

Evaluating Technologies

UNC COMP 523

Wed Sep 2, 2020

Prof. Jeff Terrell

Announcements

- intro music: *Any Colour You Like* by Pink Floyd, because (a) you get to pick colors (colours) in your designs and (b) most of you get some say in what technologies you use for your project
- good input on TAQs, thanks
- gradebook is current
- A3: User Stories ☕☕ due next week
- note that the journal should be updated every week from now on
 - include at least decisions from team meetings and mentor meetings
 - how can you remember to do this?
- week after next, A4: Clickable Prototype ☕☕☕☕
 - suggestion: sketch some lo-fi designs on paper (or maybe FreeHand) and show to mentor next week

Plan for today

- announcements
- testing designs (continued from last lecture)
- design tools (continued from last lecture)
- evaluating technologies
- Figma tutorial

Testing designs

(Note: adapted from Graham Langdon's design principles lecture.)

Topics

- Prototyping
- Getting feedback
- Iterating

Prototyping

- Problem: hard to see something with new eyes
- Once a design is no longer new, hard to spot mistakes or rough spots
- If the user interface emerges over weeks or months as development progresses, hard to see any problems with it
- Prototyping helps you quickly dump a vision and examine it while it's still fresh

Getting feedback

- Find people with fresh eyes (even friends & family) and ask for their perspective
- Give them a goal, but let them find their own way.
- (Note: this can be painful to watch! But it's very useful.)
- Ask *open-ended, non-judgmental* questions
 - "Why did you do it that way?"
 - "How did you expect that to happen?"
 - Not, "why didn't you do it this way?"
 - And not, "have you used a computer before?"
- Ask them to think out loud
 - "If you wanted to accomplish X, what would you do first?"
 - if you don't ask real-time, those details might be forgotten
- Invite criticism: what was the most confusing part of that for you?

Iterating

- Don't scrap your design and start over!
- Instead, try tweaking one thing and doing another interview
- Be willing to change anything
- Try the simplest thing that can possibly work

Plan for today

- announcements
- ~~testing designs~~
- design tools (continued from last lecture)
- evaluating technologies
- Figma tutorial

Design Tools

(Note: copied Graham Langdon's content from his slides here. Used with permission.)

- Invision Freehand
- Figma
- Noun Project
- UI Kits
- Note: all of these are free or have good free versions

Freehand

- Whiteboard on steroids
- Infinite canvas, infinite zoom
- Approachable
- Collaborate remotely
- Annotate screenshots

Figma

- Create clickable prototypes
- Components and styles
- Collaborate remotely
- Share prototypes
- Get feedback

Icons

- Material icons: <https://material.io/resources/icons/>
- The Noun Project: <https://thenounproject.com/>

UI Kits

- Figma Freebies <https://www.figmafreebies.com/>
- Figma Material Design Kit <https://www.figma.com/resources/assets/material-design-themeing-ui-kit/>
- Setproduct (paid kits) <https://setproduct.com/>

Jeff's additions

- UNC people have free access to Adobe XD
 - Similar to Figma, but a desktop app rather than web app
 - And AFAIK no real-time collaboration abilities
- Framer: web app intended for relatively standard experiences
- I've also heard recommendations for InVision (like XD)
- Balsamiq makes lo-fi "wireframe" prototypes easy; 30-day free trial

Poll

Do you already know what technologies you'll be using?

Plan for today

- announcements
- ~~testing designs~~
- ~~design tools~~
- **evaluating technologies**
- Figma tutorial

Course context

- This lecture helps with A6: Application Architecture
- That's due Sept. 27
- So this is premature for some
- Suggestion:
 - connect with future you, who needs to know this stuff
 - or, come back and watch the recording of this lecture later
- (I'm trying to keep lectures relevant by staying ahead of assignments.)
- (Assignments provide project-generic structure to keep things moving.)

Evaluating tech: the big idea

- You have to build an app.
- How do you decide which tools to use?
 - programming language
 - framework
 - database or backend
 - platform (application or infrastructure)
 - build tools, IDEs, libraries, etc.

Evaluating tech: outline

- evaluation approaches: survey and analysis (today)
- the virtue of simplicity (next Wednesday)

Diverse priorities

Recall Assignment 0: Project Preferences.

There were many ways to decide on a project.

- clarity of the proposal
- maturity of the idea
- client flexibility, friendliness, or availability
- social impact
- *et cetera*

Diverse priorities

So it is with evaluating a technology

- How new is it?
- How popular is it?
- Is it maintained?
- How easy is it to start with?
- Am I already comfortable with it?
- Will it help me get hired after graduation?
- How easy is it to understand?

Discussing your priorities explicitly with your team might help y'all find consensus.

Evaluation approaches

Let's take a closer look at a few priorities

- Newness
- Popularity
- Is it maintained?
- Is it easy?

Newness

Pros

- It can expand your mind
- It can be fun
- It feels cool to be using cutting edge things; you're a pioneer
- We're into computers and programming, so we must have some tolerance if not enjoyment of new things

Cons

- There may be bugs
- Documentation may be lacking
- You may be relatively alone using it, so may be hard to get help

Popularity

Pros

- If others are successfully using this, that implies that you can too
- It can be easy to get help, even for uncommon uses
- Following the crowd tends to be easy

Cons

- Popularity is a cruel and whimsical master
- Many excellent engineers are poor marketers, and their excellent work isn't popular
- Many excellent marketers are poor engineers, and their mediocre work is popular

Is it maintained?

Proxy measure: when was the most recent commit to the repository?

Pros

- If you discover and report a bug, it may get fixed quickly
- You may get new features or better performance “for free”

Cons

- Maybe bug fixes are still happening because it is buggy software, and there are plenty more bugs not yet found
- You might be coupling your project to one that is constantly moving, and you might also have to stay moving to keep up. (I.e. “free” upgrades usually have an associated cost.)

Ease

Word origin: to lie near; adjacent \

Pros

- It's an easy lift—for you—to use it
- No or little learning curve required
- Start being productive fast

Cons

- Maybe others on your team have a different sense of what's easy—**easy is *relative***
- Maybe a different tool would be worth the learning curve in the end
- Easy to start does not imply easy to finish

**Any priority you have has tradeoffs.
There are tradeoffs everywhere!**

Plan for today

- announcements
- ~~testing designs~~
- ~~design tools~~
- ~~evaluating technologies~~
- **Figma tutorial**

Figma tutorial

Follow along with me

Also, writeup available here:

<https://comp523.cs.unc.edu/f19/lectures/figma-tutorial.pdf>