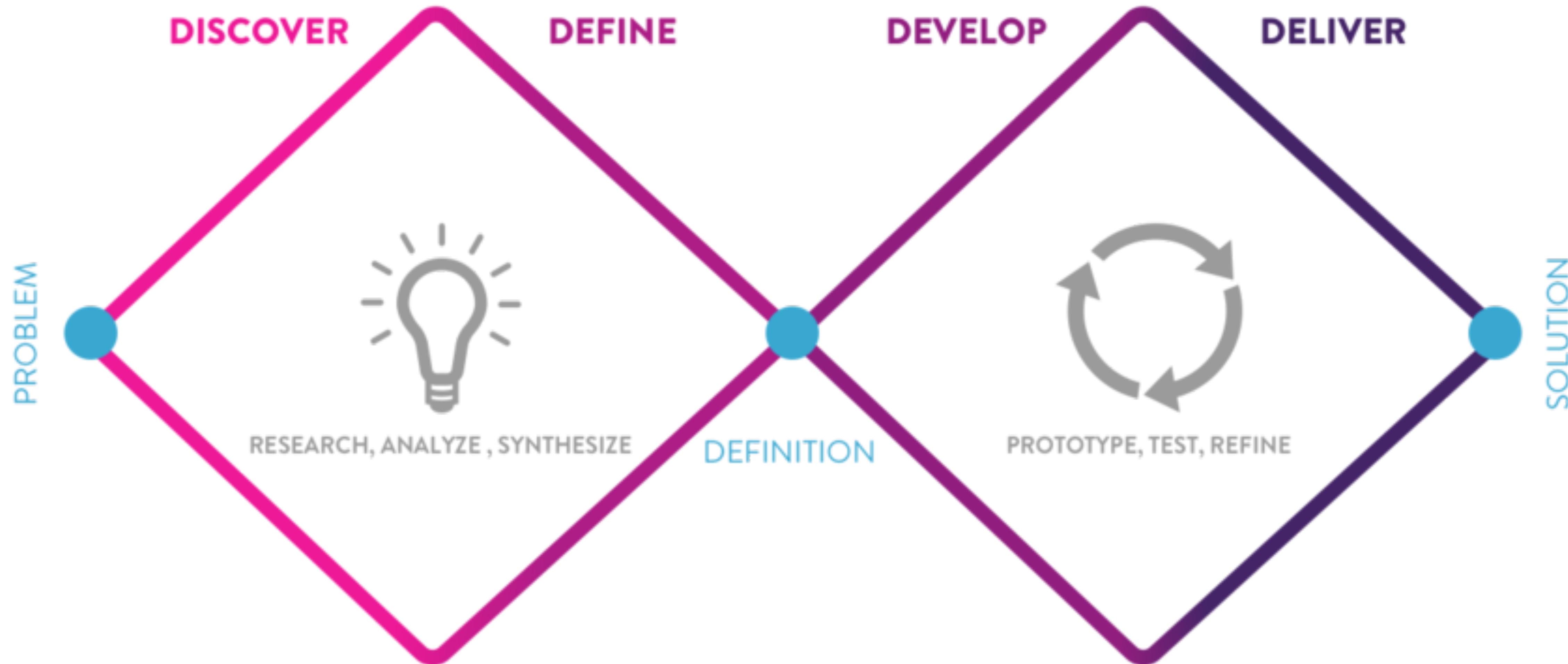




CSMA 113 - Mixed Reality Studio



INSPIRATION

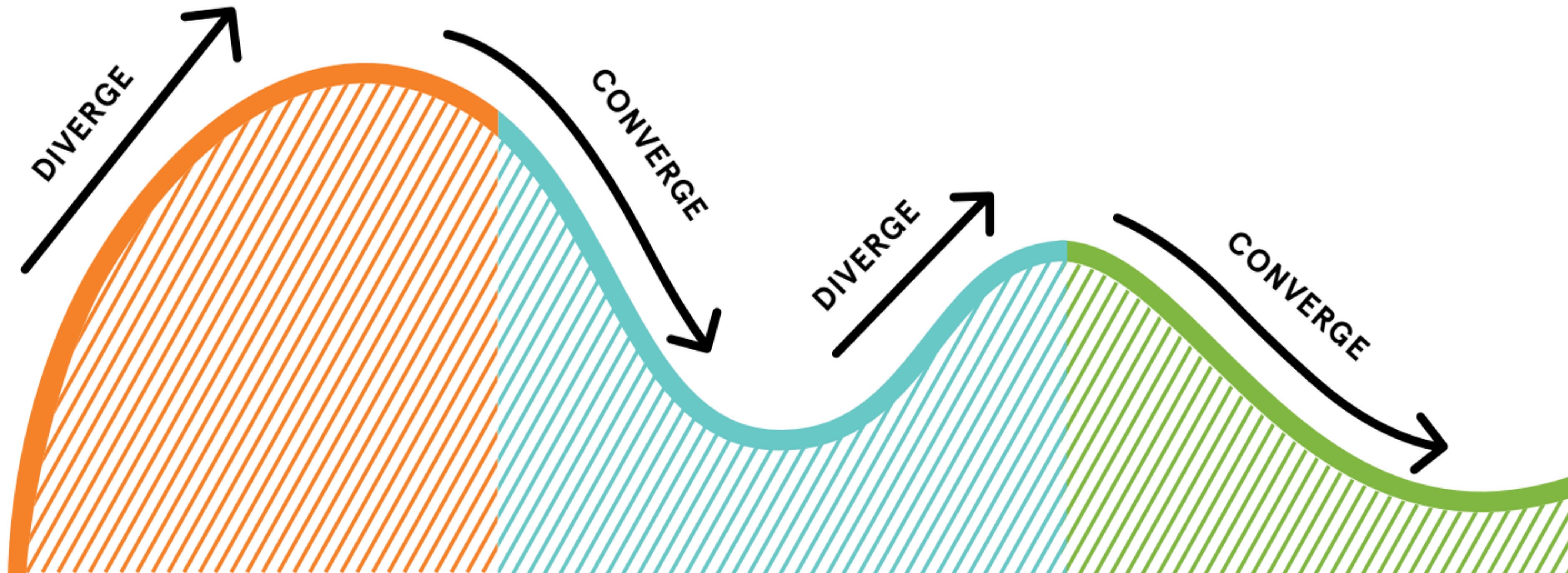
I have a design challenge.
How do I get started?
How do I conduct an interview?
How do I stay human-centered?

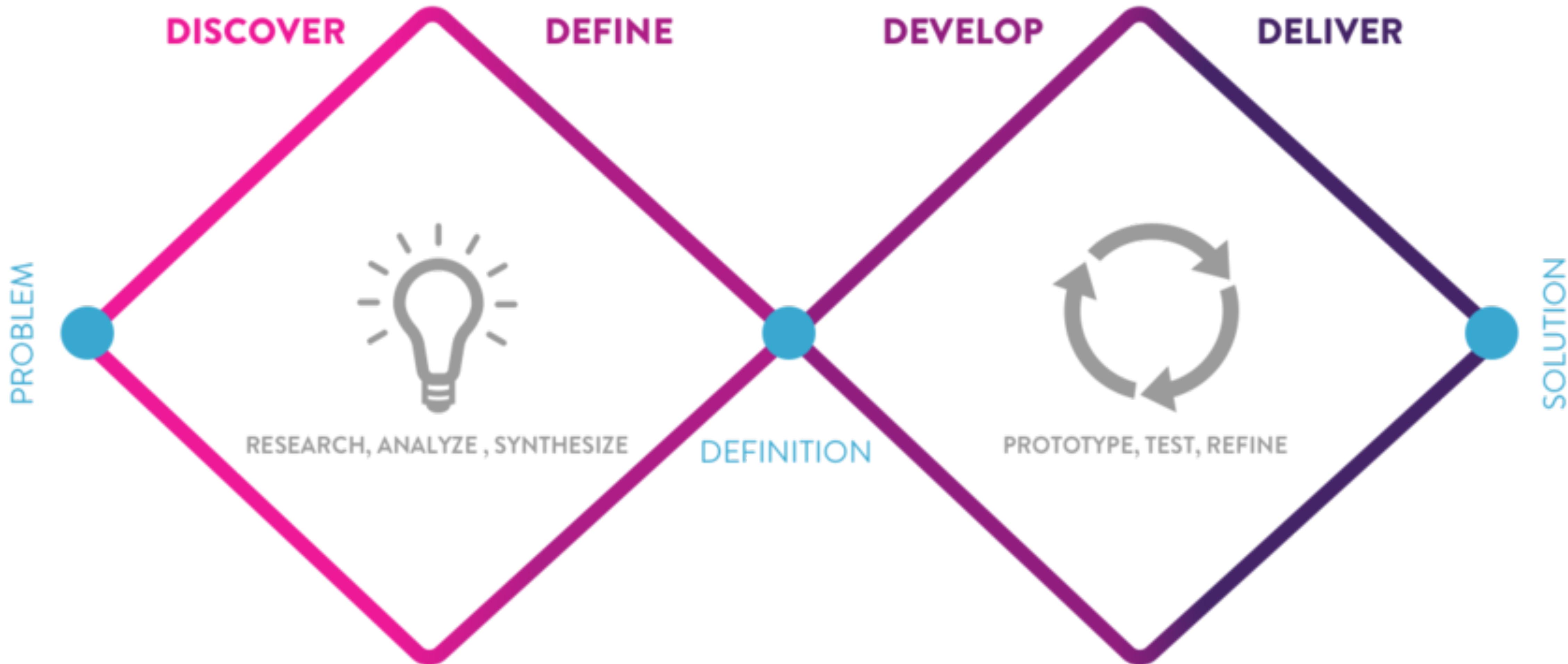
IDEATION

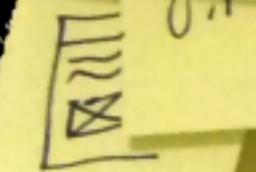
I have an opportunity for design.
How do I interpret what I've learned?
How do I turn my insights into tangible ideas?
How do I make a prototype?

IMPLEMENTATION

I have an innovative solution.
How do I make my concept real?
How do I assess if it's working?
How do I plan for sustainability?



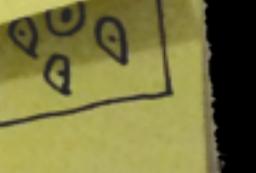


1. Fo
2. T. Ca' log-in
3. G-
translation
Manager
Tira

file
to be done?
Page

Price
banana, log search
home page
Newsletter sign-up
tests
menu
" " page
Brew coffee script
Sort? Octopus deploy
HTTPS

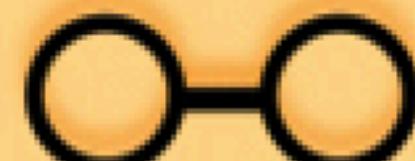
flat design
GA?
Mixpanel?
Notifications
Got
Sitemap XML


colour
emarscheme
confirms
SEO
keyword
landing page
vipping costs
Availability
share/
social links
dd this?
AWS

fragment
Au 

HOW TO BUILD A MINIMUM VIABLE PRODUCT

NOT LIKE THIS



1

2

3

4

LIKE THIS



1

2

3

4

5





make space

How to Set the Stage for Creative Collaboration

Scott Doorley & Scott Witthoft

With a Foreword by David Kelley

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

discover a mix 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.

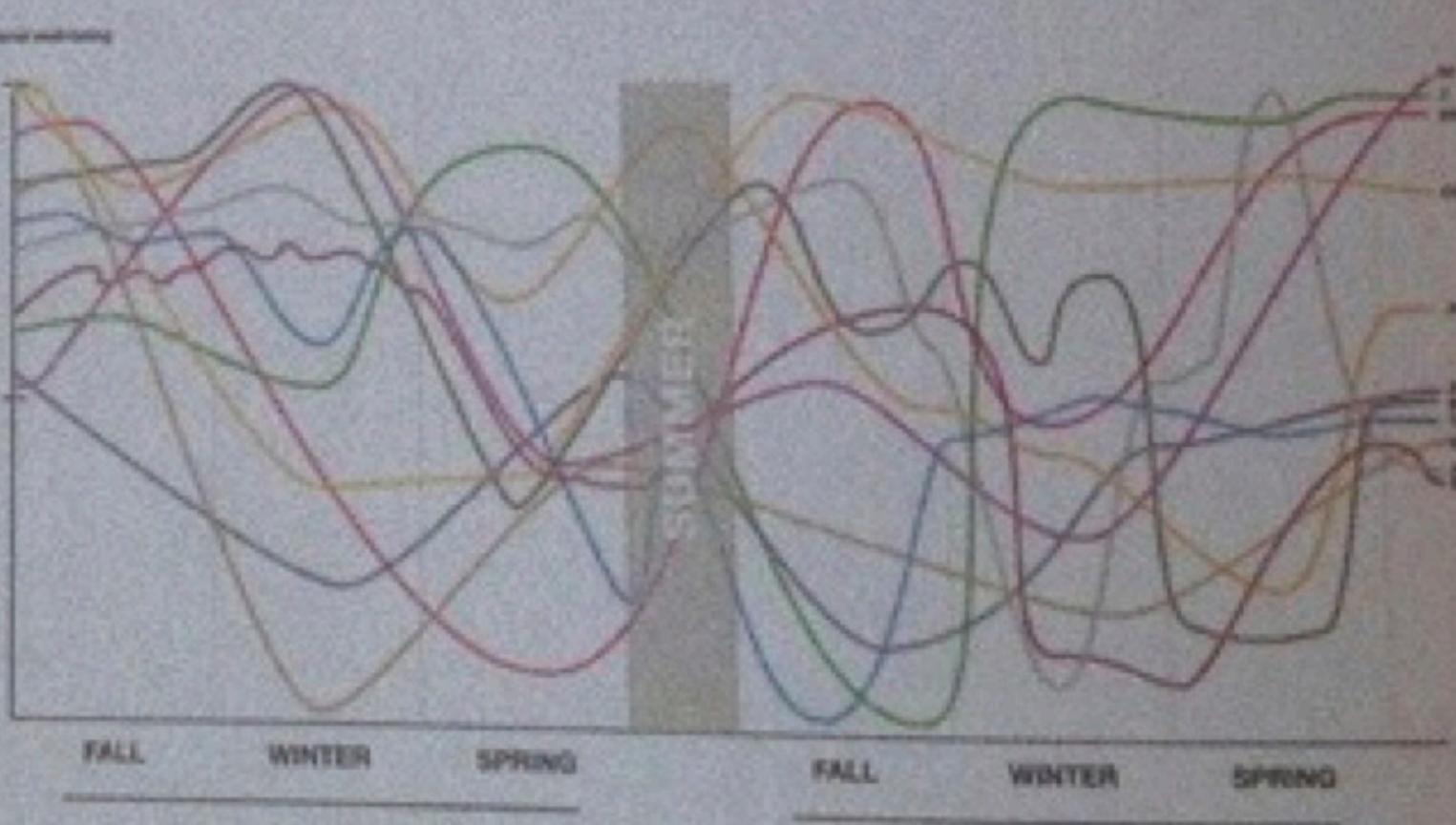
+ **Unifying insights.** These are moments of clarity when you feel you've got it all figured out. They are the siren songs: glorious but potentially derailing. Strive with all your might to get to this point, but diligently question this clarity when you arrive.

- **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.



A plot of the self-described emotional well-being among Stanford graduate design students.

Walls VS. Open Space

Pro: 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.

Damn.
This is like

A sense of excitement
and limitless
possibility

Overwhelming
complexity

Unifying
insights

The brutal realities
of implementation

Use What You Know

Don't go down
rabbit holes...

Don't forget the
boring stuff

Tuesday 2/11:
Midterm
Proposals

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...

App Delegate
Listens for and responds to system-level events

Comes with:
appStarted()
aboutToEnterBG()
didEnterBG()
willResume()
didResume()
willTerminate()

View Controllers
Instances of Class to handle view states (see UI chart)

Send UI events to StateManager and receive UI changes from StateManager

State Manager
Singleton accessible to any other class
All program logic goes through here.

closestBeacons[5]
currentStoryScene
possibleNextStoryScenes[]
conditionsForAdvancingStory
new Authorizer()
new LocationManager()
new DataManager()
new AudioPlayer()

startLocationChecks()
updateClosestBeacons()

startStory()
isStoryReadyToAdvance()
doesLocationMeetConditionsToAdvance()
advanceStory()
pauseStory()
saveStoryProgress()
loadStoryProgress()

Authorizer
Class to handle checking permissions and hardware and prompting user to authorize use.

isLocationEnabled()
isBluetoothEnabled()
isWifiEnabled()
areHeadphonesIn()
areAllEnabled()
listenForChangesToStates()
askPermissionForState()

Location Manager
Class to handle beacon ranging and GPS

geoFenceRegions[]
beaconRangingRegions[]

start/stop beaconRanging()
start/stop GPStracking()
isInGeofence()

Audio Manager
Class to handle playing audio

currentlyPlayingAudio
audioReadyToPlay

play/pause/stop/resume()
setFgBgMix(ratio)
prepAudioOnDeck([audioReferences])

Data Manager
Class to hold references to audio files on disk, hold the structured data file with story logic, and update new versions of both as needed

storyDataObject
referencesToAudioFilesonDisk[]
mostRecentWeather
stateOfRelevantWorksOfArt[]
new WebConnectionManager()

checkIfNewVersionOfStory()
downloadNewMedia()
verifyNewMedia()
clearOldMediaFromDisk()
returnReferenceToAudioOnDisk(ID)

WebConnectionManager
Class to access external, frequently-changing data like weather and art-on-view.

Possibly also to handle syncing groups over webSockets

getOverheardStoryData()
getOverheardStoryVersion()
getWeather()
getArtStatus([artWeWant])

Key:
Class
Properties/Instances
Methods

State Manager

Singleton accessible to any other class
All program logic goes through here.

```
closestBeacons[5]
currentStoryScene
possibleNextStoryScenes[]
conditionsForAdvancingStory

new Authorizer()
new LocationManager()
new DataManager()
new AudioPlayer()
```

```
startLocationChecks()
```

```
updateClosestBeacons()
```

```
startStory()
```

```
isStoryReadyToAdvance()
```

```
doesLocationMeetConditionsToAdvance()
```

```
advanceStory()
```

```
pauseStory()
```

```
saveStoryProgress()
```

```
loadStoryProgress()
```

Authorizer

Class to handle checking permissions and hardware and prompting user to authorize use.

```
isLocationEnabled()
```

```
isBluetoothEnabled()
```

```
isWifiEnabled()
```

```
areHeadphonesIn()
```

```
areAllEnabled()
```

```
listenForChangesToStates()
```

```
askPermissionForState()
```

Location Manager

Class to handle beacon ranging and GPS

```
geoFenceRegions[]
beaconRangingRegions[]
```

```
start/stop beaconRanging()
```

```
start/stop GPStracking()
```

```
isInGeofence()
```

Audio Manager

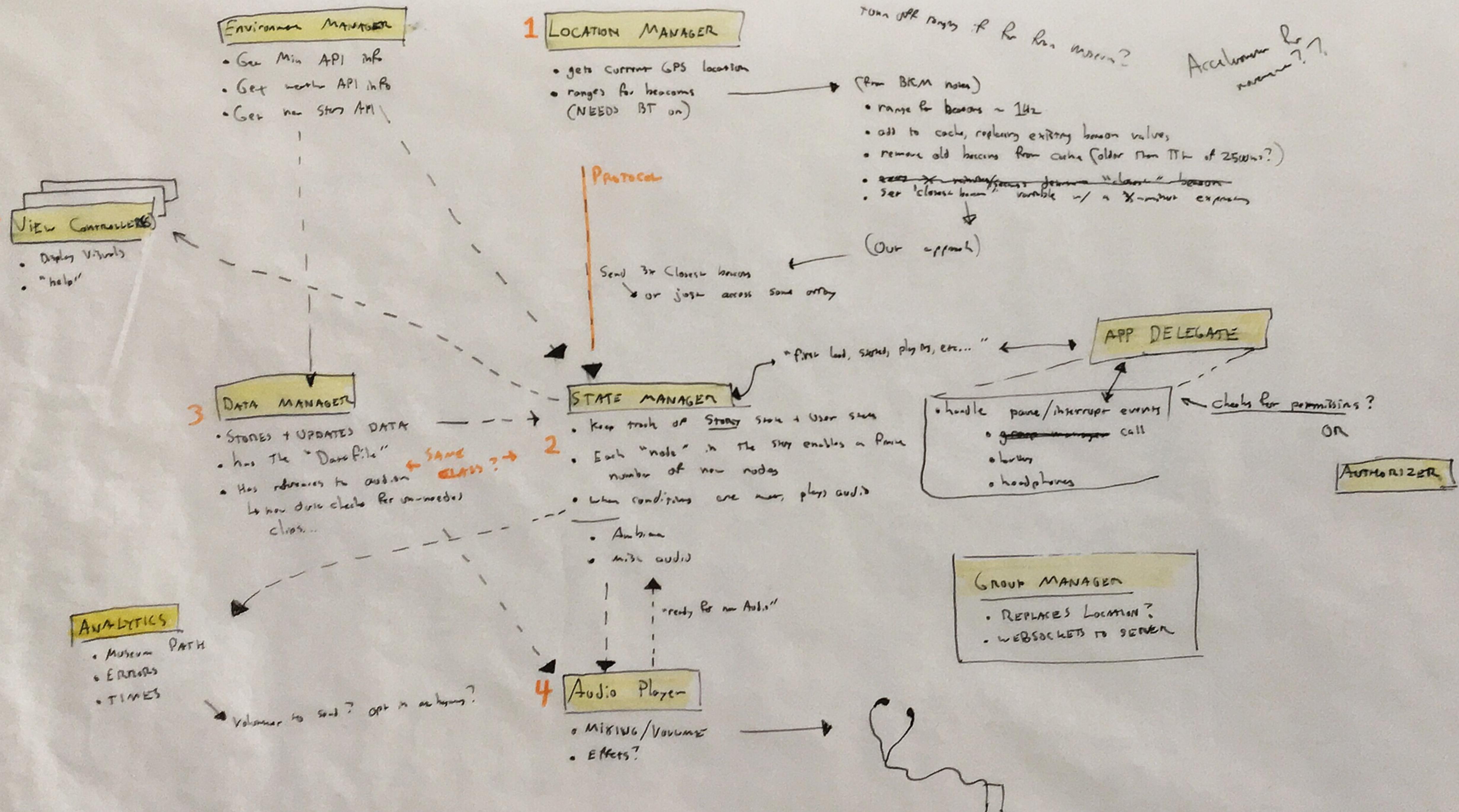
Class to handle playing audio

```
currentlyPl
audioR
```

```
play/pause/stop/resume()
```

```
setFgBgMix(ratio)
```

```
prepAudioOnDeck([audioRef]
```





CSMA 113 - Mixed Reality Studio