tool that is flexible can be used in multiple environments and may address diverse needs
    - Stability – A stable system is a robust system
      * A system that functions consistently well will be more useful
* Efficiency
  + Goal stipulates that a design should enable a user to accomplish tasks in the easiest and quickest way possible without having to do overly complex or extraneous procedures.

**Describes usability**

* + - Simplicity – 80/20 rule, 80% of an applications usage involves 20% functionality
      * Show the user only the basics unless they ask for more.
      * Constraints – limiting the actions that can be performed
      * Conventions – exploit learned behaviour to influence user actions
      * Symbols – can influence the way in which we interact with an interface by defining meaning and constraining our possible interpretations of interface elements.
    - Memorability – interfaces that have high memorability will be easier to learn and use
      * Many different parameters affect memorability
        + Location
        + Logical grouping
        + Conventions
        + Redundancy
      * And simplicity helps memorability, too
    - Predictability – predictability involves a person’s expectations and his ability to determine the results of his actions ahead of time.
      * Consistency reinforces our associations and therefore increase our ability to remember and predict outcomes and processes.
      * Generalizability: Use knowledge gathered from previous experience and apply it to similar situations
      * Conventions: use intuition
      * Familiarity: Familiar options/names help users locate object and functions more easily.
      * Location: not all areas on the screen are created equal. Top-left is most prime real estate; bottom and right for closure
      * Modes: modes create instability in mental models because they change the way objects function.
    - Visibility – the principle of visibility involves making the user aware of the system’s components and processes, including all possible functionality and feedback from user actions.
      * Overload(bad): visual overload can be caused by not following simplicity principles
      * Feedback(good): Direct manipulation interfaces provide immediate visual feedback about user actions. It is the task of the interaction designer to decide what form that feedback takes.
      * Recognition/Recall: The principle of visibility is based on the fact that we are better at recognition than we are at recall
      * Orientation: People need to be able to orient themselves, especially in complex information spaces

**Affordance and Metaphors**

* Affordance is the concept that particular items suggest how they should be used
  + Affordance confusion is when certain aspects of an object do not work in a way in which we assume they should
* Metaphor is when we take a physical world concept and use it on computers
  + Most icons are metaphors

# Lecture 10: Physical Design

**Learning objectives**

* To appreciate the value of prototyping to the design process
* To be aware of the different types and options in prototyping and be able to choose the most appropriate ones for specific design problems
* To be able to develop useful prototypes for eliciting user feedback

**Physical Design – Low-fidelity prototypes**

* The physical design involves:
  + What is will look like
  + What components it will require
  + How the user interface will be laid out
* Want constant feedback in the early stages – receiving feedback will help cut costs in the future and be less expensive.

**Physical Design – Low-fidelity prototypes**

* Nielsen distinguishes between two types of prototypes
  + Horizontal
    - Full breadth of functions in the design, but probably at the cost of detail
  + Vertical
    - Drill down on one area of the design
    - This is key if one area is particularly novel and is critical to the success of the design.
* The three main criteria for low-fidelity prototypes:
  + Easy and inexpensive to make
  + Flexible enough to be constantly changed and rearranged
  + Complete enough to yield useful feedback about specific design questions.
* More comfortable criticizing paper prototypes

**Types of prototypes**

* Post-it’s in paper prototypes
  + Can layer them to show pop-ups or other dialogue progression
  + Can stick them on top to revise a screen
  + Cut into strips for pulldown menus
* Make a device
  + Use the cornflakes box, a hunk of polystyrene
  + Paint/draw on the controls
  + Stick on junk
  + Use buttons to represent dials
* Paper prototypes
  + They can be used as early and often
  + They are inexpensive and easy to create
  + They make design ideas visual
  + No special knowledge
  + They are not interactive
  + They cannot be used to calculate response timings
  + They do not deal with interface issues such as colour or font size.
* Electronic prototypes
  + If you still want it quick and easy but want more interactivity than paper
    - Not convinced with paper
    - E-mail prototype to the user and do the session remotely
  + PowerPoint is good
    - Slideshow is a reasonable metaphor for sequence of screens
    - Can insert actions buttons or hyperlinks to traverse between slides in non-linear fashion to simulate an application
* Storyboards
  + Like a comic strip of the persona in a scenario
  + Can be earliest stage of physical design
  + The start of what you might call “visioning” the solution before you start prototyping.
* Whiteboard prototypes
  + Popular for early physical design development among team members
    - More for brainstorming than role-playing
  + But overall can be used much like a paper prototype just another medium.

# Lecture 11: User interface Aesthetics Patterns and Standards

**Learning objectives**

* What are the principles of aesthetics, and how do they apply to UI designs?
* What are user interface patterns? How and why can they help improve my designs?
* How do standards and toolkits relate to usability?

**Aesthetics**

* Visual aesthetics of an interface affects user’s perception of the system.
* An aesthetically pleasing site is perceived as:
  + More usable
  + More trustworthy
  + More error tolerant
* Principles of design are concepts used to organize or arrange the components in a design
* We are going to review 3 principles:
  + Balance
  + Emphasis
  + Unity

**Balance**

* Distribution of the optical weight in an interface (perception that some objects appear heavier than others)
* The balance in screen design is achieved by providing an equal “weight” of screen elements, left and right, top and bottom.
* Symmetry: a mirror image
  + Can occur in any orientation as long as the elements are the same on either side
  + Also, called formal balance because a form (formula) is used
* Asymmetry – without symmetry
  + Also, known as informal balance
  + The term, however, is usually used to describe a kind of balance that does not rely on symmetry
  + There are no rules or limits with asymmetrical balance. It can be achieved by careful placement of objects and the use of other organizational devices.

**Emphasis**

* You can control the attention of someone viewing the visual by creating a hierarchy of dominance among elements, i.e. greater emphasis on some elements.
* The principle of “emphasis” is sometimes called “dominance” because of this relationship
* There are three major methods for controlling emphasis in a visual image:
  + Contrast
  + Placement – central vision is dominant
  + Isolation
    - Dominance of things emphasized by standing alone/apart

**Unity**

* The relationship between the individual parts and the whole of a layout
  + Aspects that are to tie the composition together, to give it a sense of wholeness, or to break it apart and give it a sense of variety
* Stems from Gestalt theories of visual perception
  + Specifically, those dealing with how the human brain organizes visual information into categories, or groups
* Careful placements of components
  + Connect by one grouping tendency
  + Disconnect by others
* Understanding gestalt concepts can help to create unity and variety

**Design patterns**

* A UI design pattern is typically presented as a review of a solution to a design challenge, including:
  + Problem that the pattern addresses
  + Solution and when to use it; how and why
  + Possibly also info on implementation, maybe references, and usually comments

**What is standard?**

* Many organizations require software to be standard compliant
* Look to standards for guidance (usually in terms of functions rather than visual appearance

**Toolkits**

* With a wide community using the same modules
  + Achieve consistency
  + And a community to consult if things aren’t working
* Wide range of modules means you can tack together functionality pioneered by others very quickly
  + Good fit to agile development
* Consider toolkits – especially in later stages of prototyping – to achieve quick and stable results that will be familiar to users.

# Lecture 12: Human Visual Perception

**Learning objectives**

* To think about the human as an information processing machine
* To be able to identify components of visual perception, including acuity, colour perception, depth perception and object recognition
* To consider how individual deficits in vision, or poor visual presentation, can impair usability
* To lay the groundwork for understanding further aspects of the visual interface
  + Including aesthetics, principles of perception and grouping, and use of text and colour
  + As well as to help in identifying novel HCI opportunities

**Human processor**

* Input
  + Senses (vision, hearing, smell, taste, touch, kinesthetic)
* Process
  + Cognition
    - Knowledge
    - Skills
    - Reasoning
* Storage
  + Memory
* Output
  + Actions
  + Speech

**Computer system**

* Input
  + Keyboard, mouse, microphone, touch
* Output
  + Graphical display, speaker, vibrate
* Mostly output is to a visual display
  + And the display is also the input medium on phones etc
* Exceptions
  + Audio
  + Vibrate

**Physical reception**

* The eye: mechanism for receiving light and transforming it into electrical energy
* Light
  + Reflects from objects (e.g. paper)
  + Or is produced from a light source
* Images are focused upside-down on retina
  + Which is a nice example of how our brain further interprets the raw signal
* Retina contains rods for low light vision cones for colour vision
* Colour
  + Made up of hue, luminance and saturation
  + Cones are sensitive to colour wavelengths
  + Blue acuity is lowest
* Luminance(brightness)
  + Subjective reaction to levels of light
  + Affected by luminance of object, and by context of viewing
* Rods provide low-light monotone (grey scale) vision
  + Rods are saturated in bright light and contribute little or nothing to what you see
  + However, the principles of brightness and contrast are equally important with colour vision
* Visual acuity increases with luminance as does ability to perceive flicker

**Visual Acuity**

* Standardly measured in comparison to performance of person with 6/6 vision (or 20/20 in feet)
  + A person with 6/6 vision can discern contours of about 1.75mm at 6m distance
* Common problems are refractive errors such as light focusing in front of the retina (myopia) or uneven focusing of light on the retina (astigmatism)
  + Generally correctable with glasses, contacts or laser surgery and not a major problem for UI design.
* Presbyopia
  + From age 35 hardening of the lens of the eyes causes the eye to focus on light behind rather than on the retina when looking at close objects
* Macular degeneration
  + Loss of acuity in centre of visual field
  + Typically starts at over age 50; affects 12% of people age 80+
* Not everybody has up-to-date or ideal correction
  + And requires accommodation from the user
  + So, your design will provide better usability if you don’t default to a very small font.

**Depth Perception**

* We can perceive depth immediately with binocular vision using parallax, and through depth of field of focus
* Or we can synthesize the perception through overlap, our understanding of size or using motion parallax

**Making sense of the signal**

* Movement detection, tracking and object recognition
  + At simplest level the eye detects changes in luminance on a given part of the retina
  + However, many layers of synthesis in perception and interpretation combine to allow us to perceive an object moving across our visual field
* Optical illusions occur when our normal mechanisms for perception and interpretation are misled.
* Reading
  + Visual patterns perceived
  + Decoded using internal representation of language
  + Interpreted using knowledge of syntax, semantics, pragmatics
  + Involves saccades and fixations
  + Word shape is very important to recognition

# Lecture 13: Grouping

**Learning objectives**

* When and where is grouping valuable in UI design?
* What techniques can be used to sort things into groups?
  + How to achieve a conceptual organization of content/functionality?
  + At the level of objects on a screen, what visual clues can you use to indicate things belong together or apart?
* Gestalt principles of perception

**Sorting / grouping**

Various techniques can be used for sorting such as hierarchy, tree or a network.

* Card sorting is one of the most effective ways to group things

Lines and borders:

* Are immediate visual clues to grouping
* “lines” may be line objects or changes in background colour
* White space also has a similar effect (in keeping with the proximity principle)

High level: grouping screens

* If the sorted objects / concepts are going to be implemented as isolated components, then sort results translate into:
  + Menu structure
  + Site map

**Some organization schemes**

Exact schemes

* Alphabetical
* Chronological
* Geographic – alphabetic by name of locality

Subjective schemes

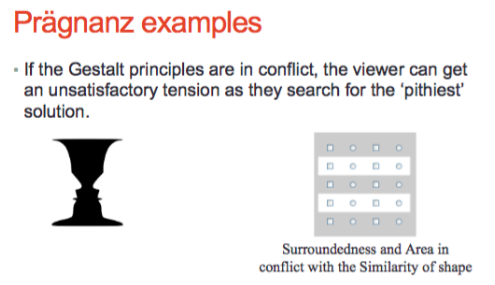
* Topic
* Task – organize content by considering the needs, actions, questions, or processes that users bring to that specific content
* Audience – by segments of your users
* Metaphor – relating content to familiar concepts can be tricky to devise, and a risk that the user doesn’t ‘get it’.

**Low level: grouping on a screen**

* “low-level” principles – used to make decisions about layout of the controls and content/data items on a screen
* Use visual cues to communicate the logical structure of the interface
  + We can do this through spatial proximity and other ways of leveraging perception grouping
  + We can do this by putting in lines and borders

**Gestalt principles of perception**

* Gestalt psychology strives to explain the factors involved in the way we group things
* At the heart of Gestalt psychology is the idea that we strive to find the simplest solutions to incomplete visual information
* The following is considered for making use of Gestalt perception
  + **Proximity** – our eyes are sensitive in making grouping judgements based on proximity in a collection of objects
  + **Similarity** – objects that have similar visual characteristics, such as size, shape or colour will be seen as a group and therefore related
  + **Common fate** – Objects that ‘move’ together (sharing a beginning and/or a direction and/or an end) are seen as related
  + **Closure** – we tend to see things as complete objects even though there may be gaps in the shape of the objects.
  + **Area** – Objects with small area tend to be seen as the figure, not the ground.
  + **Continuity** – we tend to see things as smooth, continuous representations rather than abrupt changes
  + **Symmetry** – symmetrical areas tend to be seen as complete figures that form around their middle.
  + **Surroundedness** – an area that is surrounded will be seen as the figure and the area that surrounds will be seen as the ground.
  + **Pragnanz** – as an overarching principle, we tend to perceive things based on the simplest and most stable or complete interpretation.



# Lecture 14: Lines and Borders

**Learning objectives**

* Relationship of wireframing to line/border design
* Why do you use lines and borders?
* What are some of the style options and why would I choose particular ones?
* How are lines and borders used in Data tables?

**Making Wireframes**

* Wireframing your site begins the process of pinning down a specific physical design that aligns to a connectional one.
* Hand-draw them or other electronic tools
* Basically, this is lo-fi prototyping, but here in we’re emphasizing the roll of deciding the layout as compared to seeking feedback in a broader sense.

**Lines and borders**

* Create physical grouping
* Borders offer possibilities for creative styling
* Many sites use very thin borders and separator lines that reflect the visual qualities of a body font.
* Likewise, hairlines are used in several ways in modern websites

**Corner treatments**

* Instead of using ordinary right angles, you can also use diagonals, curves or cutouts for some of the interface’s box corners
  + Make sure they are consistent across the interface

**Angles**

* A page composed of straight up-and-down lines and right angles generally looks calmer and more still than diagonal lines and non-rectangular shapes
* Likewise, a page with many different angles has more apparent motion.

**Curves**

* Curves can also add motion and liveliness but not always
* A design made with a lot of circles and circles arcs can be calming and restful
  + But a curve swiping through a page sets the whole design in motion, and a few carefully chosen curves in an otherwise rectangular design add sophistication and interest.

**Incorrect!**

* Adding lines between connected bits of information disrupts the visual flow
* Grouping incorrectly is promoting confusion of your users

Laying out data tables involves explicit decisions about: headings, font, alignment and of course lines/borders.

# Lecture 15: Text

**Learning objectives**

* Describe the main purposes of text in user interface designs
* Appreciate the balance between art fonts and readability
* Explain what makes a font readable
* Identify and utilize the characteristics of text that we can control
* Select an appropriate font scheme for a user interface design

**Text in Interaction Design**

* Commentary text in UI
  + Commentary text informs user of system status
* Instrumental
  + Instrumental text is text that works e.g. hyperlinks, button, labels/captions
* Whether it’s body content or part of UI
  + Legibility
    - Can the reader discern the words?
  + Readability
    - Can the reader easily read the text?
  + Physical factors
    - What is the screen resolution, brightness?
* We read lowercase more quickly than UPPERCASE
  + Uppercase is good for picking out specific word or characters at a distance
  + Lowercase presentation is more common

**Fonts for title vs. body**

* Decorative fonts (sometimes called display fonts or title fonts)
  + Suitable for titles and headings
  + Strong personality:
    - Grab people’s attention
    - Reinforce the message of the word
* Neutral looking fonts such as Georgia or Arial are more suitable as body copy (easier to read)
* Georgia, Verdana, and trebuchet maintain legibility at small sizes and have been designed to facilitate reading on the web.

**Font Size**

* Factors that affect perceived font size
  + Reading distance – greater distance requires larger text
  + Screen Resolution: smaller text requires greater resolution to keep the characters clear and legible
  + Text/background contrast – positive contrast is optimal
  + Visual acuity of user – not all users have 20/20 vision, and from age 40- something presbyopia really takes hold: focusing on near objects doesn’t work so well and hence small fonts require magnification
  + Purpose – text can be scanned, read word by word, or read character by character.
* Fonts for body copy usually don’t work well when set too large, they tend to come inelegant and clunky
* The opposite is the case when setting tilting fonts too small – the title/heading will lose its dominance and the page looks bland.
* Different x-heights cause fonts of the same size too look different.
* **Often use more stylish font in the title**
* Context of use (device, setting, purpose) how you’ve displayed it, user version, the font itself (good x-height) all play a part in what makes the font **readable**

**Weight and Style**

* Weight
  + Bold – when two fonts differ in weight, they form a strong and vibrant visual contrast
* Style
  + Italics – underlines (can be mistaken as a hyperlink, so not advisable to use them)
* Besides emphasizing points, creating contrast by varying weight and style can contribute to a dramatic and eye-catching look to an interface.
* However, use with restraint and be consistent
  + Too many weight and styles on one screen gets confusing.

**Alignment, Line length & margins**

* Use left or justified
* Right and centre are harder to read because you can’t find the beginning of the line
  + Right justification is good to glue caption text to a data entry field or figure to the right.
* Line length affects reading performance but not comprehension
* Lines of greater length are read more quickly
* But people prefer medium line lengths

We have the ability to control all these types of text:

* Font, size, weight, style, spacing and alignment

# Lecture 16: Colour & Images

**Learning objectives**

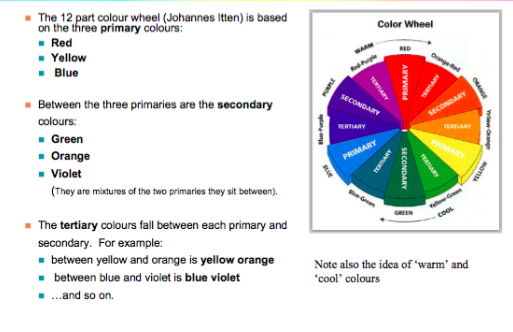
* Describe colour properties
  + Value
  + Hue
  + Saturation
* Describe and identify colour schemes
  + Monochromatic
  + Analogous and Complimentary
  + Ready-made colour schemes
* Explain how colour is used for branding
* Explain how images contribute to the colour scheme of a UI
* Apply colour principles to a UI design

**Colour**

* Colour has three distinct properties:
  + Value – lightness or darkness (luminance)
  + Hue – spectral colour name (blue, red)
  + Saturation – brightness or dullness
* **Value**: is defined as the perceived lightness or darkness or a colour
  + Value can be used to increase/decrease contrast consider these examples:
    - Low contrast, ‘low-key’
    - Low contrast, ‘high-key’
    - High contrast
    - High contrast (inversed)
  + Note: greater contrast makes the darker object more dominant
    - Hence, inverse is harder to read – should only be used for titles or emphasis
  + Create movement
    - Objects of the same value create a static design with all objects equal in visual importance
    - Varying values gives a more dynamic appearance and creates a ‘pecking order;
    - Mix elements of different values to add visual movement to your design or to create a hierarchy of importance

**Colour: Hue**

* Colours at the lower end of the spectrum are more comfortable to look at
* Based on vector value moving from 0 to 360 degrees on a colour wheel



**Colour: Saturation**

* Saturated colours don’t contain black, white and none of their complimentary or opposite colour.
* Intensity of colour in percentages scale: 100 percent is pure colour, 0 percent is black, white or grey.
* Highly saturated or pure, colours
  + E.g. brilliant yellows, reds, and greens
* Advantages:
  + Evoke energy, vividness, brightness and warmth
* Disadvantages:
  + When overused, they can tire the eyes; seem “childish”
* Muted colours make up the bulk of most colour palettes
  + Tints: adding white
  + Shades: adding black
  + Tones: adding some of both black and white

**Colour schemes**

* Monochromatic or monochrome
  + One hue (containing or using only one colour)
* Analogous
  + Hues near each other on the colour wheel
  + The hues may vary in value
  + Analogous colour schemes look harmonious
* Complementary
  + Hues opposite each other on the colour wheel such as red and green, yellow and purple.
  + Produces exiting dynamic pattern
* Others
  + Triadic, split complementary (complementary plus analogous hues on one of the complementary ones)
* Further, any of these types will use variations to achieve a specific “scheme” of colours based on their hue(s)
  + Various shades, tints and tones, including fully displacing the colour for black, white and shades of grey.
* Discordant colours are visually disturbing – we say they “clash”
  + Colours that are widely separated on the colour wheel are discordant
  + Can be eye-catching and are often used as attention-getting devices in **advertising**

**Colours scheme and images**

* The colours of images you choose can reflect upon the colour scheme of your interface
* Basing colour schemes around photos is also a great technique.
* Luminance (black and white) contrast is more significant than colour (hue contrast)
  + Hence a near-equal value red-on-blue scheme has nil contrast
  + Contrast sensitivity decreases with age
* Sometimes colours are used as a code to communicate information like a “traffic light”

High contrast is important for readability.

# Lecture 17: Colour & Images

**Learning objectives**

* To be effective in use of standard user interface components
  + Use containers for top-level grouping
  + Use buttons, icons and hyperlinks for tasks such as function/tool selection and navigation
    - Understand the issues in designing effective icons
  + Apply common data entry controls for appropriate roles
* To be able to combine appropriate use of controls and other visual interaction principles to design highly usable forms

**Containers for visual organization**

* Containers organize functionality and facilitate navigation
* Top-level containers:
  + Primary windows
    - Users main interaction with the application or document
  + Secondary windows
    - Dependent on a primary window
      * Important subclasses include modal dialogues, multiple document interface (MDI) document windows
  + Utility windows
    - Contents affect an active primary window
  + Plain windows
    - No title bar or window controls typically used for splash screens.
* Middle-Level and nestable containers
  + Panels
    - Used to group related functionality
  + Frames
    - Can be resized by dragging the splitters at their edges
    - May be minimized and possibly removed by tailoring options
  + Tabs
    - Increase the size of the dialogue by stacking layers on top of each other and allow more elements to be accessed from one dialogue
  + Stacked tabs move around to accommodate the different levels
    - This destroys location consistency

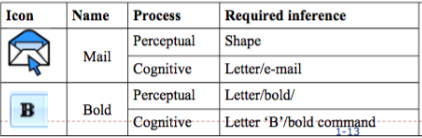
**Controls: Buttons and Icons**

Command buttons – have short labels

* Toolbar/palette buttons:
  + Function like command buttons, but they have icons instead of labels
  + They are usually grouped by function

**Icons versus Text**

* Icons
  + Are signs and as such represent a significant degree of cognitive complexity
  + If designed properly, they are quick to recognize
  + Otherwise, they risk being obscure and ambiguous
* Text buttons
  + Are less libel to be misinterpreted
  + But it takes longer to read a word
  + Takes more screen space
* Icon analysis
  + By examining the steps in interpreting an icon, you can gain inside into whether they will be usable
  + Perceptual
    - Recognizing the icon as an envelope
  + Cognitive
    - Mapping envelope image -> email
    - Mapping **B** -> Bold command



**Controls: Hyperlinks**

* Hypertext/hyperlinks
  + Have various states
    - Normal - e.g. blue
    - Active – e.g. red
    - Visited – e.g. purple
  + Must be obvious
  + Try to avoid “click here”
    - Not very useful
  + CSS can be used to change the colour of hyperlinks

**Data Entry**

* The right control for the type of data
  + Meets the simplicity principle through the tacit of constraints
  + Only take the input of the legal type and range, then you avoid all the interaction around error messages
* Principle of visibility with controls that make options apparent (list boxes, checkboxes etc)
* Using controls in the expected way gives predictability
* Text box/field:
  + Text box should be used when there is a need to gather small, discrete amounts of information
* Captions (or labels)
  + Passive components that guide the user in what to enter
* Combo box
  + Combination of a drop-down list or list box and a single-line textbox
  + User can either type a value directly into the control or choose from the list of existing options

**Support fast form completion**

* Keyboard entry is faster than mouse for most users
  + Support keyboard entry for all controls and navigation, especially for frequent users
  + Provide logical tab order throughout form
  + Support short-cuts for radio buttons and checkboxes
  + Optimize list box contents
* Provide shortcuts and accelerators for frequent users where there are multiple paths through the form
* Type of control (choosing an auto-complete are nearly always better than just free text)
* Spatial relationship between label and control
  + Left side is superior to above, below or right side
  + Logical order of data
    - Name, address, phone number

**Make good use of screen ‘real estate’**

* Screen space is at a premium (scrolling makes experience less visible and coherent for user and can waste time)
  + Minimize gaps between controls
  + Consider different strategies to indicate grouping
    - Lines, white space, background shading
  + But aim for the simplest feasible grid
  + Many times, it won’t all fit on one screen
    - Use card sorting or observation to decide what to put where
    - “less is more”

# Lecture 18: HTML for Prototyping

**Learning objectives**

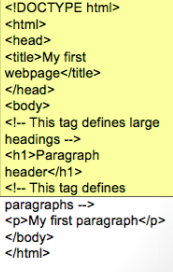
Learn how to write dynamic HTML for high-fidelity design prototypes

* Demonstrate layout and colour as it would appear in a full implementation
* Also demonstrate “dynamics” such as data entry control Behaviour, conditionally-visible elements, and links to other screens.

Includes:

* Combining the document object model (DOM) and web storage with JavaScript and cascading style sheets (CSS) to enable dynamic content and embed functionality in web pages.
* Implementing drag&drop and other event handling and on-screen movement in web pages using HTML, JavaScript and CSS

**Hypertext Markup language**

* The standard markup language used to create web pages
* Used along with CSS and JavaScript to create web pages, including user interfaces for mobile applications.
* Basic structure of html file ->

**What is DHTML?**

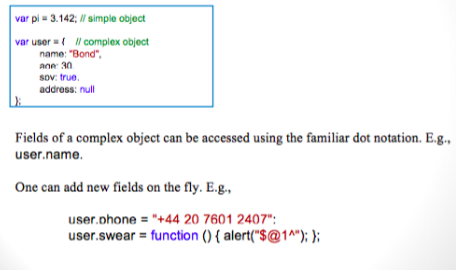
* Dynamic HTML
* DHTML is an umbrella term for techniques to create interaction and animation using
  + JavaScript
  + CSS
  + DOM
* DHTML activities include
  + Event detection (mouse over)
  + Form validation
  + Changing content
* Contrast with Ajax in that all processing is done on client side

**The Document Object Model(DOM)**

* The HTML DOM is a standard object model and programming interface for HTML. It defines:
  + The HTML elements as objects
  + The properties of all HTML elements
  + The methods to access all HTML elements
  + The events for all html elements
* Tree structured
* Mostly access using JavaScript

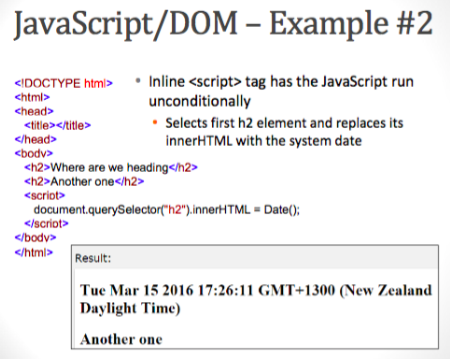
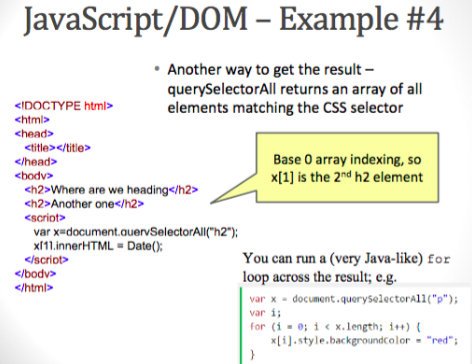
**JavaScript**

* Enables dynamism and tailored behavior in your web pages
* Some major uses in DHTML are:
  + Event handling
    - Creating custom behaviour for user actions on HTML elements
  + Accessing the DO
    - Basing login on the values of parts of the web page
    - Dynamically updating attributes of variables or adding and deleting elements
  + Accessing web storage
    - Persisting data in the browser for continuity
* Scope
  + A variable’s scope is the function it is declared in
  + A variable declared outside a function has a global scope
    - But assigning a value to a variable that has not been declared will mark the variable global
  + Value – flexible
    - Can have a single value or multiple fields
    - Can change type or add new fields on the fly
  + Variables are dynamically typed: the runtime infers the type
  + A variable can have different types during its life
  + The operator == checks if the compared values are equal (converting both values to a common type), while the other operator === checks if the compared values and their associate types are equal.



**CSS**

* CSS can control where an element appears
  + Position property can be static (default), relative, fixed or absolute
  + With relative position, you can indent
* An element with position: absolute is positioned relative to the nearest position ancestor (instead of position relative to the viewport, like fixed)
  + A positioned ancestor element being one with position other than static
* If an absolute positioned element has no positioned ancestors, it uses the document body, and moves along with page scrolling.
* Float allows content to flow around another element
* Note the margin property adds spacing (top, right, bottom and left, respectively) of displayed content away from other elements.



**Forms and Validation**

* HTML5 gives you convenient types for form input
  + Other convenient types: date, time, password
  + Note each browser brand will display a bit differently
  + Use CSS pseudoclasses to control appearance based on whether input meets the form’s validation.
* Constrain validation CSS Pseudo selectors
  + Selector – description
  + Invalid – selects input elements with invalid values
  + Optional – selects input elements with no “required” attribute specified
  + Required – selects input elements with the “required” attribute specified
  + Valid – selects input elements with valid values

**Handling events**

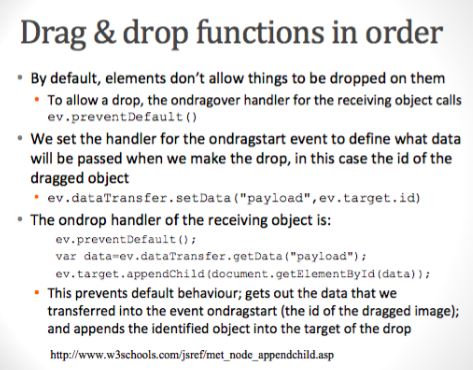
* You can give any button, including the submit input field of a form, an onclick value
  + Name it to a JavaScript function you’ve declared higher in the HTML
  + Can pass data to this event handler
* There are other events than onclick such as mouseover, mouseout, onblur etc.

**Embedding data in a Web page**

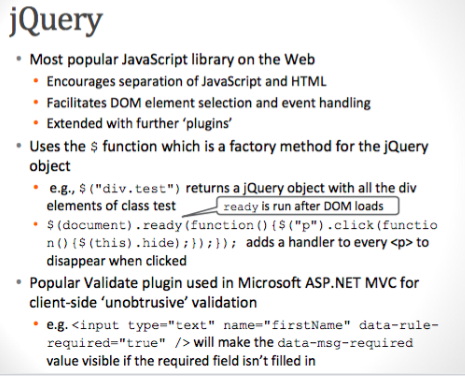
* User-defined attributes
  + HTML 5 endorses hidden data using custom attributes if you start your attribute names with ‘data- ‘

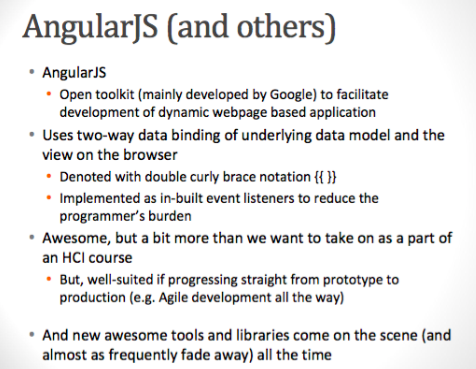
**Web Storage**

* HTML5 web storage provides two new objects for storing data on the client
  + Window.localstorage – stores data with no expiration date
  + Code.sessionStorage – stores data for one session (data is lost when the tab is closed)
* Faster and larger data limit than old method, ‘cookies’
* sessionStorage has setItem and getItem methods









# Lecture 19: Human Memory

**Learning objectives**

* Describe the major categories of human memory
* Describe the major organizational structure of long term memory
* Understand how to apply these organizational structures to make more usable UI designs

**Memory**

There are three main types of memory function:

* Sensory memories
  + Attention – short-term memory or working memory
  + Rehearsal – Long-term memory
* Selection of stimuli governed by level of ‘arousal’ (our level of interest or need) Some ‘interest’ is automatic

**Sensory memory**

* Buffers for stimuli received through senses
  + Iconic memory: visual stimuli
  + Echoic memory: aural stimuli
  + Haptic memory: tactile stimuli
* Continuously overwritten
* Decays exponentially and rapidly

**Iconic memory**

* Has two components:
  + Visual persistence (impossible colours)
  + Information persistence (deeper in the brain, associated with object motion)
* Key benefit is ability to recognize change in a visual scene

**Short-term memory (STM)**

* Scratch-pad for temporary recall
  + Rapid access roughly 70ms
  + Rapid decay roughly 200ms
  + Limited capacity – 7 2 chunks
* Some research suggests that programmers have better short-term memory than “average” people
  + Can relate to in-born difference and/or practice to cope with STM demands
  + A chunk is 1 item in short term memory

**Long-term memory (LTM)**

* Repository for all our knowledge
  + Slow access roughly 1/10 second
  + Slow decay, if any
  + Huge or unlimited capacity
* Two dominant types of access structure
  + Episodic – several memories of events
  + Semantic – structure memory of facts, concepts, skills
* Semantic memory structure
  + Provides access to information
  + Represents relationships between bits of information
  + Supports inference
* Model: semantic network
  + Inheritance – child nodes inherit properties of parent nodes
  + Relationships between bits of information explicit
  + Supports inference through inheritance

**Models of Long Term Memory – Frames**

* Information organized in data structures
* Slots in structure instantiated with values for instance of data
* Type-subtype relationships

**Models of LTM – Scripts**

Model of stereotypical information required to interpret situation. Script has elements that can be instantiated with values for context.

**Models of LTM – Production rules**

* LTM is not only ‘facts’ it is also ‘rules’
* Representation of procedural knowledge
* Condition/action rules
  + If condition is matched then use rule to determine action

**LTM storage of information**

* Rehearsal
  + Information moves from STM to LTM
* Total time hypothesis
  + Amount retained proportional to rehearsal time
* Distribution of practice effect
  + Optimized by spreading learning over time
* Structure, meaning and familiarity
  + Easier to remember things that makes sense to us

**LTM – forgetting**

* Decay
  + Information is lost gradually but very slowly
* Interference
  + New information replaces old: retroactive interference
  + Old may interfere with new: proactive interference (hard to “unlearn” a bad habit)
  + Applies to motor as well as cognitive recall

**LTM – retrieval**

* Recall
  + Reproducing information from memory can be assisted by cues
* Recognition
  + The presented information gives knowledge that it has been seen before
  + Less complex than recall – presented information is cue
* Spatial
  + Strong memory of where things are relative to other things

**What does this mean for HCI?**

* A few categories of applications
  + Don’t overload the user’s STM
  + Organize to make things seem simpler
  + Minimize the amount of LTM retrieval a user must do.
* Anything you ask the user to remember, loads their STM and essentially reducing their working memory for thinking about their task.
* Avoid creating interruptions
  + If you pop up a dialog then the STM from their task gets lost to deal with the new dialog!

**Minimize retrieval from LTM**

* A set of visible controls to be adjusted provides immediate cues to the options available
  + Very different from needing to recall command-line parameters
* A dropdown list or other visible selection of values allows user to recognize what they want

# Lecture 20: Sound

**Learning objectives**

* Describe the basics of human hearing
* Explain the difference between visual and auditory interaction
* Describe the classes and subclasses of sound output and the attributes of each
* Describe the classes and subclass of sound input and recognition and attributes of each

**Hearing**

* Physical apparatus:
  + Outer ear – protects inner and amplifies sound
  + Middle ear – transmits sound waves as vibrations
  + Inner eat – chemical transmitters are released and cause impulses in auditory nerve
* Sound
  + Pitch – sound frequency
  + Loudness – amplitude
  + Timbre – type of quality
* Humans can hear frequencies from 20Hz to 15kHz
  + Less accurate distinguishing high frequencies than low
  + Higher frequency hearing disappears as you get older
* Auditory system filters sound
  + Can attend to sounds over background noise
  + Hearing aids disrupt this filtering
* Hearing is involuntary
  + Suddenly ‘grabs’ attention before we think
  + And some sounds are harder to ignore

**Sound interaction**

Computer output/generation (input to human)

* Non-speech
* Speech

Computer input/recognition

* Speech
* Non-speech
  + Environmental
  + Music

**Computer Output: Music**

* Immersive experiences
  + Activates your brain in a different way from language
  + Acts almost entirely independently from hand-to-eye processing

**Auditory Icons and Earcons**

* The difference between these two is subtle
  + Auditory icons: emphasis on ‘natural’ sounds and metaphor with real world; caricatures of naturally occurring sounds
  + Earcons: “Artificial” sounds (generated)
    - E.g. more abstract metaphorical relationship to action or purely a convention.
* Redundant encoding
  + It aids memory by adding additional associations
  + Can alert without interrupting
  + An alternative communication channel
* Positive/Negative Feedback
  + Auditory alarms might be crucial to the safe operation of computer-operated machinery or mission-critical environments
  + But too many alarms may be:
    - Annoying
    - Ignored

**Using Sound in Interaction Design**

* Learnability of the mapping between the icon and the object represented

**Speech Recognition**

Two distinct applications:

* Transaction: giving commands, issuing queries, making menu selections
* Transcription: “typing” a document by voice
* Transaction
  + Telephone menu systems
* Automatic speech recognition (ASR)
  + Built into operating systems
* Difficult, ongoing research problem
  + Not just about recognizing phonemes (language sounds) but also finding the “right” interpretation

**Searching Speech and Audio**

* Sound files do not afford easy opportunities for indexing and searching
* Speech recognition can be used to transcribe speech files and create transcripts that can be searched like any other text file

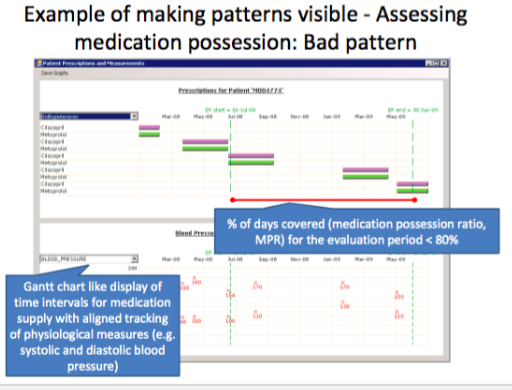
# Lecture 21: Data Visualisation

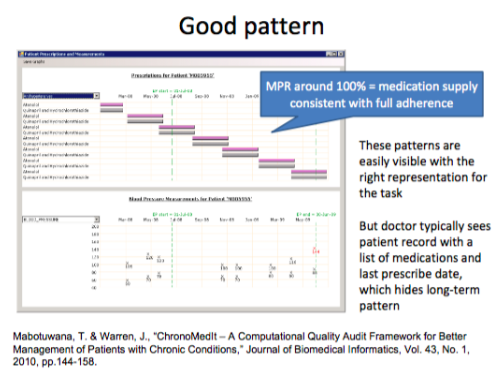
**Learning objectives**

* To be able to describe options for display for large and complex data to aid human understanding, including
  + Display of high-dimensional data and temporal change
  + Display of probabilistic linkage among elements
  + Use of animated and direct-manipulation displays to view successive slices of a large data set while moving through time or spatial dimensions

**Some domains with Big Data**

* Astronomy/physics
* Big data is when the data is so big that it’s serious computational challenge just to transfer and store it
* Other domains with lots of data
  + Weather and climate modelling
  + Biomedicine
    - Genomics, proteomics, metabolomics
  + Healthcare delivery
  + Retail and marketing
  + Finance and economic modelling
  + Security





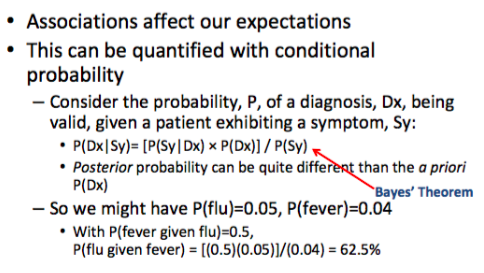
**What’s behind the prediction?**

* Logistic regression
  + Log of the odds of an outcome as a weighted function of a number of risk factors
  + Weights are learned by fitting to population health data
* For the scientific mind, seeing the 95% confidence interval of a Beta may be the way to go, but most people will appreciate the graphics.

**Modern Geographic information systems**

* Minard\* has inspired generations of analysts to look at displaying data on maps
* Particularly useful to see geographic variation

**Bayes Theorem**



**Probability in user interaction**

* Can use prior prevalence and posterior probability as basis for layout decisions
  + Can estimate contextually-likely actions for right-click options, or to offer help topics
* I developed Mediface a few years ago
  + Used general Practice electronic medical records to estimate prevalence and conditional probabilities on diagnoses, symptoms and treatments

The world is increasingly overflowing in data. Clever use of colour and layout lets us see high-dimensional data. Animations and direct manipulation give us a time-space experience of data. Probability estimates can be used to choose layout and menu options, and can be explored graphically to understand large data sets.

# Lecture 22: Design for Mobile

**Learning objectives**

* Identify the major challenges in designing web application for use on mobile devices
* Consider how mobile scenarios may differ from desktop scenarios
* Be aware of technical solutions to allow Web applications to be responsive to user display size.
* Take inspiration from successful patterns used by others.

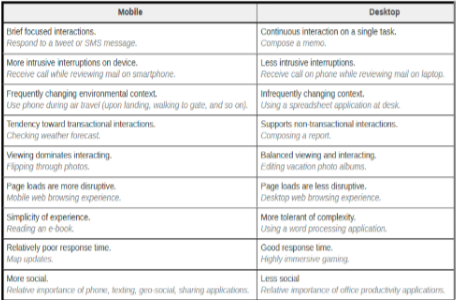
**Challenge: touch screen instead of mouse**

* Mouse and touch “feel” different
  + The mouse is a Fitt’s law device: you can drag the cursor through the screen place between its old position and the new target
  + A touchscreen promotes jumping to a new location
* The stack of events evoked are different
  + Typically, browsers map a touch to a sequence of traditional events
  + But there are also touch-specific events
    - Need to be careful not to run multiple event handlers for the same user action
  + Hover is particularly tricky
* And now touch PCs are becoming popular

**Challenge: system diversity**

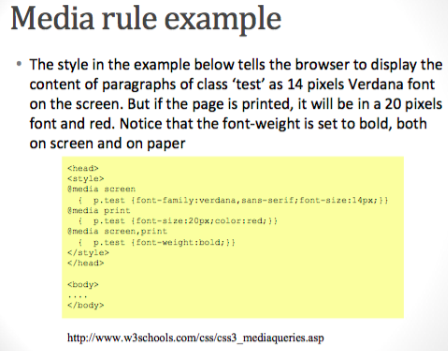
* IOS(Apple)
  + Uses gestures and widgets to move the user through views
  + Home button on the bezel used to close applications and navigate out of folders
* Android
  + Uses gestures, widgets and hardware buttons

**Multi-device interaction**

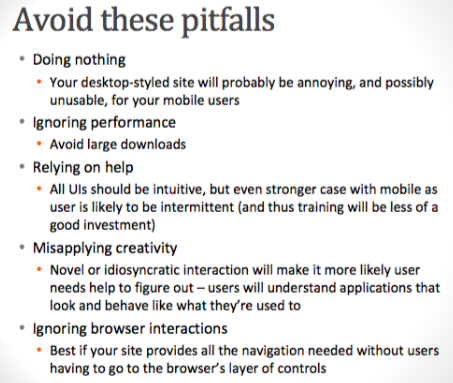
* Need to rethink the scenarios for your application with mobile in mind
* In many cases the user may sometimes use a desktop interface and sometimes a mobile one
  + On the whole, they will probably tend to use your application differently in each setting
  + The following table lists general tendencies in the contrast of mobile and desktop experience

**Media queries**

* Using the CSS media “rule” you can adapt presentation for handheld devices , printing and other formats
* Some potentially-useful media types never caught on
  + But can make media queries that give us enough information to tailor our content effectively



**Recycle**

* Successful patterns tend to represent:
  + Significant design effort by others
  + Darwinian selection
* Further, there’s the ‘external consistency’ advantage of looking and acting similar to what people are used to
  + Not the path of innovation but lead to efficiency
* Use of low saturation text to indicate desired input is an interesting convention
* Well-designed graphics will convey key points more readily than textual or tabular data

# Lecture 23: Frontiers of HCI

**Learning objectives**

* Describe haptics in terms of
  + Human perception
  + Applications
  + Devices
* Describe application of eye tracking and visual gesture recognition
* Describe the exploration of
  + Olfactory detection and production
  + Brain wave based user interfaces

**Haptics**

* Applying forces, vibrations, or motions to the user

**The Human perceptual system**

* Physical aspects of perception
  + Touch (tactile/cutaneous)
    - Located in the skin, enables us to feel
      * Texture
      * Heat
      * Pain
  + Movement (kinesthetic/proprioceptive)
    - The location of your body and its appendages
    - The direction and speed of your movements
  + Proprioception
    - We use sensation from our joints and our muscles to determine the position of our limbs and perceive body position
  + Combine with vestibular system to perceive motion, orientation and acceleration
    - This combination is sometimes called the kinesthetic sense

**Mobile devices also take kinetic input**

* Phone input
  + Touch screens
  + Accelerometer – shaking actions
  + Passive input
    - GPS
    - Altimeter, temperature, humidity
    - Specialized fitness or medical monitors
      * Step counter, sleep quality measures
* Phone output
  + Vibrate –silent alert

**Using haptics in interaction design**

* Medical uses
  + Surgeon controls ‘robot’ with zoomed view and automated enhancements over manual surgery

**Eye Tracking**

* Most consumer friendly modern method uses infrared light reflected off different parts of the eye to detect angle of gaze.
  + Measures of these angles from multiple locations, combined with measure of head position, allow estimation of gaze point on screen.

**Eye tracking applications**

* Using eye tracking to estimate gaze point over time provides rich insight into how users consume a visual presentation
* Rather more limited as an input method
  + Careful control of eye gaze to act as pointer can result in eye strain
  + But still useful for people with disabilities
  + Can use dwell time to indicate click, but error-prone.
* Room for further research in combining with other input methods

**Visual gesture recognition**

* OpenKinect – sort of pirate community making API to use Xbox Kinect hardware widely
* Makes skeleton from video and learns gestures for control
  + Kinect reasonably well received as a video game enhancer

**Olfactory – Odour /Smell**

* Smell is essentially our ability to detect specific chemical particles in the air
* We can detect about 4000 different smells
* And they can be combined in millions of different ways
* Smell is very deep in our brain

**Smell – current research**

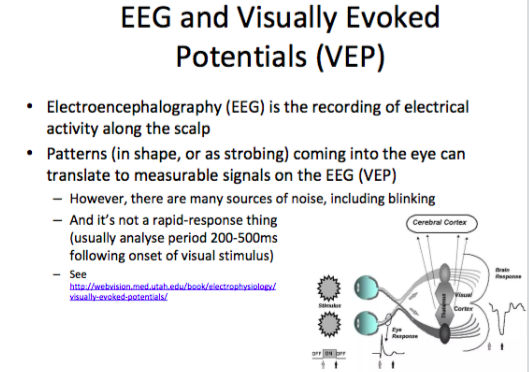
* Using sounds and smell signatures to aid recall of and affinity with individuals
* A wire in the glasses heats 8 perfumes to release a scent.

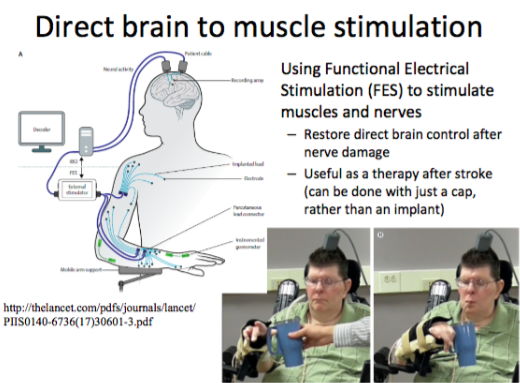
**Technology of Odour**

* Input
  + Detecting particular chemicals is possible
    - Drug/explosive sniffers
  + Detecting the range of smells in anything like human terms is extremely difficult task
* Outputs
  + Manufacturing particular smells possible
  + Active generation of a range of smells very difficult, but choosing a single smell to assert branding and positive associations for a retail outlet or such is already done.

**Brain Computer Interaction**

* Detecting the brain waves and interpreting
* From outside the skull – not very accurate
* Inside the skull – accurate but invasive





# ` Lecture 24: Gamification

**Learning objectives**

* Identify and consider use of game elements in user interface design
* Break down gamification into
  + Gamification objectives
  + Types of user motivation
  + Interaction patterns and user interfaces elements
* Recognize more, and less, successful gamification in applications
* Appreciate the growing blur in the game/reality boundary through the Internet of Things

**Motivation and definition**

* Games have always had an aspect of functioning as training tools
* The immense popularity of video games
  + They must be doing something right
  + Global video game market revenue
  + Average gamer age 13+
* Gamification is the use of game design elements in non-game contexts

**Gamification objectives**

* Be more compelling than the competitor’s product
* Get the user to use the application more and so they
  + Stay with the programme longer
  + Get in more training
  + Get fed more of our messages
* Guide the user to specific behaviors
* Get the user to try harder

**What’s fun, anyway?**

* Two approaches
  + Play
    - Exploration curiosity, free-form
  + Game proper
    - Competitive strife toward goals
* Elements of fun
  + Exploring, socialize, achieving, learning, collecting, surprise, humour, crafting or
  + A fun game provides some of these
  + Different elements appeal to different people, and at different times

**Patterns**

* Some major themes are popular to make an experience into a “game”
* Progress
  + The idea that you’ve able to have explicit advancement that persists from session to session
  + Keeps you coming back
  + Could be general or more fine-grained
* Quest
  + A specific goal/challenge
  + Designed to be hard but do-able
  + Quest variant: might take the form of a “Race”
  + Another variant a “puzzle”
  + Culminates with that victory moment!

**Basic elements**

* Levels
  + A beginner level for ease of getting started – may provide guidance / tutorial
  + Regular advanced levels with progressively greater features, challenges and/or rewards
  + Straightforward element for progress
* Points
  + A score that achievements get mapped to
  + May be compared to scores of others, or used to track your own improvement between sessions
  + May be redeemable for physical goods
  + Considered a naïve game element – not very rewarding without links to other elements.
* Badges
  + Acknowledges a specific achievement
  + May be unique
  + A thematic set of badges can be fun by way of collecting

**Some elements for upping the fun (and intensity)**

* Collecting
  + Hidden or defended items, or items that cost points
  + Can be represented by badges or otherwise be in the player’s inventory
  + Fundamentally satisfying to complete a collection, may be tied to completing a level
* Time pressure
  + Simple way to drive up user intensity
  + May be represented by metaphors such a bar of decreasing size and increasingly urgent coloration or sound like an accelerating heartbeat
* Modifier

**Elements for a Virtual World**

* Avatars
  + Projecting a consistent visual representation / persona for the game world
  + Fun to pretend
* Teams
  + Forming teams of players to take on a quest
* 3D and physics
  + Making the game world share properties with the real one
  + Solid
  + Newtonian physics: momentum, action/reaction, gravity
  + Fun, and aid to learnability

**Breaking the game-reality boundary**

* Real activities having meaning in the game
  + Directly as with exercycle, or to learn game rewards/unlock game abilities through real-word achievements
* Game achievements being redeemable for real rewards
* Searching for objects in the real world
* Augmented reality
  + Overlaying computer graphics on live image of the real world
* Solving real problems in games
  + Protein folding

**The Internet of Things (IOT)**

* The internet reaching into physical world
  + Through cheaper sensors that get put all kinds of strange place
  + Better ability to connect sensors to the larger network
* Enables deep inter-penetration of video games and reality

**RFID**

* Radio Frequency Identification
  + Automatic identification method, relying on storing and remotely retrieving data using device called RFID tags or transponders
* AN RFID tag is a small object that can be attached to or incorporated into a product, animal, or person
  + RFID tag contain silicon chips and antennas to enable them to receive and respond to radio-frequency queries from and RFID transreceiver
  + Tag is size of a grain of rice

# Lecture 24: Agents and Interaction

**Learning objectives**

* Understand and recognize the agent paradigm in human-computer interaction
  + Agents, critics and mixed-initiative
* Awareness of methods and potential for intelligent tutoring systems
* Consider search and social media as agent paradigms
* See how this comes together in modern system development

**Agents v Direct manipulation**

* Point and click interface treat systems like tools
* But another approach is to treat systems like an agent
  + This is always present to some degree when using a command language, especially to the degree it resembles natural language
* It’s just a ‘stance’ – a mental model – held by the user
  + But it can provide guidance as an interaction paradigm and lead to any of a host of compelling metaphors.

**User Model and “Agency”**

* What we need to make the dialogue better – more convincing, more helpful – is more explicit knowledge in the system
* Any knowledge the system stores about the user can be termed its user model
  + In particular knowledge of user goals helps the system to function as an agent
* A software component is only considered an ‘agent’ if it has a degree of autonomy in undertaking an action for the user.

**Avoiding the Greek Oracle effect**

* In many domains either the user does not want the system to give them ‘the answer’ or indeed we are unable to make a system to do that
* Critics represent a type of collaboration where the human generates and the system provides evaluation

**Alert fatigue**

* Too much criticism, or too naïve of criticism can cause the user to “turn off”
* We need to detect the user’s goal

**Changing the user behaviour**

* Rather than having strict “agent” perspective, we might want to change the user’s goals and behaviour
  + This could be our approach if we’re implementing a component that will try to sell the user our product
* Or our goal might be on the benevolent side, such as population health or education goal
  + To educate the user (build effective beliefs and correct false ones)

**Information Retrieval (IR)**

* IR is an interaction paradigm
  + You ask a question you get an answer
  + The more a search engine tailor’s results to you the more it’s using its user model

