

Objectives

- Usability
- Project organization

Review: Software Tools

- What are the benefits of Maven?
- What are the benefits of issue tracking software?
 - What issue tracking software are we using?

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

May 7, 2019

Sprenkle - CSCI335

3

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 page sucks” and “I can improve this page by changing X,Y, and Z”
- Life-long habits for engineering usable products

May 7, 2019

Sprenkle - CSCI335

4

Don't Make Me Think Discussion

- What were the most important points?
 - What is easy to implement?
 - What is trickier?
- What was most surprising?
- What web site did you think had good usability?
- What web site did you think had poor usability?

May 7, 2019

Sprenkle - CSCI335

5

Analysis

- Consider the “good usability” sites and the “poor usability” sites
 - What do they have in common?
 - What lessons should we take from them?

May 7, 2019

Sprenkle - CSCI335

6

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

May 7, 2019

Sprenkle - CSCI335

7

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

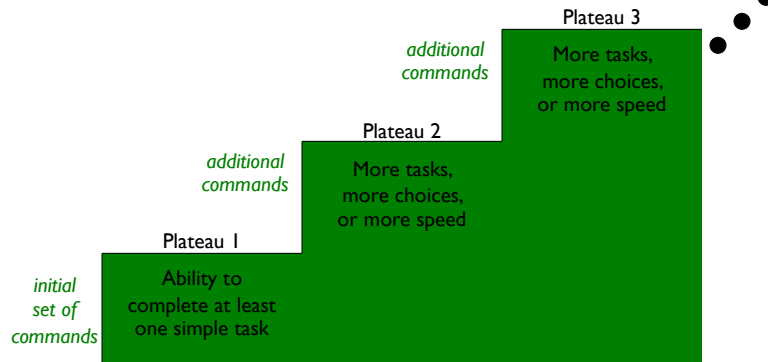
May 7, 2019

Sprenkle - CSCI335

8

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

May 7, 2019

Sprenkle - CSCI335

9

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

May 7, 2019

Sprenkle - CSCI335

10

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

May 7, 2019

Sprenkle - CSCI335

11

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

May 7, 2019

Sprenkle - CSCI335

12

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
- Retention is typically more important with UIs that are **hard to learn**

May 7, 2019

Sprenkle - CSCI335

13

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 !**

May 7, 2019

Sprenkle - CSCI335

14

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

Analysis of Usability from DMMT

- “Trunk test” questions
 - 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?
- For your sites (good and poor), answer these questions
 - Are these answers related to your thoughts on usability?

May 7, 2019

Sprenkle - CSCI335

17

Web Accessibility

- Goal: Everyone should be able to access all web content
- Intro to Web Accessibility
 - <https://www.w3.org/WAI/fundamentals/accessibility-intro/>



- <https://wave.webaim.org/>

May 7, 2019

Sprenkle - CSCI335

18

CODE ORGANIZATION

May 7, 2019

Sprenkle - CSCI335

19

Understanding Your Project's Organization

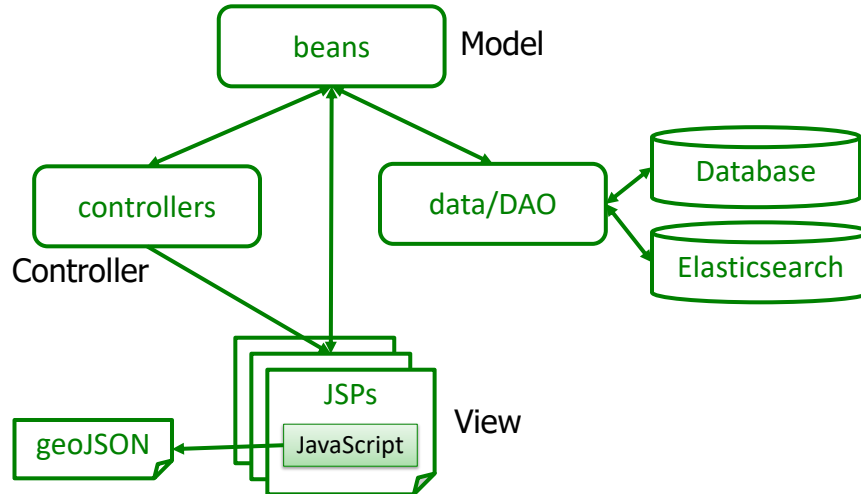
- Draw a picture of the [current] code organization/architecture
 - MVC
 - Components
 - ...
- What is the flow? Who talks to what?
- Where is the source of the various components located?
- Where are the URLs coming from?

May 7, 2019

Sprenkle - CSCI335

20

AGP Overview

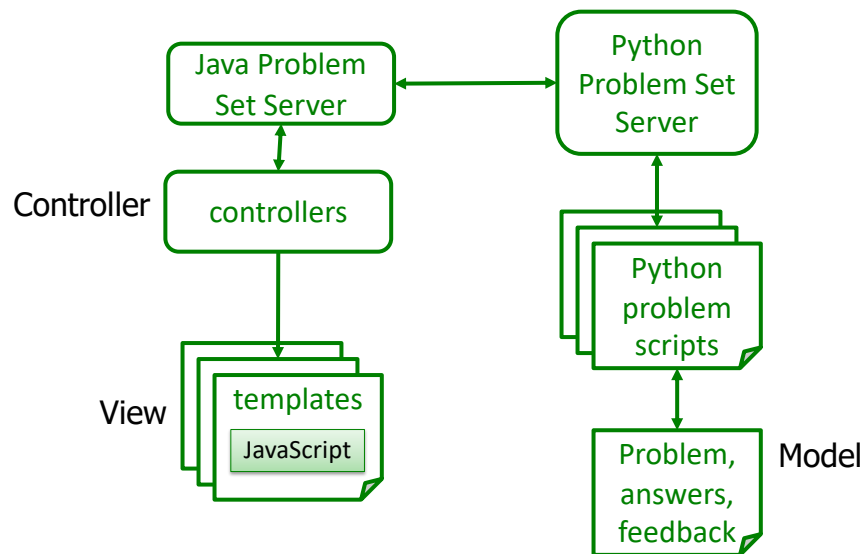


May 7, 2019

Sprenkle - CSCI335

21

ChemTutor Overview



May 7, 2019

Sprenkle - CSCI335

22

TODO

- Understand project code organization
- Develop a plan of attack
- Consult with Prof Sprenkle
 - Help queue
 - Any questions for clients?
- Prepare for Thursday's exam