

# Objectives

- Usability
- Project Discussion



May 9, 2016

Sprenkle - CSCI335

# JavaScript review

- True or False: JavaScript is just like Java
- How do you declare a variable? (2 ways)
- How do you write text to the web page?
- What is the syntax for functions?
- What are some examples of events?
- How do you access a particular element in a document?
  - What are some ways to change that element?

What have we talked about usability so far?

# User-Friendly

- The term **user-friendly** is over-used and under-defined
  - What is “friendly” to one person may be trite, tedious, or confusing to another
- “User appropriate” is a much more meaningful term
  - But we have to know the user

# Usability

- Engineering principles for designing and building software interfaces that are
  - Fast to learn
  - Speedy to use
  - Avoid user errors
- How to recognize and articulate the difference between “this program sucks” and “I can improve this program by changing X,Y, and Z”
- Life-long habits for engineering usable products

# Fundamental Software Design Principle: the $7 \pm 2$ Rule

- Human's short-term memory can only hold about seven things at a time (plus or minus 2)
- When we get more than about 7 items, we get confused

# Shneiderman's Measurable Criteria

- User interface design has long been considered an **art** rather than a **science**
  - Decisions made **subjectively** rather than **objectively**
- There has been a lot of effort to make UI design more **objective**
  - an **engineering activity**

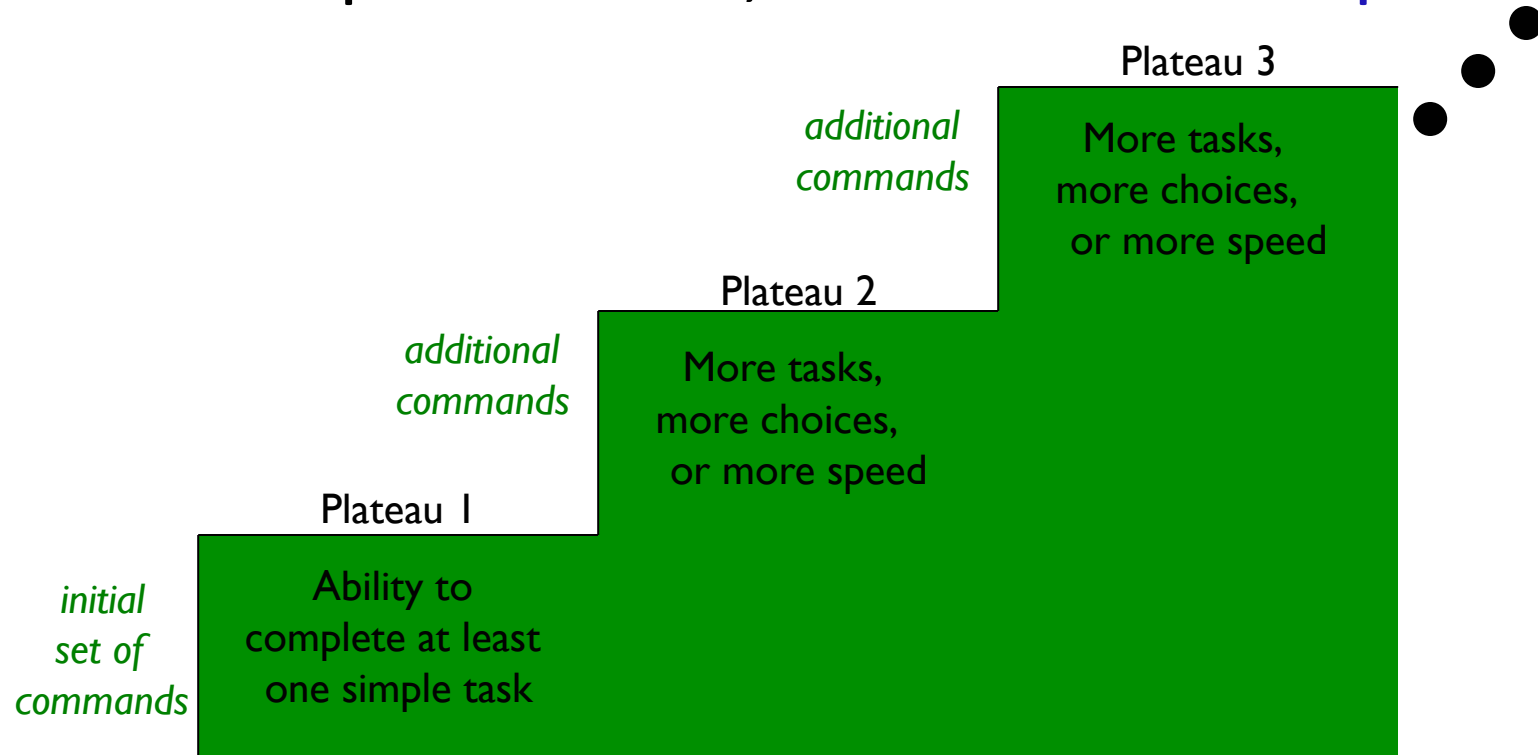
# Shneiderman's Measurable Criteria

1. **Time to learn** : The time it takes to learn some basic level of skills
2. **Speed of UI performance** : Number of UI “interactions” it takes to accomplish tasks
3. **Rate of user errors** : How often users make mistakes
4. **Retention of skills** : How well users remember how to use the UI after not using for a time
5. **Subjective satisfaction** : The lack of annoying features



# 1. Time to Learn

- With complicated UIs, the users must “plateau”



- Well-designed interfaces make
  - the first plateau easy to get to
  - subsequent plateaus clearly available

## 2. Speed of UI Performance

- This is about **navigating** through the interface, **not** how fast the software or network runs
- *Interaction points* are places where the users interact with the software:
  - Buttons
  - Text boxes
  - Commands
- Speed of UI performance is roughly the **number of interactions** needed to accomplish a task

## 2. Speed of UI Performance:

### The tyranny of the mouse

- The simplest way to **slow down** a UI is to use the **mouse**
- The mouse is incredibly slow: Most users can type between **8 to 15 keystrokes** in the time it takes to move the hand from the keyboard to the mouse
  - The two activities use different muscles and parts of the brain
- Good UI designers need to reduce the number of **keyboard-to-mouse** switches

# 3. Rate of User Errors

- Users will always make mistakes
- UIs can encourage or discourage mistakes
  - Consistency, instructions, navigation, ...
- Consider :
  - C/C++ : The lack of typing, particularly on pointers, and the complexity of the syntax actively encourages programmers to make mistakes. Thus we become debuggers, not programmers
  - Unix : The large, complicated command language encourages many mistakes as a result of simple typos and confusion
  - Entering grades in a dropdown instead of radio buttons

## 4. Retention of Skills

- “Once you learn to ride a **bicycle**, you never forget”
- Some interfaces are **easy to remember**, some are hard
- If they **flow logically** (that is, match the user’s **mental model** or expectations), they are very easy to remember
- If an interface is very **easy to learn**, then the retention is not important
  - **users can just learn again**

# 5. Subjective Satisfaction

- Subjective satisfaction is how much the users “like” the UI
  - depends on the user (thus the word “subjective”)
- Think of it in reverse: Users are **unhappy** when there is something annoying in the interface
  - Blinking
  - Ugly colors
  - Spelling errors in messages
- Most important in **competitive** software systems
  - Like ... everything on the Web !

## Celebrate SPCA Style

**Birthday Parties for children 3 and up**

**Party times available are: Saturdays 11:00-1:00 or 1:30-3:30 and Sundays 1:00-3:00**

Select a Dog Diggity Party, Cat Critter Party or a Dog and Cat Furry Friends Party.

**Party Packages Include:** 2 hour party time slot, party hostess, SPCA Furry Friends, Party Activities, Party Barn with table and chairs, SPCA water bottle souvenir for party goers and Invitations.

**Price:** \$150.00 for up to 8 children. \$5.00 for each additional child. And a \$50 non-refundable deposit when booking a party.

**MAKE YOUR RESERVATION TODAY**

# Establishing Criteria Priorities

Before designing,  
decide what is acceptable  
for each of the five criteria

- Order of priorities
- Minimally acceptable
- Optimistic goal



# Have a Point, Make Your Point!

You have less than two minutes to convince first time visitors to stay on your web site

Every page must justify  
**WHY** the user should stay

# MENUS

# Web Interfaces

- Web interfaces are composed of
  - Menus
  - Forms
  - GUIs

# Use Menus When...

- A finite list of well-defined choices
- Users will understand choices without help
- Users need to be reminded what they want

# Menu Considerations

- Menu system structure
- Number of items
- Sequence of items
- Titles
- Prompts
- Phrasing of items
- Online help
- Selection

# Menu Types

- Binary
- Multiple-item
- Extended
- Pop-up
- Permanent
- Multiple selection

# Binary Menus

- One of two choices can be made (Yes or No)

Do you want another transaction (Y/N)?

- Brief
- Concise
- But not descriptive

# Binary Menu: Alternative

You may

1. Open a new account
2. Login to an existing account

- More space
- Just as fast
- Instructions clear
- Results are clear
- Uses conversational dialogue, not stilted formalism

Possibly a little too wordy



# Multiple-Item Menus

- Only one item can be chosen

**What would you like to do?**

Withdraw

Deposit

Check Balance

Quit

Select your choice.



Do we need this?  
Clear without it?

# Ordering Menu Items

- Frequently used first
  - Most common strategy
- Numeric
- Alphabetic
- Chronological (time)
- Grouping of related items

# Adaptive Menus

- An adaptive menu changes to adapt to the user's habits.
  - Reorders the menu choices
  - Infrequently used items put in a “background” menu (as in MS Office 2000)
- Risky
  - Can be confusing
    - Different when returning to site
  - Different for different users
    - No “average” or “typical” user
  - (Old) empirical evidence was not positive

# Screen Design Hints

- Legibility
  - Sans-serif
  - Upper & lower case
- Titles
  - Alone
  - Top middle or top left
  - Titles & selections--same text !
  - Same titles in documentation
- Text
  - Brief, descriptive and consistent grammatical style

# More Screen Design Hints

- Instructions
  - At top, concise
  - Consistent on each screen
  - Offer help
- Navigation
  - Escape labeled—how to get out
- Options
  - How to make choices
  - Few options (about 7)
  - Numbering (we measure from 0, number from 1)

# More Screen Design Hints

- Consistent Layout
  - Menu status always in the same place
  - Error messages in the same place
- Take screen size into account
  - Responsive design
- Left-justify items
- No irrelevant info
  - Get rid of “happy talk”
- No strange codes or symbols

# Phrasing of Menu Choices

- Familiar
- Consistent
- Distinct
- Concise
- Use the keyword first
  - Improves accessibility

# Clarity vs. Social Amenities

- "Please", "do you wish", "If you want" can be eliminated to improve clarity
- Questions can be implicit

Please select the criteria for class choice:

VS.

Class choice criteria:



# Wordy Version

Please enter course or name:

If entering course, also enter dept:

(Enter category for level 2 reports only)

1. Add a class
2. Drop a class
3. Change section

Can we make this menu better?

What are the real goals/options of this menu?

# Simpler Version

## CLASS REGISTRATION

1. Add
2. Drop
3. Change section

Registration options:


Class Number or Name: \_ \_ \_ \_ \_

Dept Code: \_ \_ \_

# Design for Common Data

<b>Name:</b>	First: _____
	Last: _____
<b>Address:</b>	Street _____
	Box or number _____
	State ____
	Zip _ _ _ _ _

Allow for all  
types of  
addresses



How can we make the menu more general?

# Better Design

<b>Name</b>	First: _____
	Last: _____
<b>Address</b>	Address 1 _____
	Address 2 _____
	State ____
	Zip _ _ _ _ _

More flexibility



# Homework: Analysis of Usability

- Answer “Trunk test” questions for four pages
  - What site is this? (Site ID)
  - What page am I on? (Page name)
  - What are the major **sections** of this site?
  - What are my options at this level? (Local navigation)
  - Where am I in the grand scheme of things?
  - How can I search?
- Find examples of sites that demonstrate good and poor usability

# Looking Ahead: Project

- JIRA
  - Requirements
  - TODO lists – assign to team member
- Tomorrow morning
  - Present to Client
    - P413

# To Do

- HW - Analysis of Usability
- High-priority functionality: Wed midnight
  - Put “issues” in Jira
- Exam: Thursday a.m.