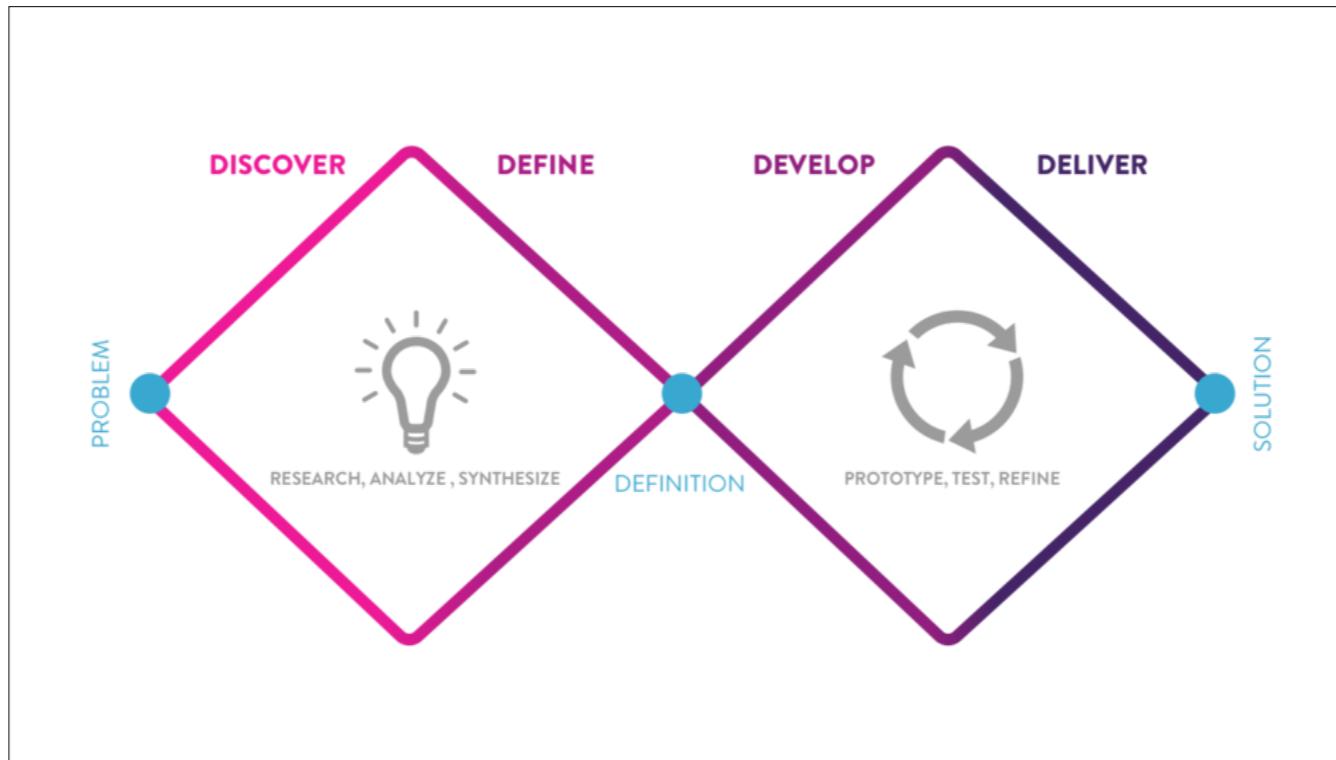




CSMA 113 - Mixed Reality Studio

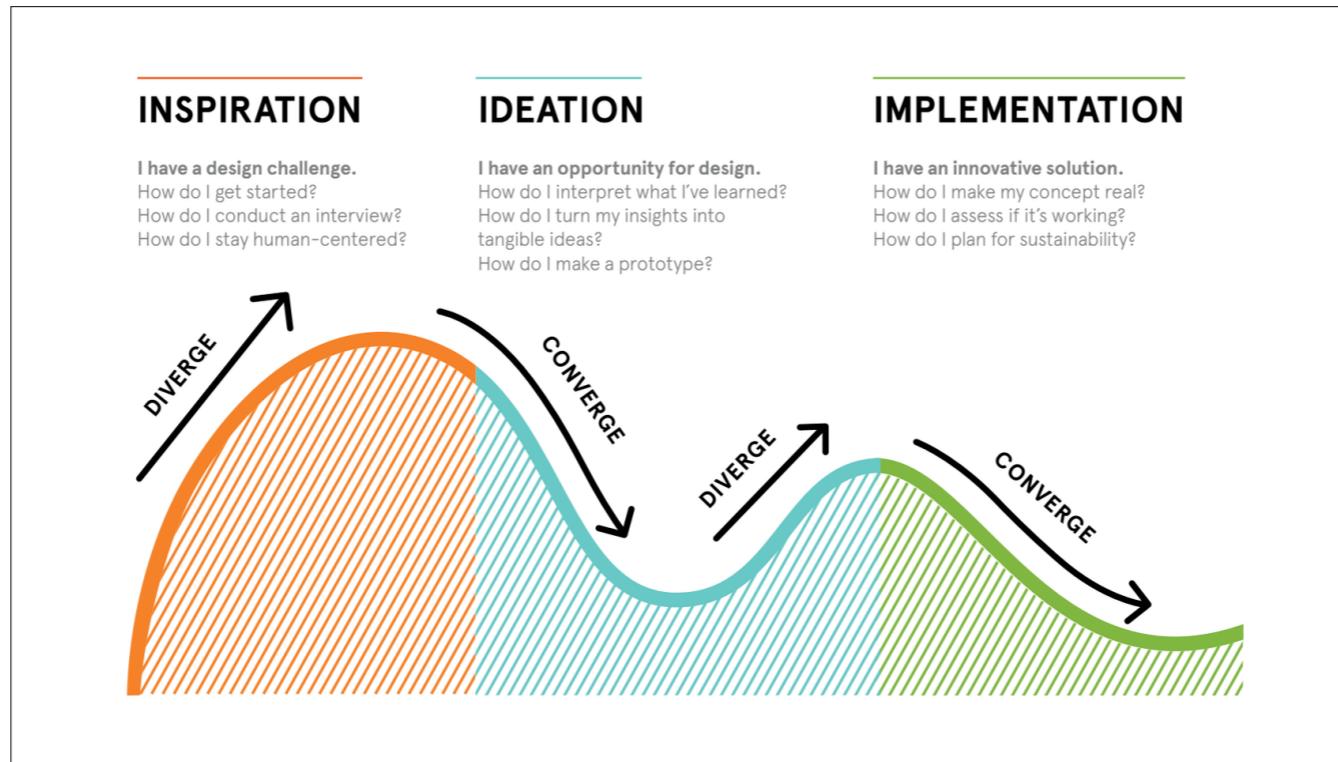


Double Diamond is the name of a design process model developed by the British Design Council in 2005

Alessi, BSkyB, BT, LEGO, Microsoft, Sony, Starbucks, Virgin Atlantic Airways, Whirlpool, Xerox, Yahoo!

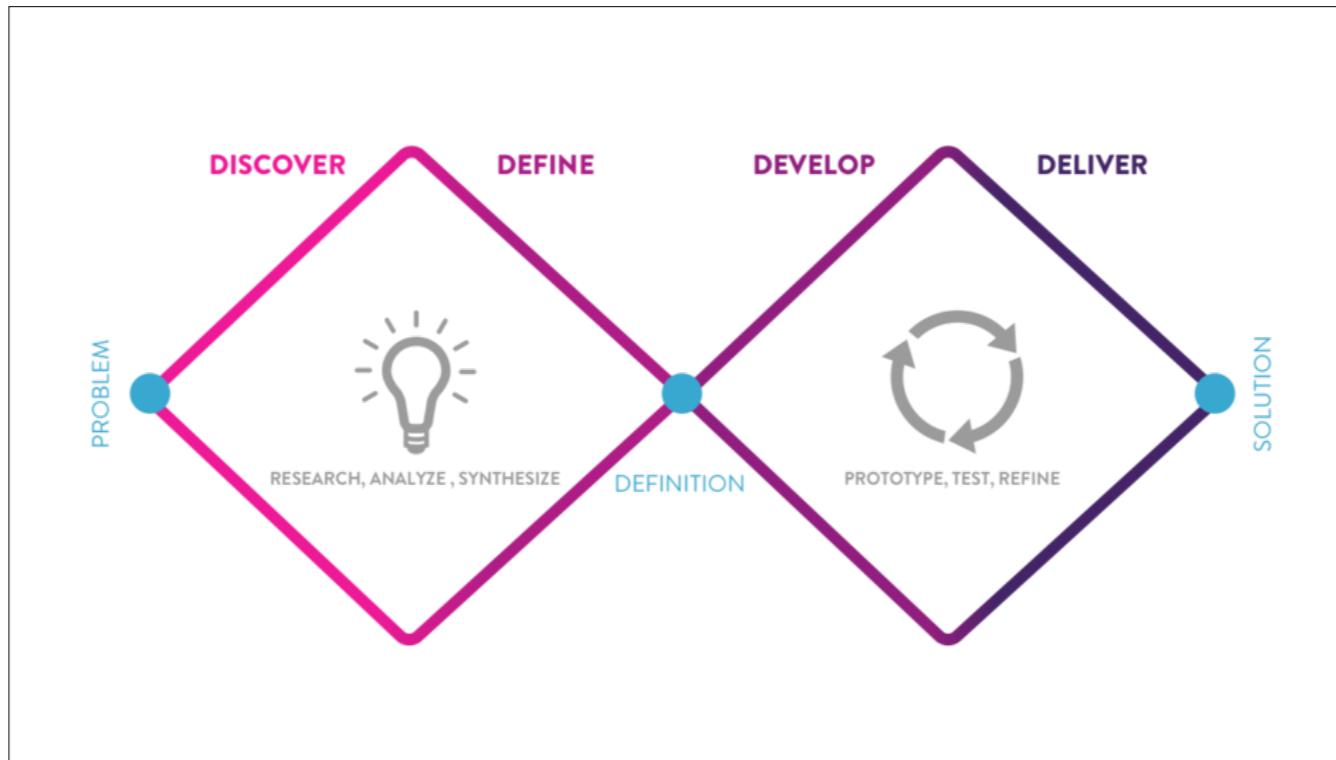
Different designers manage the process of design in different ways. But when we studied the design process in eleven leading companies, we found striking similarities and shared approaches among the designers we talked to.

Very useful to structure development of ideas when otherwise so open ended



Ideo Human Centered Design

Phases of this process are either diverging or converging. During a diverging phase, you try to open up as much as possible without limiting yourself, whereas a converging phase focuses on condensing and narrowing your findings or ideas.



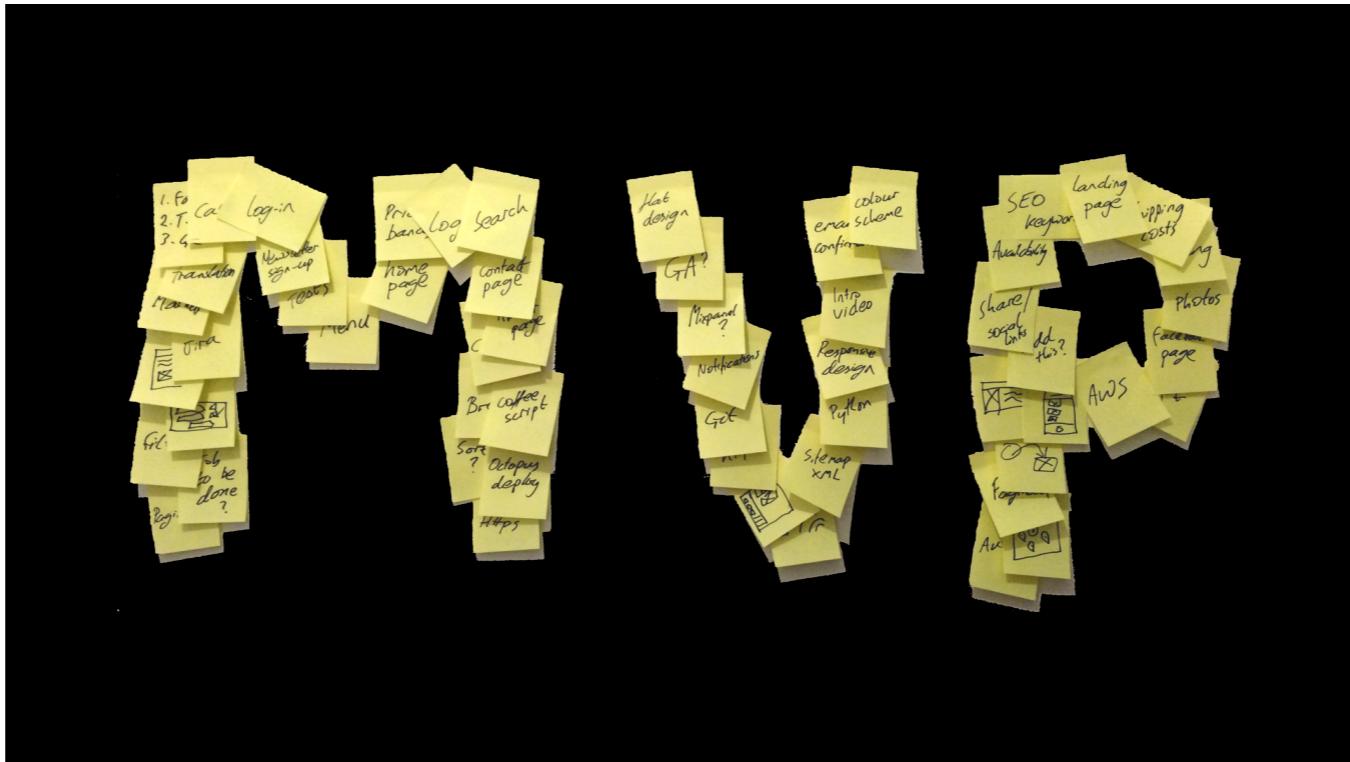
Not equal time....equal weight

Discover – The first quarter of the Double Diamond model covers the **start of the project**. Designers try to **look at the world in a fresh way, notice new things and gather insights**.

Define – The second quarter represents the definition stage, in which designers try to **make sense of all the possibilities identified in the Discover phase**. Which matters most? **Which should we act on first? What is feasible?** The goal here is to develop a clear creative brief that frames the fundamental design challenge.

Develop – The third quarter marks a period of development where **solutions or concepts are created, prototyped, tested and iterated**. This process of trial and error helps designers to improve and refine their ideas.

Delivery – The final quarter of the double diamond model is the delivery stage, where the **resulting project** (a product, service or environment, for example) is **finalized, produced and launched**.



Minimum Viable Product

The Lean Startup - by Eric Ries

Maximum amount of validated learning about customers with least effort.

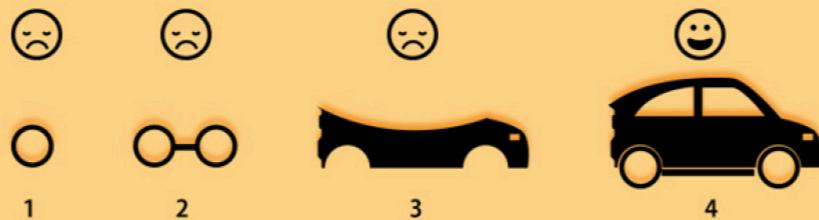
(Grown to mean a lot more than it was originally)

Riskiest Assumption Test - There is no need to build more than what's required to test your largest unknown.

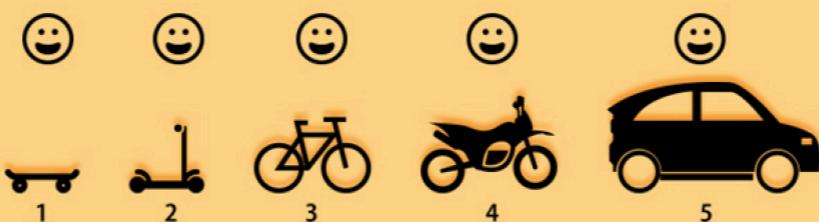
No expectation of perfect code or design. No danger it will prematurely become a product.

HOW TO BUILD A MINIMUM VIABLE PRODUCT

NOT LIKE THIS



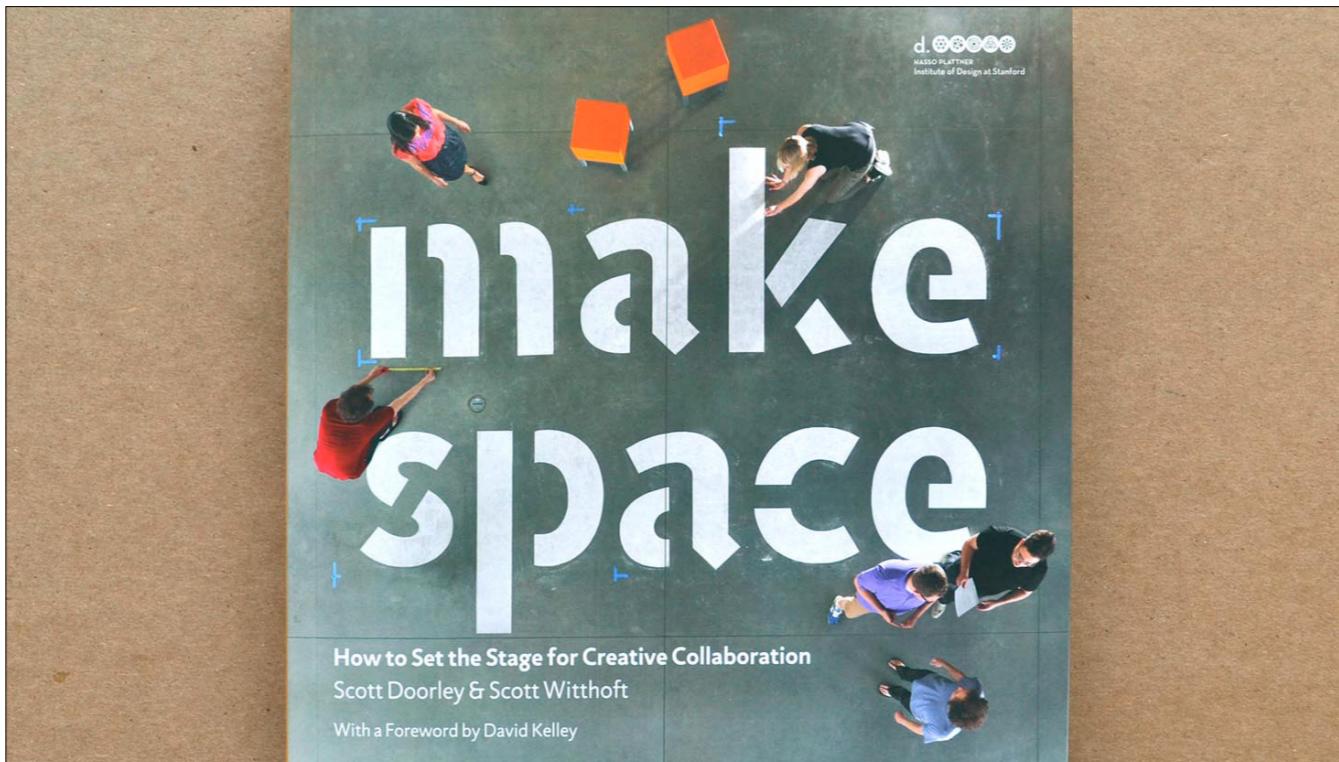
LIKE THIS



Very quickly get to something usable that gets you “from point A to point B”



Don't try to do too much. Focus on designing a focused user experience.



Implementation horrors

Recognize Your Emotional Arc during a Project.

This is a note just for you: creative projects have an emotional frequency. Being aware of this can help you successfully navigate the tricky peaks and troughs.

The arc of a space design process rivals that of any fictional drama. Hopes and dreams hang in the balance. Fear and excitement battle for the upper hand. Egos are invested, and status jockeying abounds. It can be largely frustrating or quite exciting, depending on how you choose to experience it.

As you progress, you will likely encounter the following phenomena. If you know how to recognize them, you can at least reconcile how you are doing with how the project is doing.

+ **A sense of excitement and limitless possibility.** An idea or discovery arrives shiny and new, without the eventual scuff marks inherent in the process of bringing it to fruition. The sense of potential is empowering, yet rarely realistic. It's like buying a Ferrari without yet having found a mechanic to service it. Enjoy this part as it happens, but not so much that you are afraid to let it go.

- **Overwhelming complexity.** As soon as you dig into a space, you'll

of emotional and logistical factors simultaneously at play. This is quicksand. Too much thinking here means trouble. When you are in this territory, focus on doing. Get right to prototyping through quick mock-ups and experiences. Acknowledge and categorize new issues as they arise, but prioritize—you'll never be able to resolve all of them. Keep your eyes open for inspiration and direction away from the soup of complexity.

+ **Unifying insights.** These are moments of clarity when you feel you've got it all figured out. They are the siren songs: glorious but potentially derailing. Strive with all your might to get to this point, but diligently question this clarity when you arrive.

- **Complete loss of confidence.** What was I thinking? I can't do this. You can, and in fact, you are doing it. This feeling is mostly useless, although it may signal that you should enlist some help to tackle a specific issue. If so, get the

experience, ignore it by taking new action. - **The brutal realities of implementation.** Almost everything you do will take longer than you think because there is a lot more to consider than is apparent. You will have to make compromises. Have the strength to stand up for the right things and the wisdom to let go of the meaningless bits. The ability to know how and when to do this often comes from the proof & intuition you'll acquire through prototyping.

+ **Completion.** It's done! Enjoy the sense of accomplishment. You may also experience some postpartum malaise. Fine—that's natural, but don't wallow in it. Do take time to celebrate your accomplishment and reflect on your process. You're never really done anyway; you will likely repeat at least some part of this process someday. Reflection is important to make your next steps more efficient.

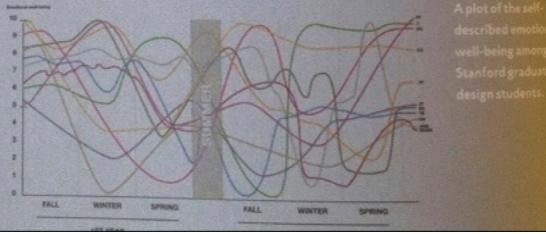
Walls vs. Open Space

Render: Walls support collaboration, yet collaborative spaces thrive in openness.

Openness is an engine of innovation. Openness in the form of visibility and room to move breeds awareness and spawns intersections that keep people inspired and keep projects in tune.

Walls are terrific display surfaces. Walls and vertical surfaces are great for making ideas visible across a group. Visual evidence of work inspires and supports discovery and synthesis.

Dann,



A sense of excitement and limitless possibility

A sense of excitement and limitless possibility

An idea or discovery arrives shiny and new, without the eventual scuff marks inherent in the process of bringing it to fruition. The sense of potential is empowering, yet rarely realistic. It's like buying a Ferrari without yet having found a mechanic to service it. Enjoy this part as it happens, but not so much that you are afraid to let it go.

Overwhelming complexity

As soon as you dig into a [project], you'll uncover a seemingly endless pile of emotional and logistical factors simultaneously at play. This is quicksand. Too much thinking here means trouble. When you are in this territory, focus on doing. Get right to prototyping through quick mock-ups and experiences. Acknowledge and categorise new issues as they arise, but prioritise – you'll never be able to resolve all of them. Keep your eyes open for inspiration and direction away from the soup of complexity.

Unifying insights

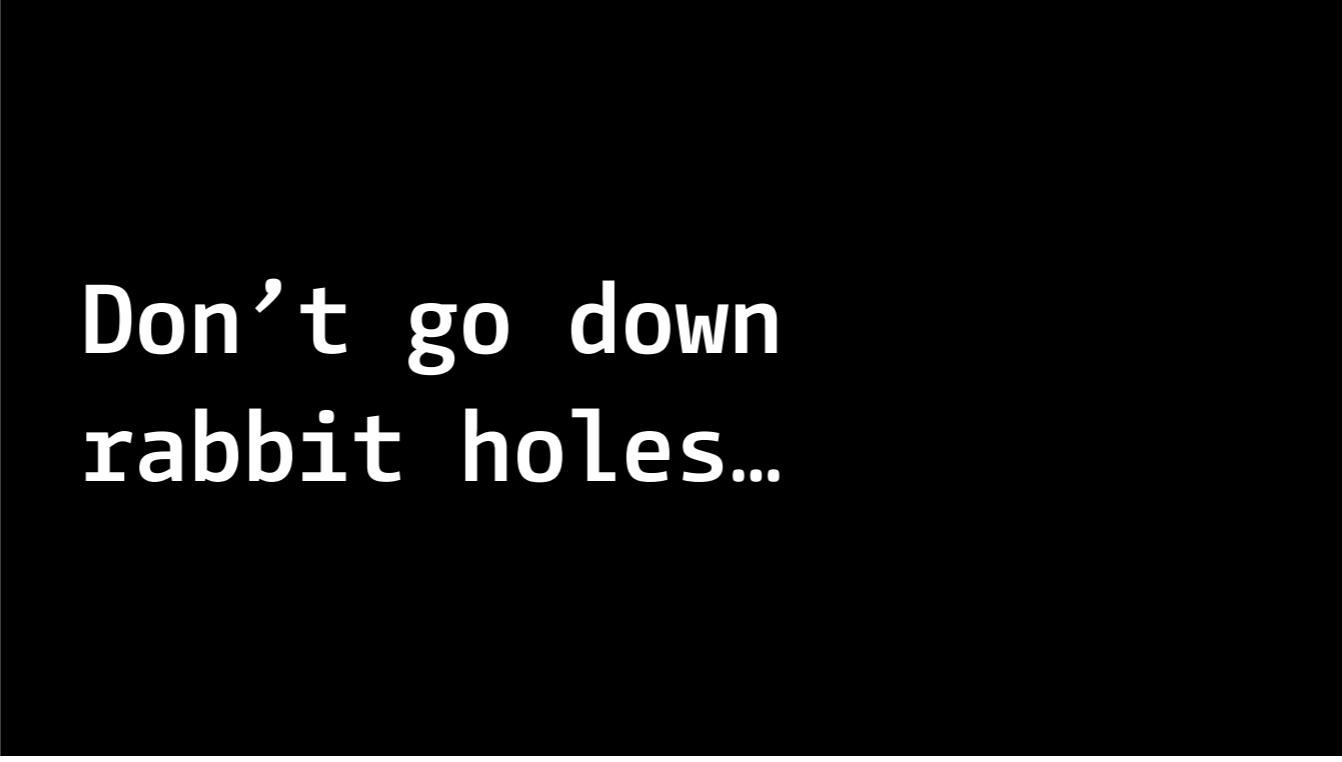
These are moments of clarity when you feel you've got it all figured out. They are the siren songs: glorious but potentially derailing. Strive with all your might to get to this point, but diligently question this clarity when you arrive.

The brutal realities of implementation

Almost everything you do will take longer than you think because there is a lot more to consider than is apparent. You will have to make compromises. Have the strength to stand up for the right things and the wisdom to let go of the meaningless bits. The ability to know how and when to do this often comes from the proof and intuition you'll acquire through prototyping.

Use What You Know

- Break problem into smaller pieces
- ‘Hack-y’ solutions are OK!



Don't go down
rabbit holes...

A beautiful 3D Model is great, but not if your project doesn't do anything.

Don't forget the boring stuff

- Organizing your program helps you make changes
- Keep track of the **state** of your program
 - For example:

```
bool isReady = false;
```
- Use if/else statements to check the state and change behavior accordingly:

```
if (isReady == true) {  
    // Do something here  
} else {  
    // Do something different here  
}
```

Tuesday 2/11: Midterm Proposals

- Next week: have an idea for your midterm
- Ask questions about if something is possible

Presentation
(e.g. powerpoint)

- Convey Your Idea
- Show examples of AT LEAST one similar project
- Think about a USER EXPERIENCE
- Demonstrate proficiency in both the visual and scripting sides of Unity
- Can be understood and used by somebody NOT taking this class

- Area interaction (something happens when two colliders or triggers touch)
- Trigger interaction (something happens when the user presses a button on the controller)
- Demonstrates all versions of movement we have discussed in class.
- AT LEAST 50% original assets

“Fake” Designs are OK
(e.g. a podcast app)

Due 3/6

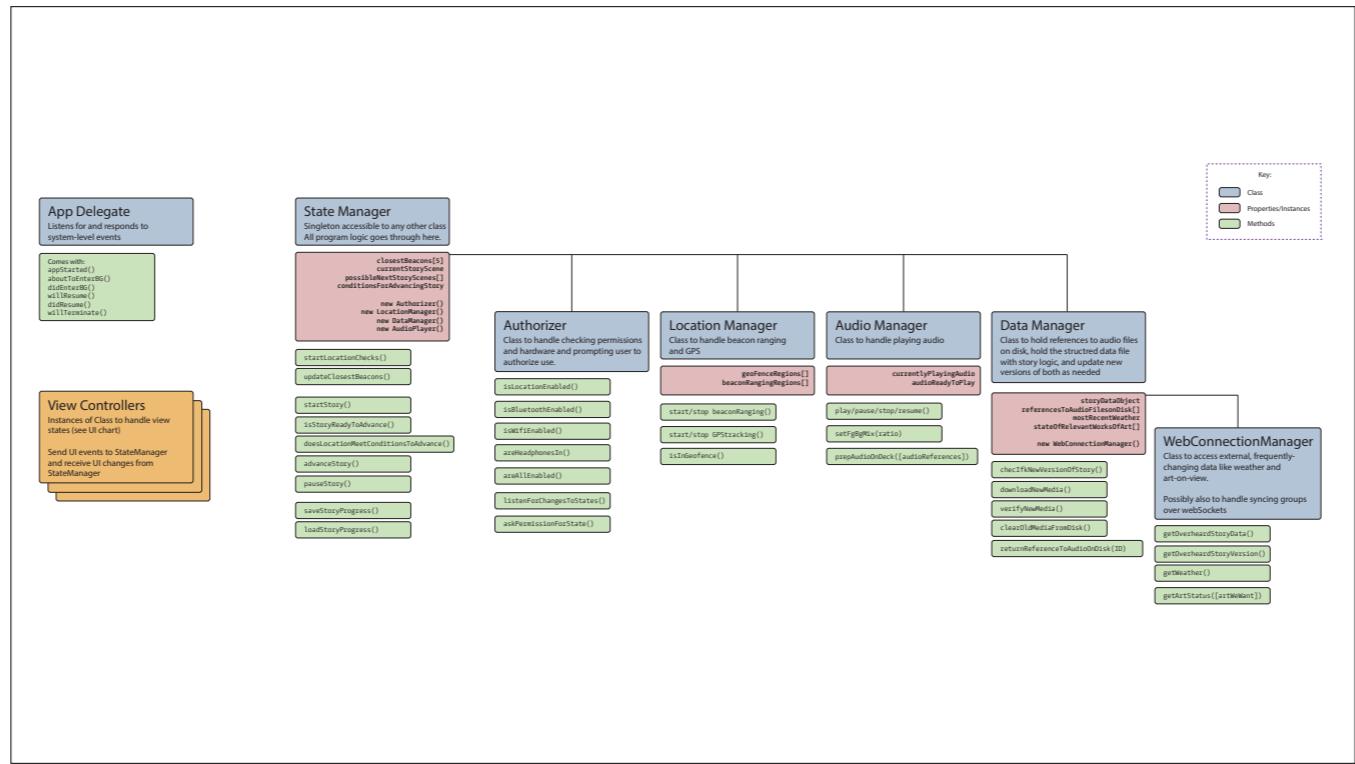
- Unity Projects
- Documentation Journals



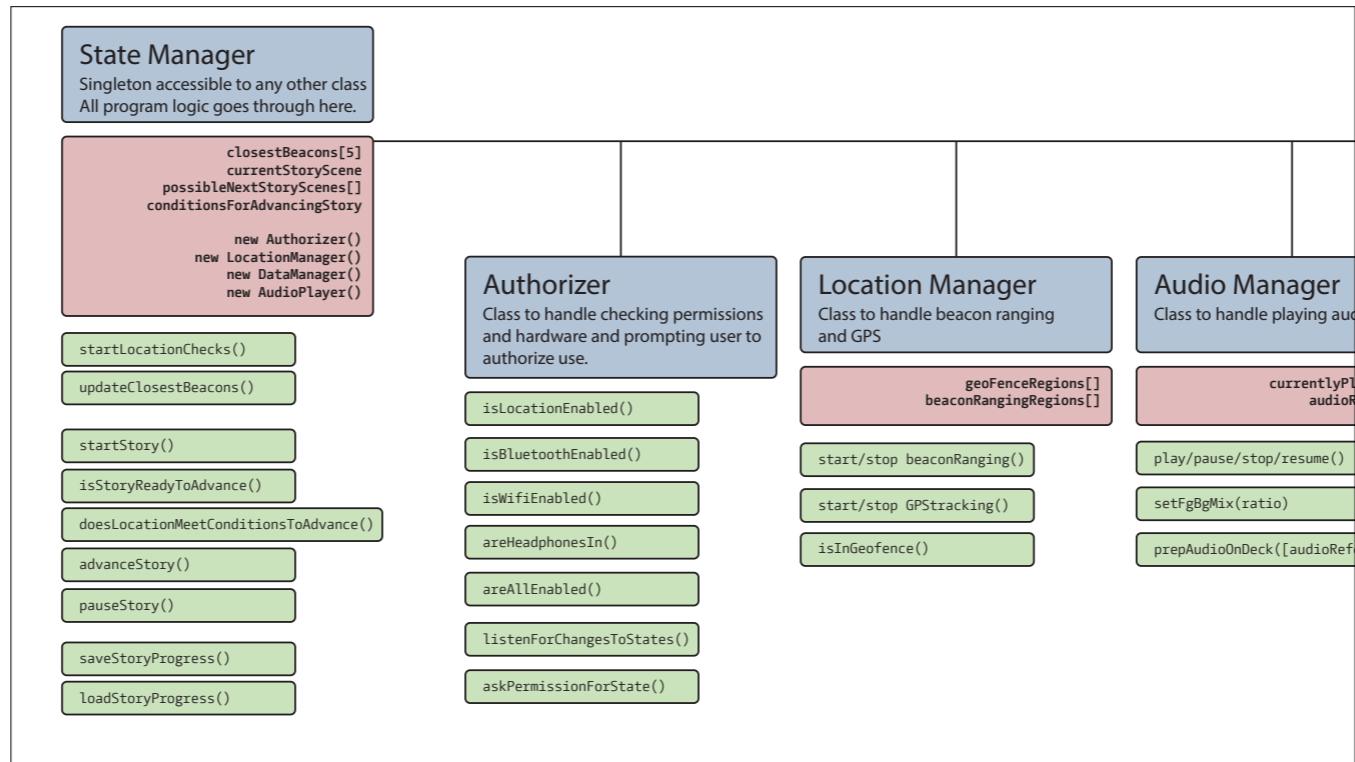
Documentation Journals?

**Three Entries between
now and due date.**

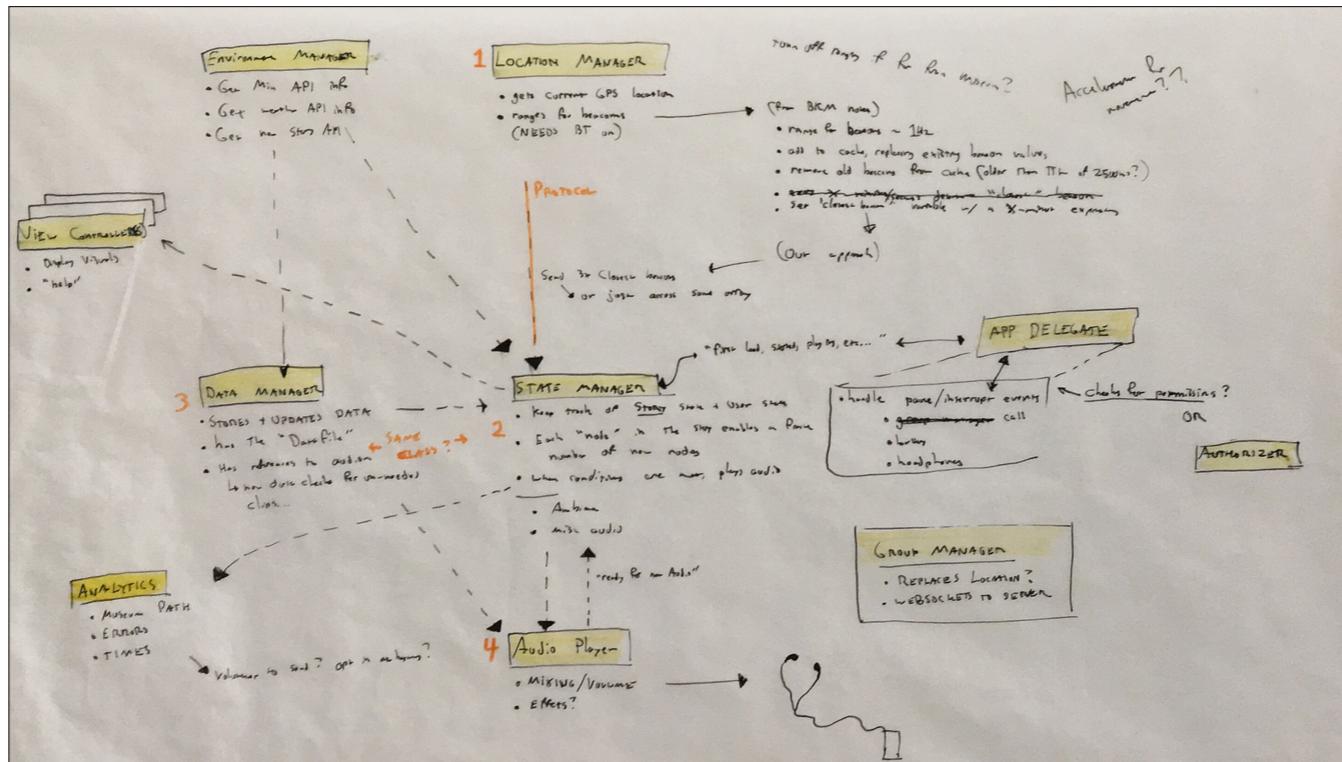
I wanted to do X, but when my friend tried it she did Y, which broke everything, so I had to go back and...



Example of a way to organize an app with lots of components



Even for simple programs, helps to organize into well-defined pieces (classes) and the variables & functions that go along with each piece.



The same chart, before it was cleaned up and finalized



CSMA 113 - Mixed Reality Studio