CS5340
Human-Computer Interaction

Nov 2, 2016

Today

- Admin
- Design Frameworks
  - Postures, Patterns
- Interface design paradigms
  - Metaphors
- Break
- Hot Topics
- Research Paper

Admin

- I2 : Nov. 9 by 6pm
- T4 : Nov. 16 by 6pm
- I3 : Nov. 22 by 11:59pm

- T5: Final presentations: Dec. 7
- T6: Final prototype + report: Dec. 11 @ 11:59pm

Creating

DESIGN FRAMEWORKS

1. Form factor, posture & input methods

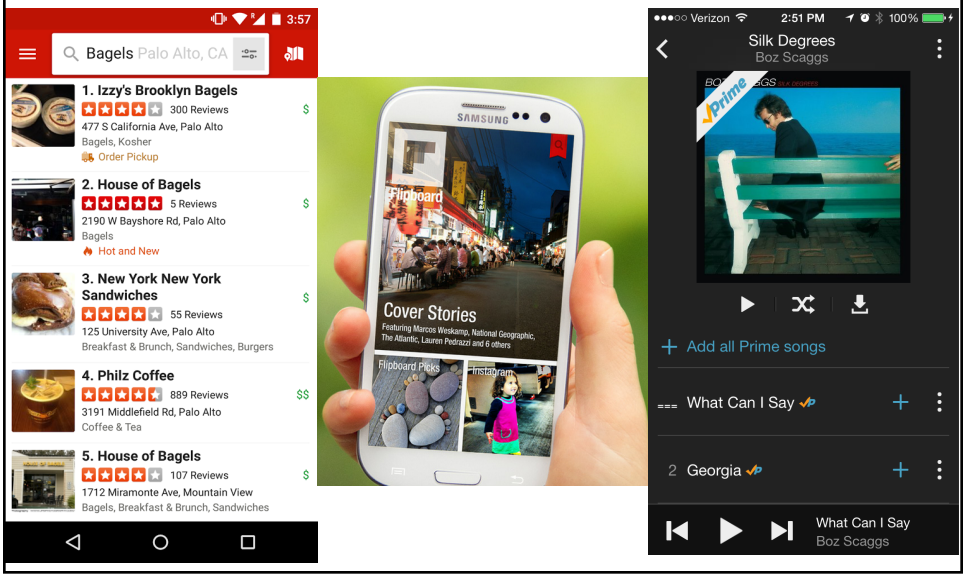
Posture

- Goal-directed behavioral stance
  - The way in which an interface presents itself to a user in look & feel
  - this presentation should be deliberate, conducive to the user's goals
- Postures may shift depending on
  - Platform
  - Context of use

Posture

- Sovereign
  - Engage users' attention for extended, continuous time periods
- Transient
  - Single or few functions w/constrained set of controls
  - Appears when needed, then quickly disappears

Mobile Postures



Satellite Posture

2. Functional & Data Elements

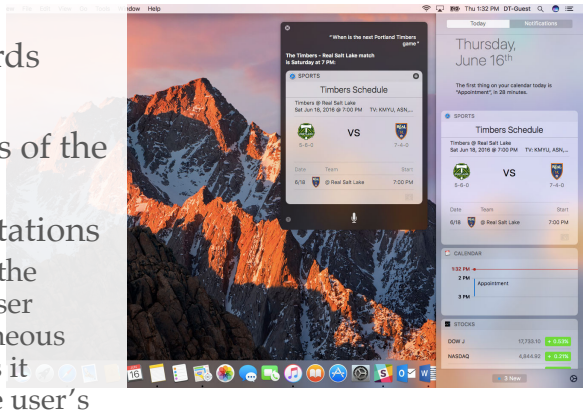
- Pretend the product is human
  - Thoughtful vs. inconsiderate design?
- Apply principles & patterns

Design Principles & Goals

- **Learnability**
  - Memorability
  - Consistency
- **Flexibility**
  - Task migratability
  - Dialogue initiative
  - Multi-modality
- **Robustness**
  - Observability
  - Recoverability
  - Constraints
  - Responsiveness
  - Task Conformance
  - Feedback

Design Principles - macOS

- Consistency
  - macOS standards
  - The app itself
  - Earlier versions of the app
  - People's expectations
    - "Does it meet the needs of the user without extraneous features? Does it conform to the user's mental model?"



https://developer.apple.com/library/content/documentation/UserExperience/Conceptual/OSXHIGuidelines/DesignPrinciples.html

Design Principles - macOS

- User control
  - the user, not the computer, should initiate and control actions
  - Challenges?
  - Progressive disclosure
    - Hiding more complex features until user requests them

https://developer.apple.com/library/content/documentation/UserExperience/Conceptual/OSXHIGuidelines/DesignPrinciples.html

Design Principles - macOS

- Forgiveness
  - "encourages people to explore without fear, because it means that most actions can easily be reversed"

https://developer.apple.com/library/content/documentation/UserExperience/Conceptual/OSXHIGuidelines/DesignPrinciples.html

Design Principles - Facebook

- Universal
  - "Our mission is to make the entire world more open, and this means reaching every corner, every person.

    So our design needs to work for everyone, every culture, every language, every device, every stage of life.

    This is why we build products that work for 90% of users and cut away features that only work for just a minority, even if we step back in the short term."

A new feature that would Adhere to this principle? Violate this principle?

https://www.facebook.com/notes/facebook-design/facebook-design-principles/118951047792/

Design Principles - Facebook

- Transparent
  - "Users trust us with their identity, their photos, their thoughts and conversation.

    We reciprocate with the utmost honesty and transparency.

    We are clear and up front about what's happening and why"

https://www.facebook.com/notes/facebook-design/facebook-design-principles/118951047792/

Design Principles - Facebook

- Consistent
  - "We invest our time wisely, by embracing patterns, recognizing that our usability is greatly improved when similar parts are expressed in similar ways.

    Our interactions speak to users with a single voice, building trust.

    Reduce, reuse, don't redesign."

https://www.facebook.com/notes/facebook-design/facebook-design-principles/118951047792/

Interaction Design Patterns

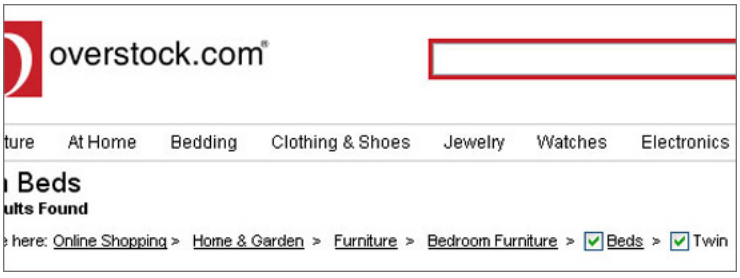
- Patterns…
  - Templates for how to solve a problem
  - Support communication + shared language
  - Speeds process
  - Defined to be applicable in design situations sharing similar contexts, constraints etc.
- Interaction design patterns guide
  - structure & organization of design elements
  - changes in design elements, in response to user action
- Focus on human implications
  - Design for user satisfaction
  - Distinguishes from software engineering patterns
    - Focus on reuse + standardization of code

Interaction Design Patterns

- Pattern library
  - Set of patterns organized by context in which they are applicable
    - Social media
    - Navigation
    - Search
    - Data manipulation
    - Shopping

Breadcrumbs

- Problem
  - Users need to know where they are in a hierarchical structure and navigate back to higher levels in the hierarchy
- Solution
  - Show the hierarchical path from the top level to the current page and make each step clickable



http://ui-patterns.com/patterns/

Accordion

- Problem
  - too many items to fit into a limited space
  - the number of items, if displayed all at once, would overwhelm the user
  - Supporting access to all items, in digestible chunks, without scroll
    - (which could remove user from preferred context/page position)

https://developer.yahoo.com/ypatterns/

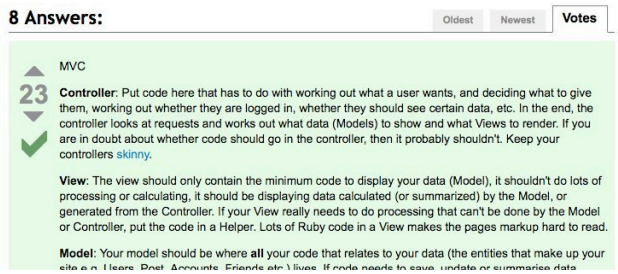
Accordion

- Solution
  - Grouped set of collapsible panels
- Recommendations
  - Have the most important panel open by default
    - expose the important choices
    - Show that each collapsed list can be individually opened
  - Highlight the current panel
    - Help user distinguish open panel headers from closed panel headers

https://developer.yahoo.com/ypatterns/

Vote to Promote

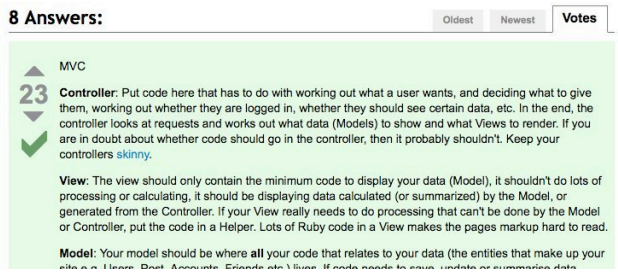
- Problem
  - High response volume can make it difficult for viewers to parse content or decide where to start
  - Users want to promote a specific piece of content in order to democratically help decide what content is more popular



http://ui-patterns.com/patterns/

Vote to Promote

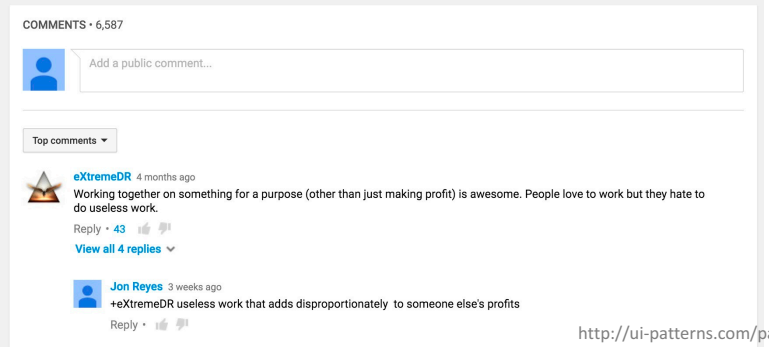
- Use when…
  - Want to encourage demographic participation (submission + evaluation)
  - You trust users' subjective evaluations
  - You have a large enough user base
    - Generate enough votes where meaningful comparisons can be made



http://ui-patterns.com/patterns/

Progressive Disclosure

- Problem
  - Users wants to focus on the task at hand with few distractions
  - Having option to get more detail if necessary



http://ui-patterns.com/patterns/

Google Material

"We challenged ourselves to create a visual language for our users that synthesizes the classic principles of good design with the innovation and possibility of technology and science. This is material design."

Design Language
Broad Principles
Patterns
Visual design guidelines

Swipe to Refresh

- Manual content reloading, to supplement syncing

Swipe to refresh

Swipe to refresh manually refreshes screen content with a user action or gesture.

There are two methods for updating content in an app:

- The preferred method is to automatically update content using sync, which keeps app content automatically updated.
- **Swipe to refresh** is a swipe gesture available at the beginning of lists, grid lists, and card collections that are sorted by recent content.

Manual refreshing can supplement syncing. It maintains the current scroll position, as when checking for new mail in Gmail.

Icon

The refresh indicator is represented by a **circular spinner,** an icon with a curved arrow that spins in a circular motion.

Position

- The **swipe to refresh gesture** is available at the top or bottom of content collections.
- The **refresh indicator's** resting position is centered horizontally relative to refreshing content.

Threshold

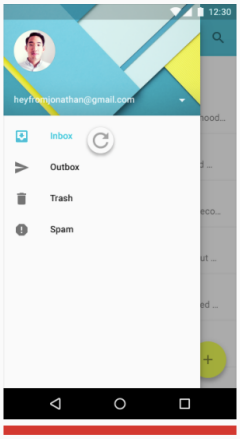
The refresh indicator must pass a threshold before an app refreshes, as indicated by the circular spinner's opacity, speed, and translation changes.

Swipe to Refresh

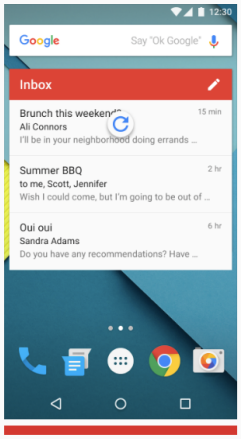
- Manual content reloading, to supplement syncing

Swipe to refresh should not be used in the following situations:

- Navigation drawers
- Home screen widgets
- Pannable content



Don't.
Navigation drawers (if present in an app) contain navigation destinations, not dynamic content.



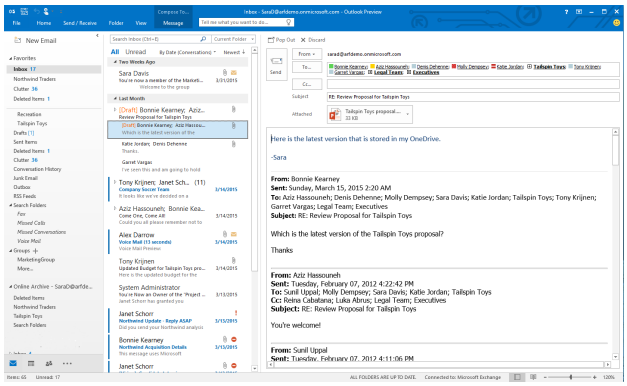
Don't.
Home screen widgets should update content automatically.

Patterns

- Not all patterns will be useful in every interface
- Consider
  - User goals
  - Constraints
    - Technical
    - Environmental
      - Social
      - Physical
    - Individual
      - Abilities

3. Determine Functional groups & hierarchy

- Screens, Views
- Panes
  - Independent views (panes) shown within a single window
  - Separated by fixed or movable dividers (splitters)



4. Sketch interaction framework

- Don't get caught up in details
- Easier to discard when little time spent

https://cs2024.wordpress.com/2015/10/02/user-experience-design-the-importance-of-wireframe/

5. Construct key path scenarios

- describes how the persona interacts with the product using the vocabulary of the interaction framework

- How compare to context scenarios?
  - More task-oriented
  - More detailed, using interaction framework language

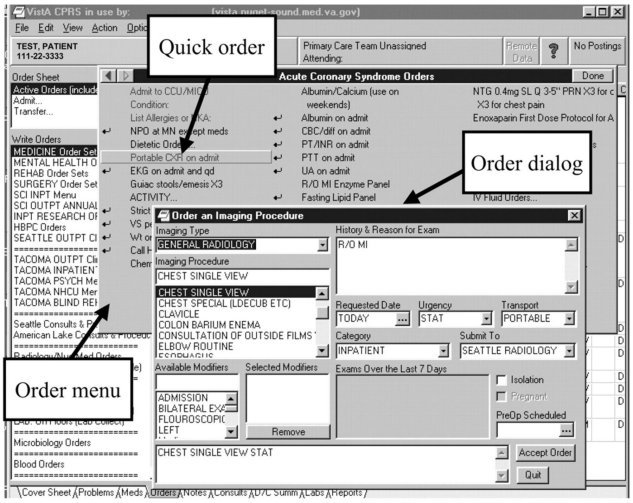
6. Check designs with validation scenarios

- Key characteristics?
  - Poke holes
  - What if questions
  - Less detail than key-path scenarios
- Types?

Interface Paradigms

Interface Paradigms

Implementation-centric



http://jamia.oxfordjournals.org/content/10/4/322

Interface Paradigms

Metaphoric Interfaces

- Interaction with system is designed to reflect something from the real world that the user is already familiar with
  - but also has own properties
- Intuition
  - Inference: see connections between disparate things (Uis & real world), despite differences
- Suggest how you might use something
  - Mental models
  - Suggest a *style* of interaction

Interaction Metaphors

- data as **files** (in folders or directories)
- deleting a file as throwing it in the **trash**
- programming as building **objects**
- Adobe Flash content played back on a **stage**
- **Shopping cart** icons in Amazon etc.

Interaction Metaphors

- can be based on activity, object or a combination of both
- can be mixed
  - e.g., garbage cans and desktops
- one metaphor is better than another if…
  - it leads users to more correct predictions about a system's behavior

Global Metaphors

- A metaphor that governs the entire user experience
  - Lower level metaphors consistent with this larger metaphor
- Does not take advantage of the power of a computational environment
  - To simplify tasks
  - Allow shortcuts
  - Alternate ways of doing things

Interaction Metaphors: Challenges

- Can be misleading
  - Break conventional and cultural rules
  - e.g. recycle bin placed on desktop
- Some things don't seem to have any obvious metaphor
- Can constrain designers in the way they conceptualize a problem space
  - Can limit imagination
- May use bad existing designs and transfer the bad parts over
- May become dated

Interface Paradigms

Idiomatic Interfaces

- Do not provoke associative connections, meanings are *learned*

- Examples of good & poor metaphors and idioms in currently-available software?
  - How do you define "good" and "bad"?
- Are we in a post-metaphor era?
  - Should we strive to be?

For Next Week

- 11/2 readings
  - Interfaces (PSR CH6). Visual Design (CRC Ch 17, 19--especially pp507-553)
- 11/9 readings
  - Orchestration, flow & excise (CRC Chs 11 & 12).
  - HCI Research as Problem Solving
- Hot topics presentations

- NO lab reflection due this Friday (11/4)
- I2 due 11/9 @ 6pm