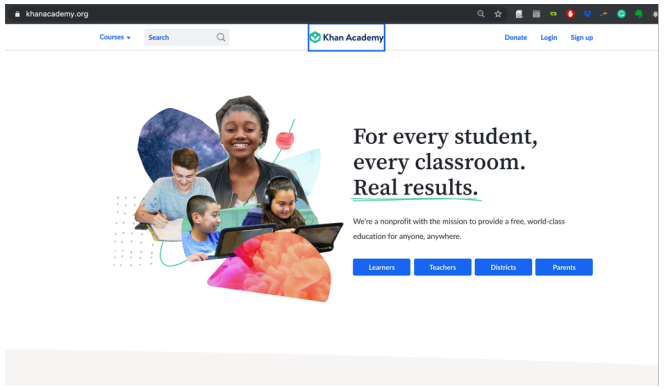
Technical Prototypes

Prof. Lydia Chilton
COMS 4170
14 April 2025

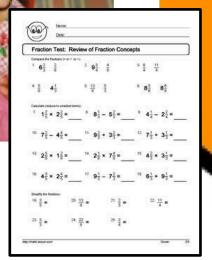
The biggest misconception about creativity

Idea



Product

Creativity is a Process



Person with a challenge

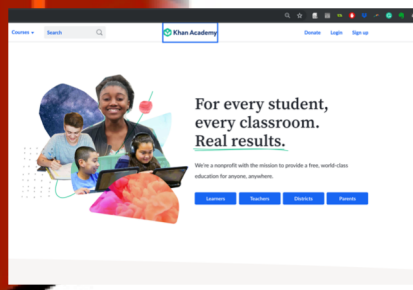
- Brainstorm
- Interview
- Research

Synthesize into insights and ideas

Ideas

- Prototype solutions
- Explore technical options

Test them



Product

Understand the problem

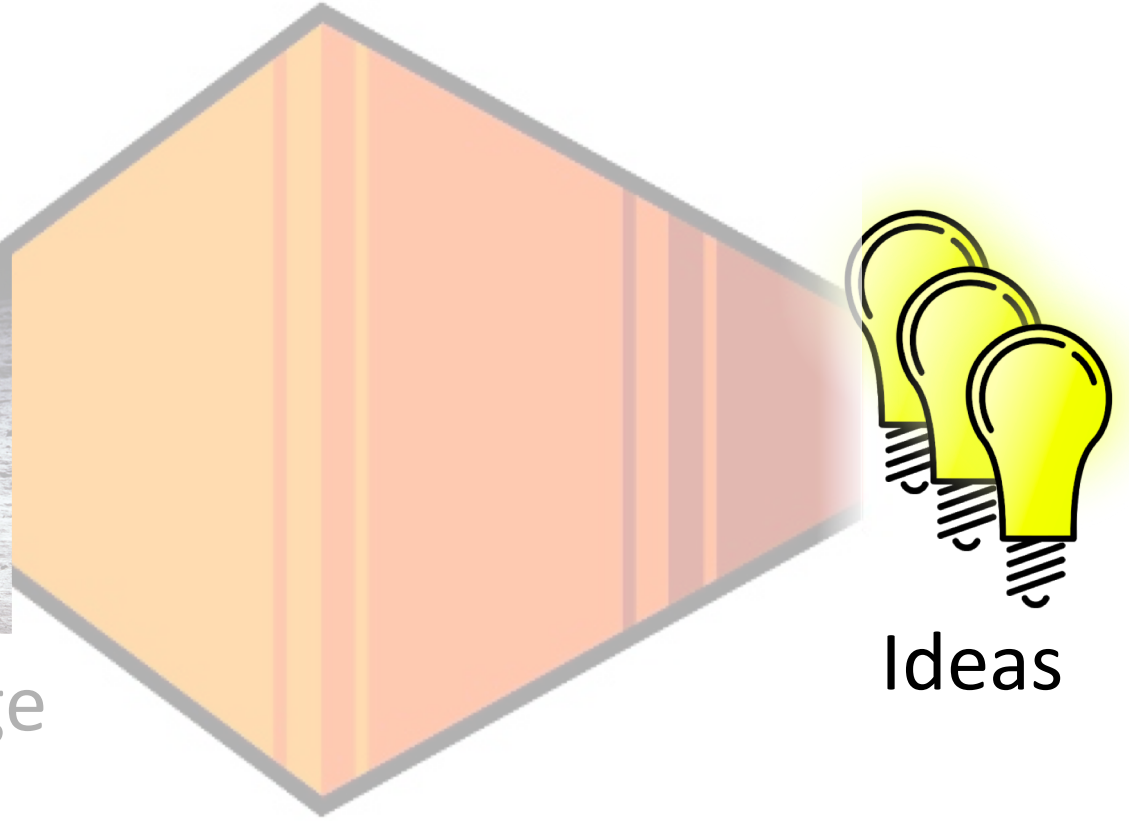
Solve the problem

Final Project:

- Design and build a **web** application
- To help a user learn an introductory topic **interactively**
- Within a **domain of your choosing**
- The interactive experience centers around **media** (image, video, audio..)
- And help them assess their learning with a **quiz**.
- And keep learning through **feedback** from the quiz.
- In under **10 minutes** total

Implementation is iterative.

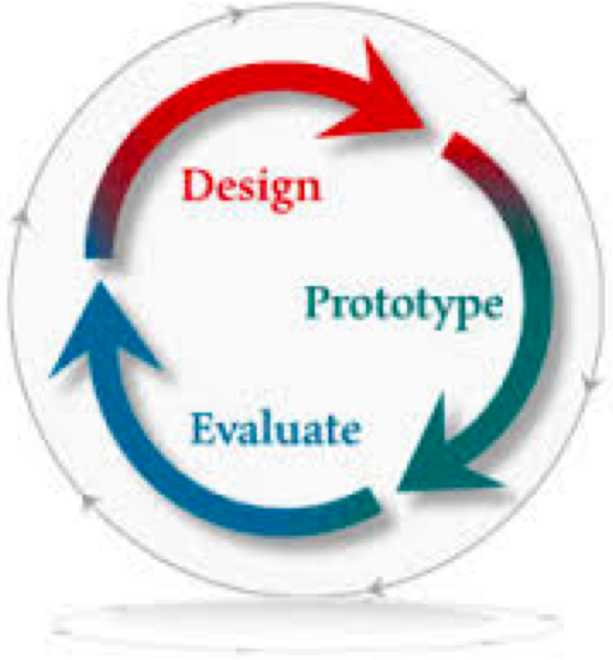
Challenge



Ideas

Ideas

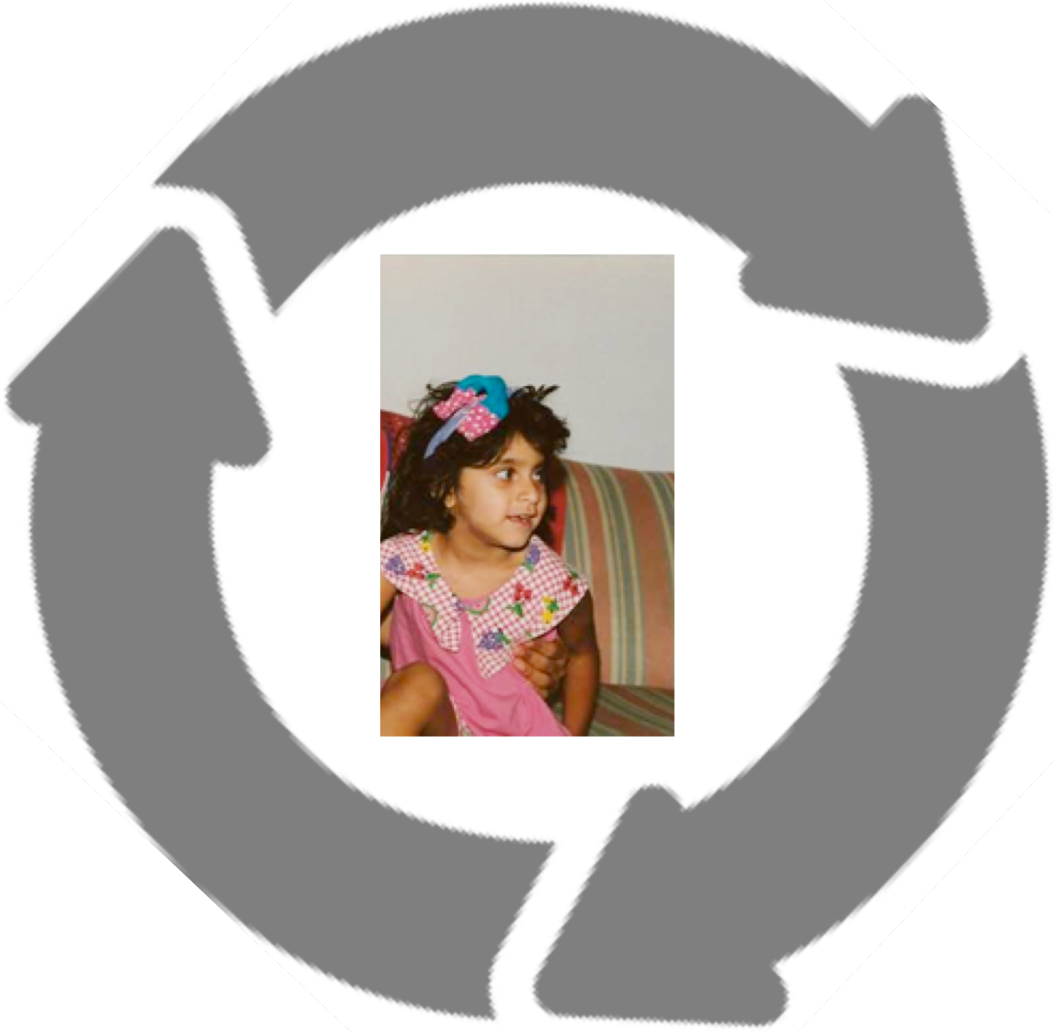
Reality



Implementation

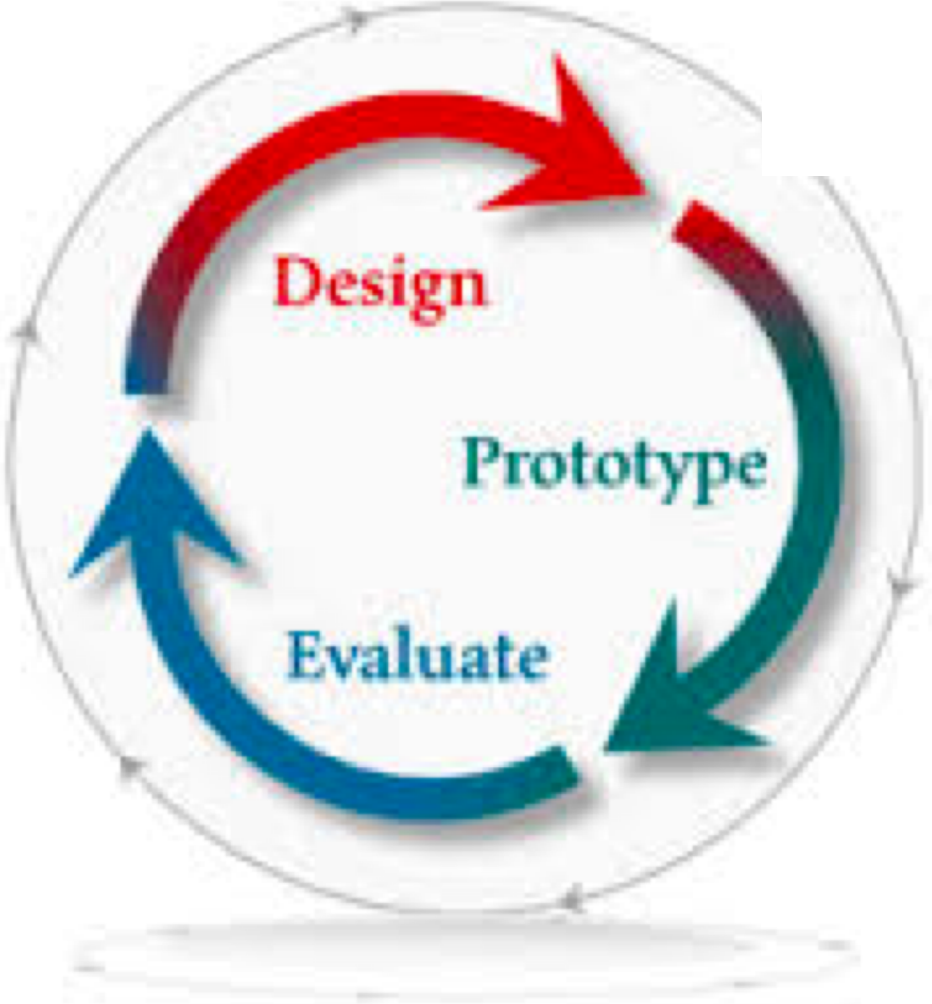
Product

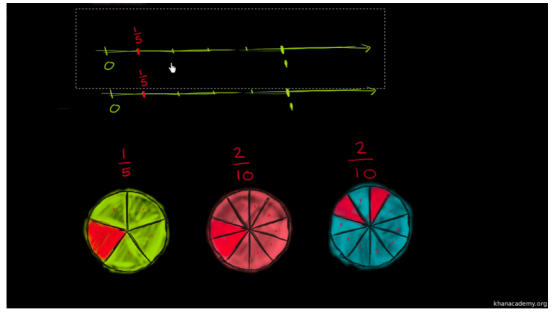
The **user** is at the center of the process



Translating an idea into a prototype is HARD.

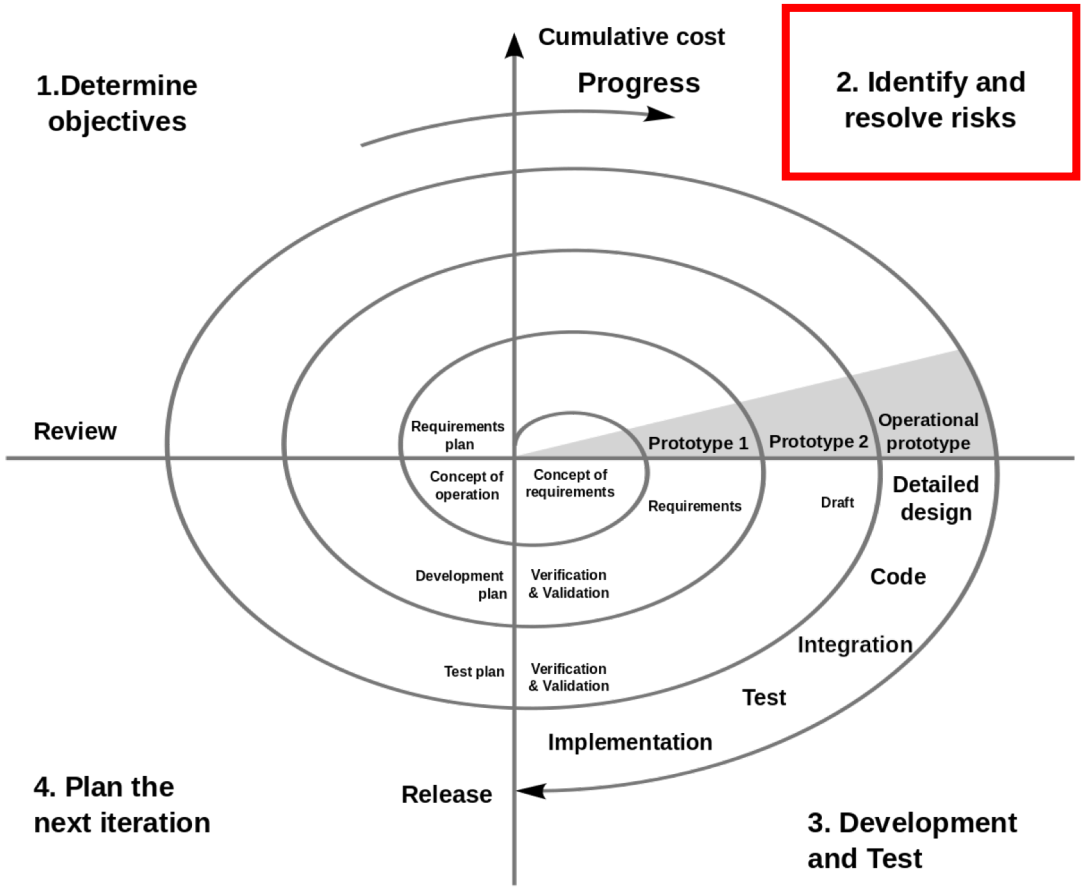
Teach fractions by working through problems slowly.





Low-Fi Prototype

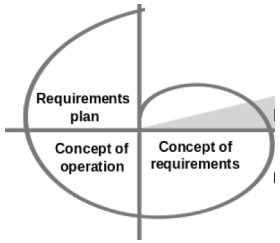
Iterative Design is good because it minimizes risk



The first iteration should be as **low-fidelity** as possible

1.Determine objectives

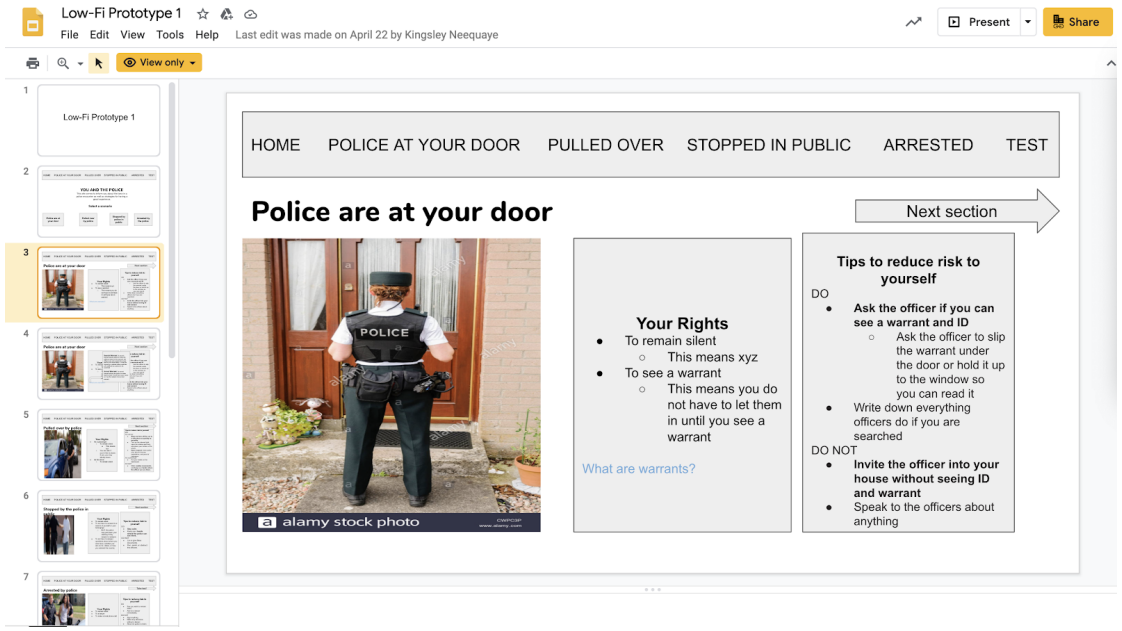
2. Identify and resolve risks

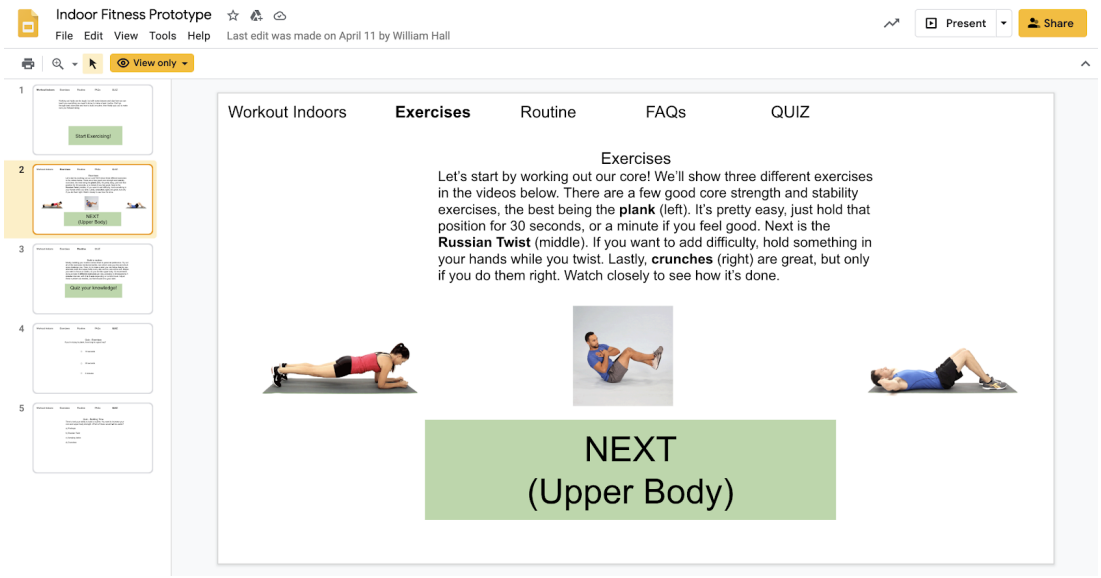


4. Plan the next iteration

3. Development and Test

Low-Fi Prototypes in Google Slides





With feedback from TA's and user tests

Now you’re ready to start implementing

But we’re going to do it iteratively

HW10: Technical Prototype

Warm up due Friday at 11:59pm (grace period until 8am the following day)
Main due Wednesday at 1pm (or otherwise before your section starts) on Courseworks (no grace period – feedback will be given in class)

Note: if your TA feedback meetings are not Wednesdays 1-2:30, then you may have slightly different deadlines. Please turn in assignments according to instructions given by your TA.

Warm-up:

1. **Group**. Which prototype will you continue to iterate on for the rest of the semester? Note: You must get this approved by your TA.
2. **Group**. Do any final iterations on your prototype that you need to before you start implementing it. Make sure you have all the media/data you need to complete it.
    - Include a PDF of the final Google Slides
    - Some groups might have a lot of iteration to do on the prototype. You can continue to work on this during the week, but be sure to turn in some substantial changes for the warm-up.
3. **Group**. Set up a github repo for your project. Include all your group members and your TA.
    - Submit a screenshot of the name and members of the repo.
4. **Group**. Job assignments. List the roles you plan to have and the names of the people assigned to them. Since you have less than 6 people in your group some people will have to be named twice. See below.

HW10: Technical Prototype

General Goal: Figure out what's the biggest technical risks, and create a prototype to test it.

Your HW 11 Goal: Program and end-to-end experience that allows the user to get through the application.
(no graphic design, little interactivity)

What counts as a technical risk?

Anything new.

Anything that has never been done before.

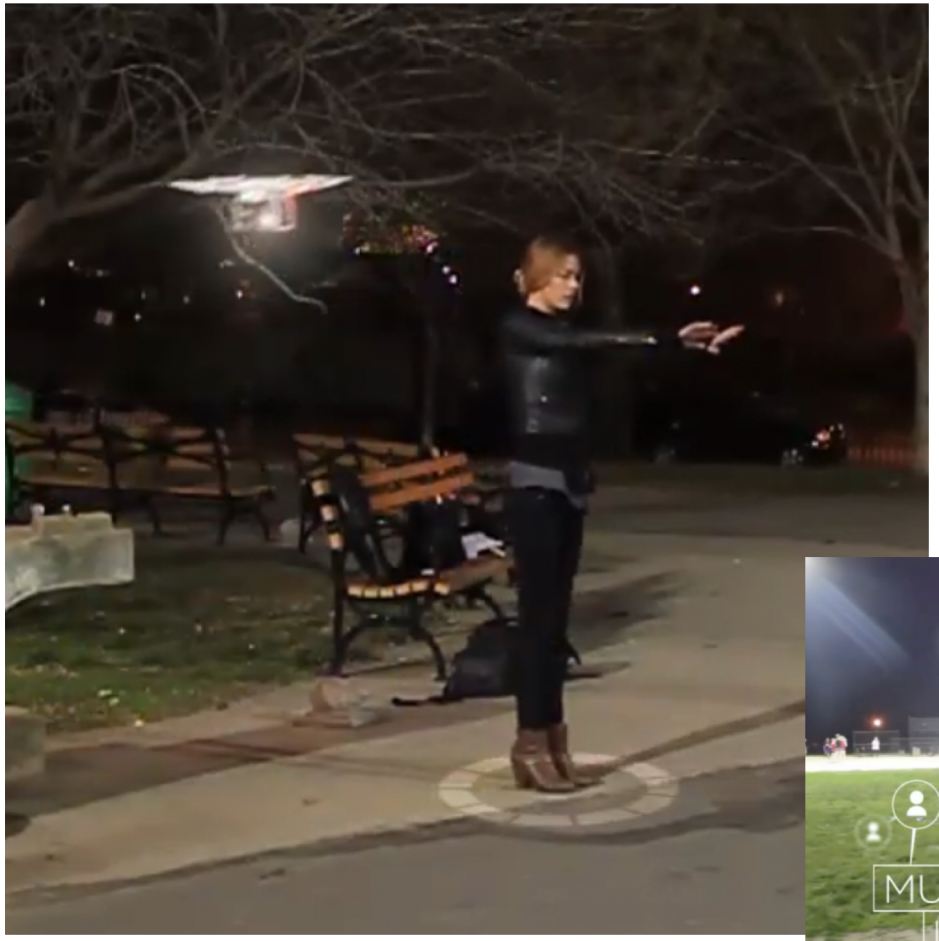
Anything **you** have never been done before.

Let’s watch a concept video for a new technology.

Which features need to be prototyped?
(List EVERYTHING remotely risky)

What tech need to be prototyped?

Drones following you!
And projecting



Menu selection on the ground

Menu selection on the hand

Multi user interaction

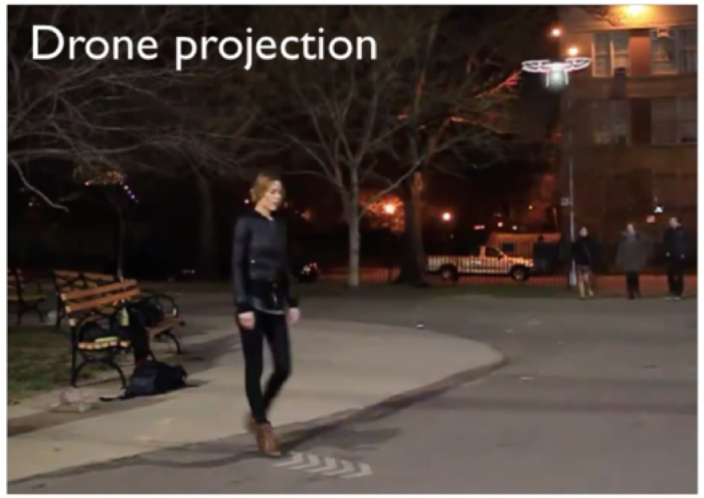
Menu selection on the wall

Now we’re going to watch a video of the things they prototyped

- Which risks did they prototype?
  - How?
- Which risks did they **not** prototype?

DHL
M
PACKSE

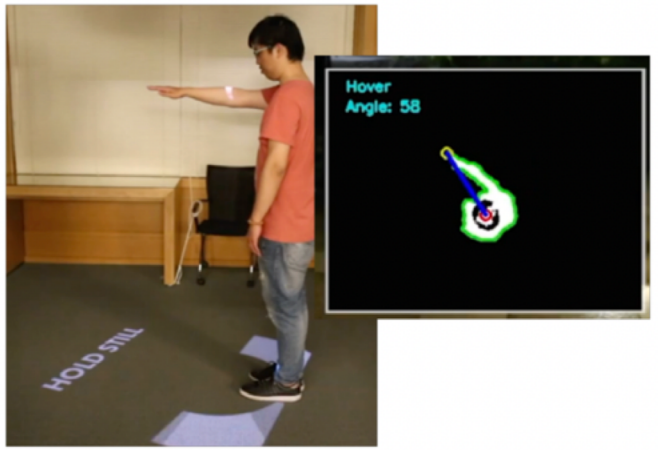
What did they prototype? How?

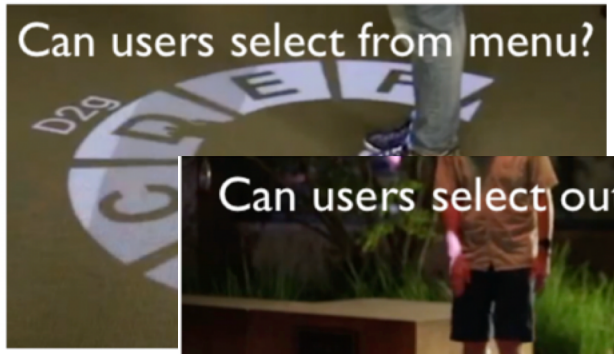


Can the drone carry the stuff?

Can it project visibly?

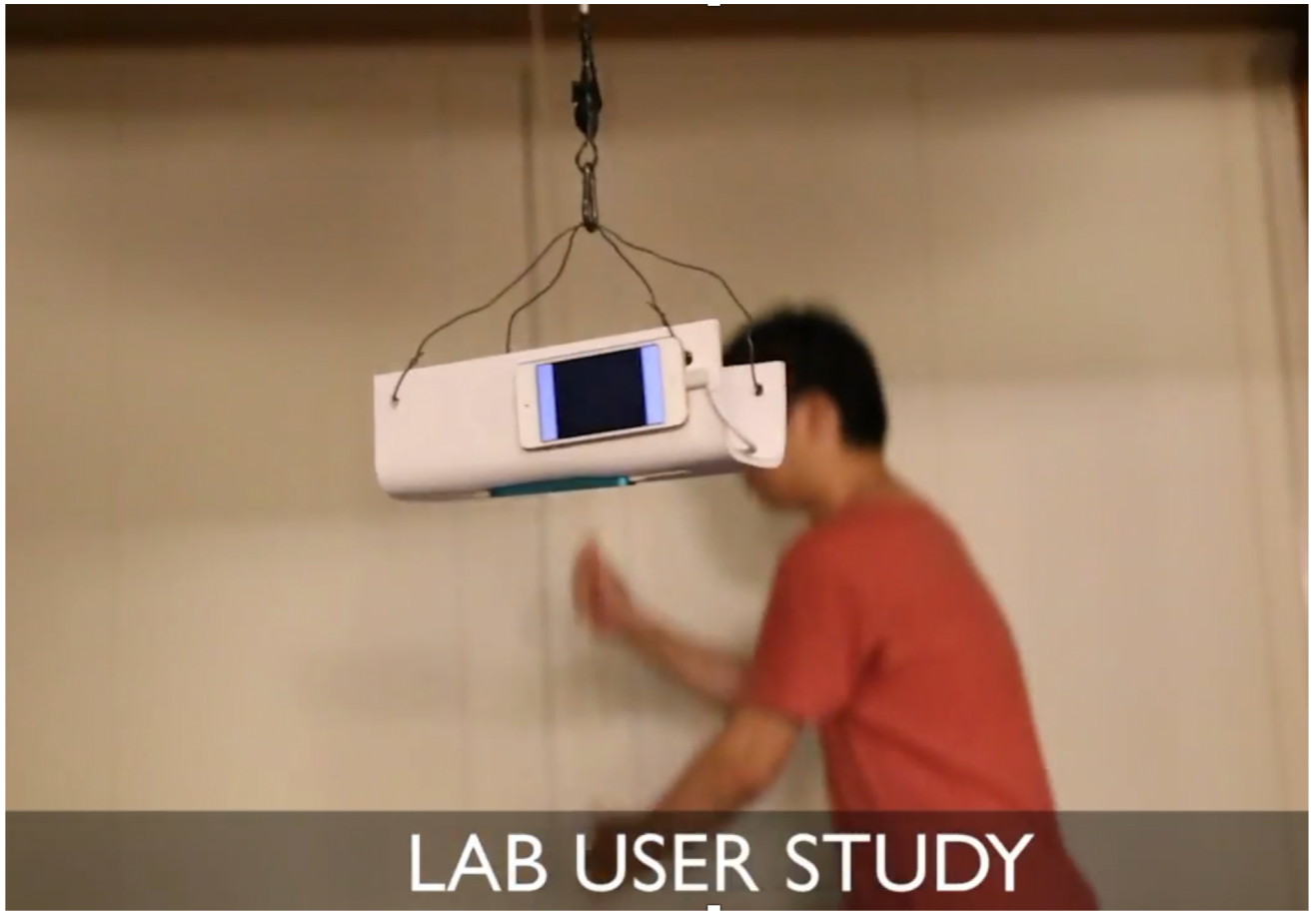
Can the camera detect arm position?





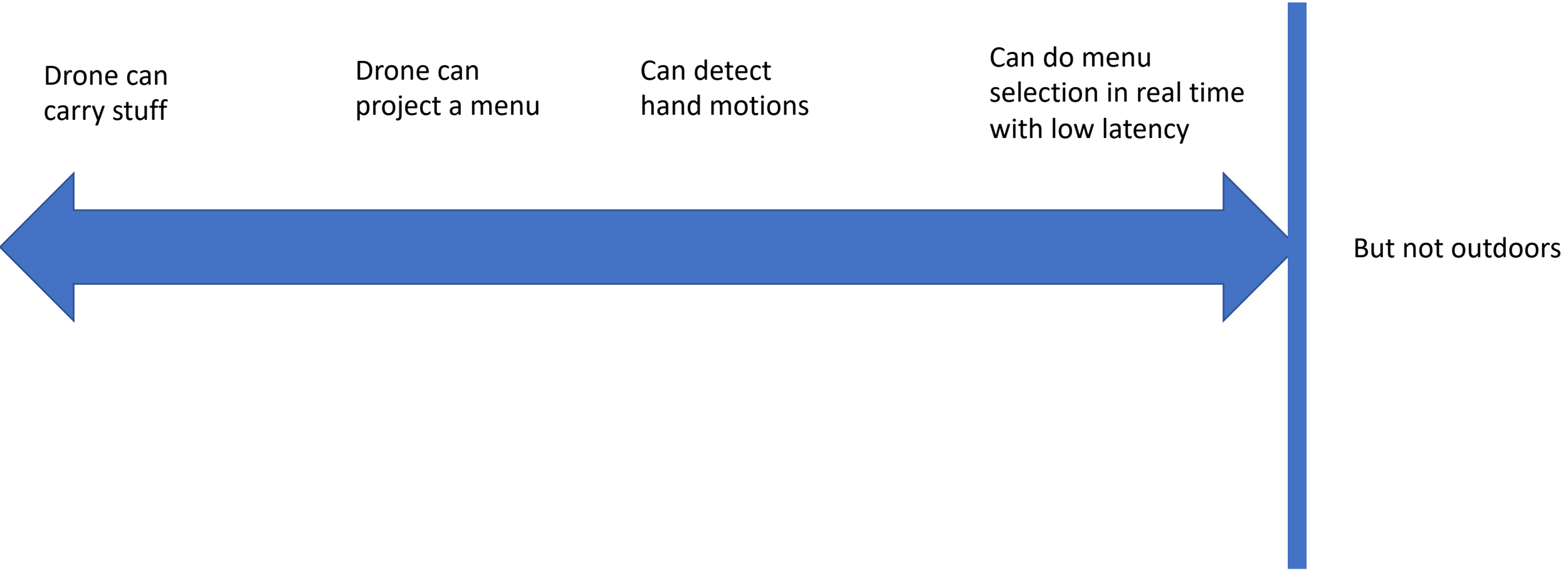
During the outdoor test, what new risk did they discover?

How could they have mitigated this risk?



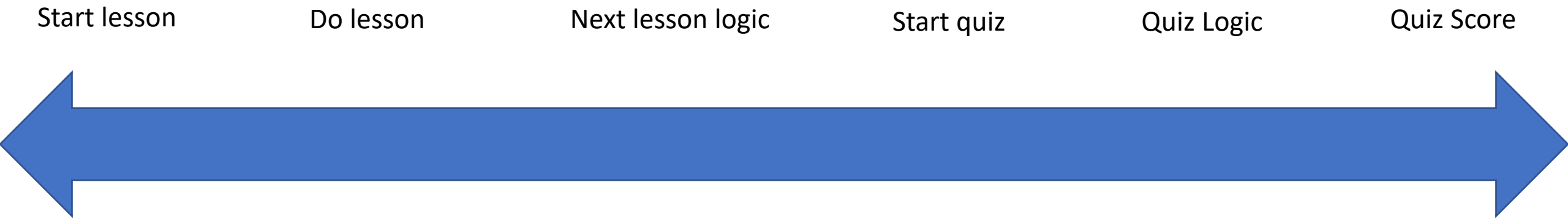
They probably should have tested projecting in flight outdoors before implementing the menu.

Everything in the end-to-end process is a risk



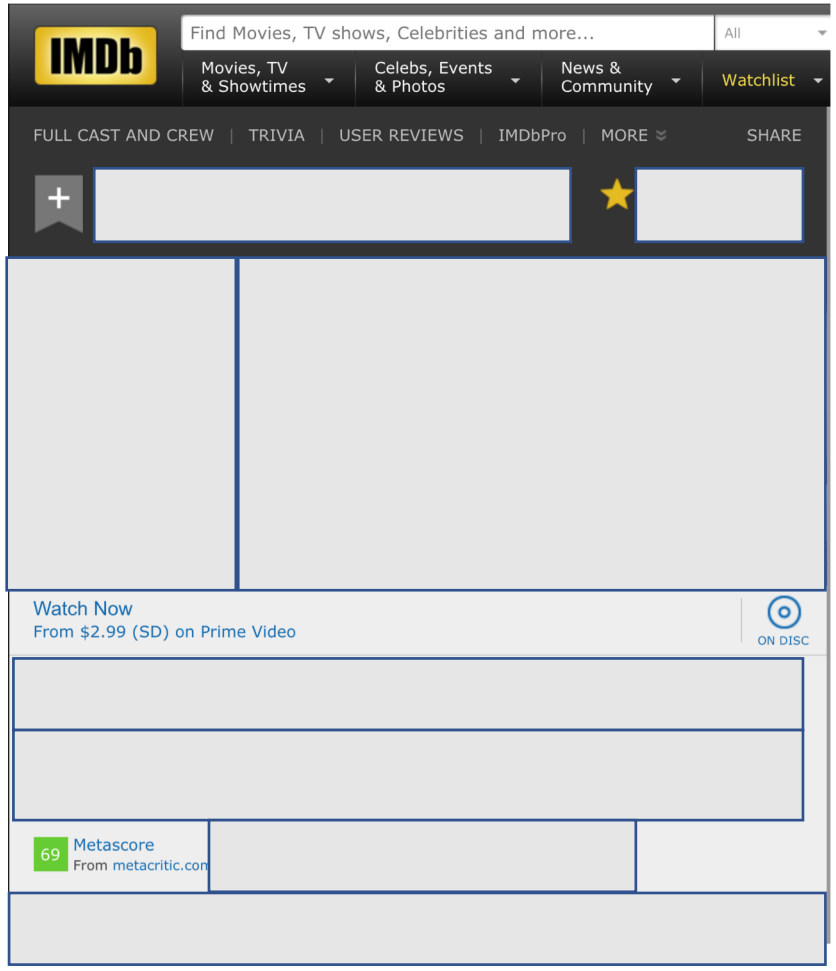
HW 10: Technical Prototypes

Your biggest risk is having an end-to-end working system.



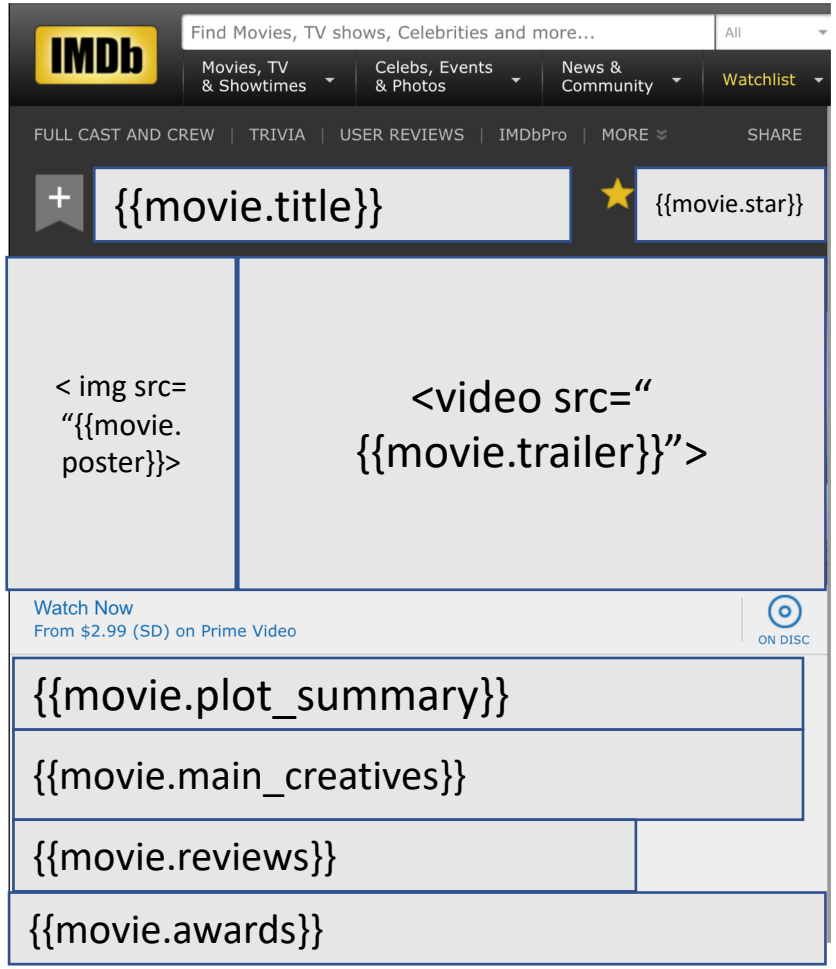
On IMDB, are there 100,000 static HTML pages sitting around?

IMDB uses a template and a database…
And fills in the data dynamically on pageload



Title	Plot summary	poster	year
The Big Lebowski	“The dude” Lebowski, mistaken…	Lebowski.jpg	1998
The Big Short	In 2006-7 a group of investors…	Big_short.jpg	2015
The Big Chill	A group of 7 former roommates	Big_chill.jpg	1983

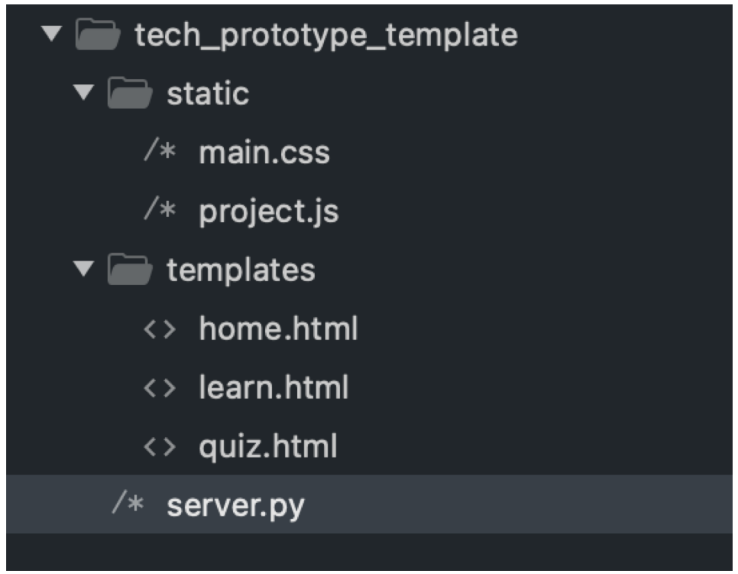
IMDB Template



Why are static pages so bad?

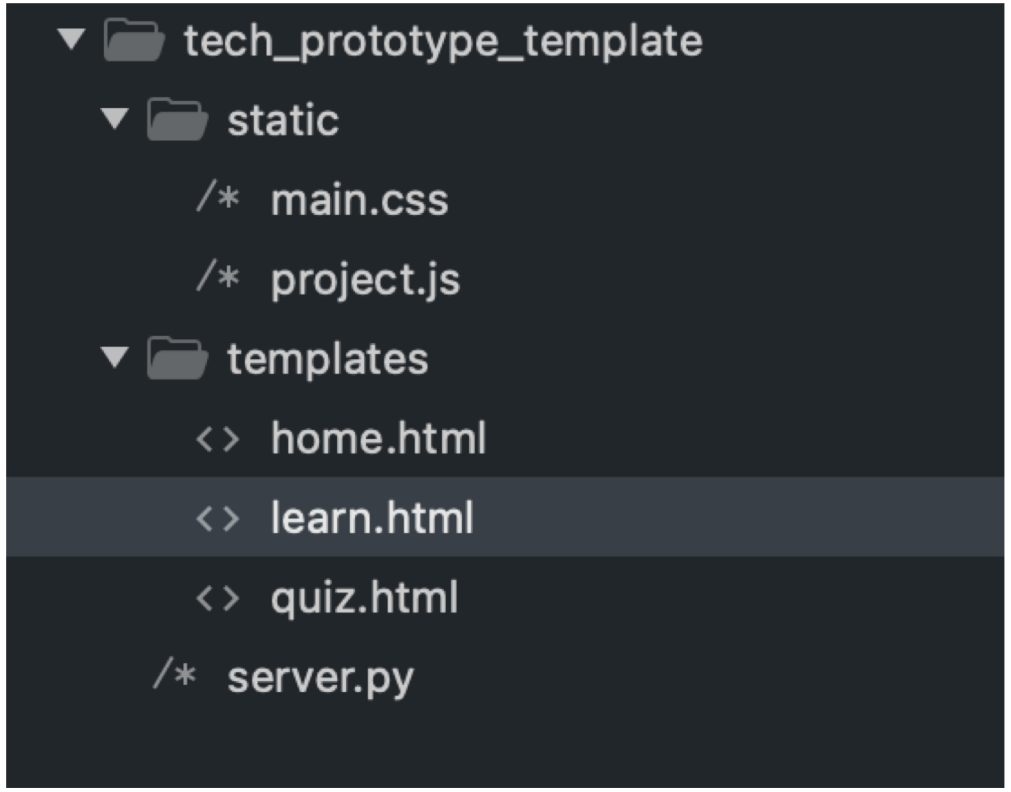
- They are very hard to change.

Represent your data in JSON (server.py)




```python
from flask import Flask
from flask import render_template
from flask import Response, request, jsonify
app = Flask(__name__)


lessons = {
    "1":{
        "lesson_id": "1",
        "title": "Say Hello in Serbian",
        "video": "https://media4.giphy.com/media/f4IitLXgcegOBzXUDe/200w.gif?cid
        "text": "Zdravo",
        "next_lesson":"2"

    },
    "2":{
        "lesson_id": "2",
        "title": "Say Goodbye in Serbian",
        "video": "https://c.tenor.com/6zCJ2mbdXg4AAAAC/ciao.gif",
        "text": "Ciao",
        "next_lesson":"end"
    }
}

quiz_questions = {
    "1":{
        "quiz_id": "1",
        "english": "hello",
        "serbian": "zdravo",
        "next_question":"2"
    },
    "2":{
        "quiz_id": "2",
        "english": "goodbye",
        "serbian": "ciao",
        "next_question":"end"
    }
}

```

Render learning and quizzing templates with correct data




```python
from flask import Flask
from flask import render_template
from flask import Response, request, jsonify
app = Flask(__name__)


lessons = {
    "1":{
        "lesson_id": "1",
        "title": "Say Hello in Serbian",
        "video": "https://media4.giphy.com/media/f4IitLXgcegOBzXUDe/200w.gif?cid=82a1493bcb6xt
        "text": "Zdravo",
        "next_lesson":"2"

    },
    "2":{
        "lesson_id": "2",
        "title": "Say Goodbye in Serbian",
        "video": "https://c.tenor.com/6zCJ2mbdXg4AAAAC/ciao.gif",
        "text": "Ciao",
        "next_lesson":"end"
    }
}
```

```python
@app.route('/')
def home():
    return render_template('home.html')

@app.route('/learn/<lesson_id>')
def learn(lesson_id):
    lesson = lessons[lesson_id]
    return render_template('learn.html', lesson = lesson)

@app.route('/quiz/<quiz_id>')
def quiz(quiz_id):
    question = quiz_questions[quiz_id]
    return render_template('quiz.html', question = question)

if __name__ == '__main__':
    app.run(debug = True)
```

On each page, show the correct data.

127.0.0.1:5000/learn/1

Say Hello in Serbian

Serbian: Zdravo

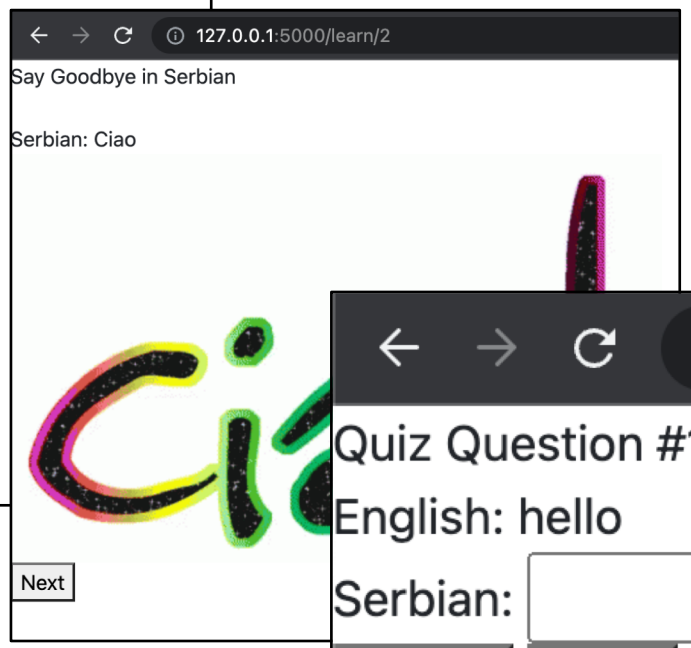
Next

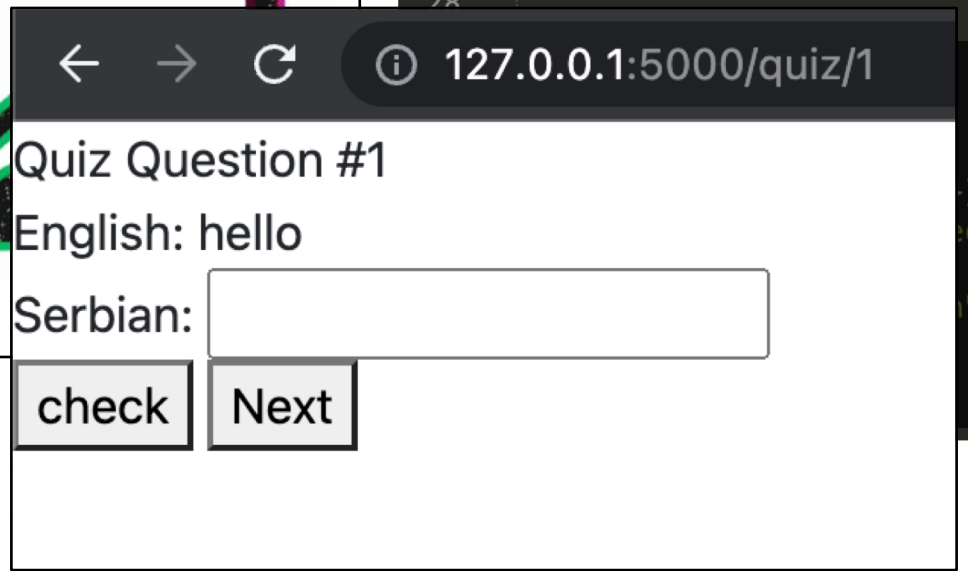
Learn.html template

```
15  <body>
16    <script type="text/javascript">
17      let lesson = {{lesson|tojson}}
18
19      $( document ).ready(function() {
20          $("#next_button").click(function(){
21              if("{{lesson.next_lesson}}" == "end"){
22                window.location.href = "/quiz/1"
23              }else{
24                window.location.href = "/learn/{{lesson.next_lesson}}"
25              }
26          })
27      })
28
29    </script>
30
31    {{lesson.title}}<br>
32    <br>
33    Serbian: {{lesson.text}}<br>
34    <img src="{{lesson.video}}"><br>
35
36    <button id="next_button"> Next </button>
37
38
39  </body>
```

On each page, have logic that advances to the next state.

Learn.html template






```
15  <body>
16    <script type="text/javascript">
17      let lesson = {{lesson|tojson}}
18
19      $( document ).ready(function() {
20          $("#next_button").click(function(){
21              if("{{lesson.next_lesson}}" == "end"){
22                window.location.href = "/quiz/1"
23              }else{
24                window.location.href = "/learn/{{lesson.next_lesson}}"
25              }
26          })
27      })
```

```
}<br>
eo}}"><br>

"> Next </button>
```

Why is representing your application in data (rather than in HTML) so good?

Easy to change **content**

```
lessons = {
    "1":{
        "lesson_id": "1",
        "title": "Say Hello in Serbian",
        "video": "https://media4.giphy.com/
        "text": "Zdravo",
        "next_lesson":"2"
```

Easy to change **logic**

```
lessons = {
    "1":{
        "lesson_id": "1",
        "title": "Say Hello in Serbian",
        "video": "https://media4.giphy.com/
        "text": "Zdravo",
        "next_lesson":"2"
```

```
<script type="text/javascript">
  let lesson = {{lesson|tojson}}

  $( document ).ready(function() {
      $("#next_button").click(function(){
          if("{{lesson.next_lesson}}" == "end"){
            window.location.href = "/quiz/1"
          }else{
            window.location.href = "/learn/{{lesson.next_lesson}}"
          }
      })
  })

</script>
```

Your biggest risk is having an end-to-end working system.

Start lesson | Do lesson | Next lesson logic | Start quiz | Quiz Logic | Quiz Score

Unless you have some huge risky interactive feature!
If do you, contact your TA at once, and we'll modify HW10 for you.

Divide up work with roles

- **Learning portion** roles:
  - Architecting the data
  - Implementing the UI
  - Testing that you can click through the app. (this cannot be done by the learning UI implementer)
- **Quiz portion** roles:
  - Architecting the data
  - Implementing the UI
  - Testing that you can click through the app. (this cannot be done by the quiz UI implementer)

If this does not work out for your group, contact your TA immediately.

HW10: Technical Prototype

Warm up due Friday at 11:59pm (grace period until 8am the following day)
Main due Wednesday at 1pm (or otherwise before your section starts) on Courseworks (no grace period – feedback will be given in class)

Note: if your TA feedback meetings are not Wednesdays 1-2:30, then you may have slightly different deadlines. Please turn in assignments according to instructions given by your TA.

Warm-up:

1. **Group**. Which prototype will you continue to iterate on for the rest of the semester? Note: You must get this approved by your TA.
2. **Group**. Do any final iterations on your prototype that you need to before you start implementing it. Make sure you have all the media/data you need to complete it.
    - Include a PDF of the final Google Slides
    - Some groups might have a lot of iteration to do on the prototype. You can continue to work on this during the week, but be sure to turn in some substantial changes for the warm-up.
3. **Group**. Set up a github repo for your project. Include all your group members and your TA.
    - Submit a screenshot of the name and members of the repo.
4. **Group**. Job assignments. List the roles you plan to have and the names of the people assigned to them. Since you have less than 6 people in your group some people will have to be named twice. See below.

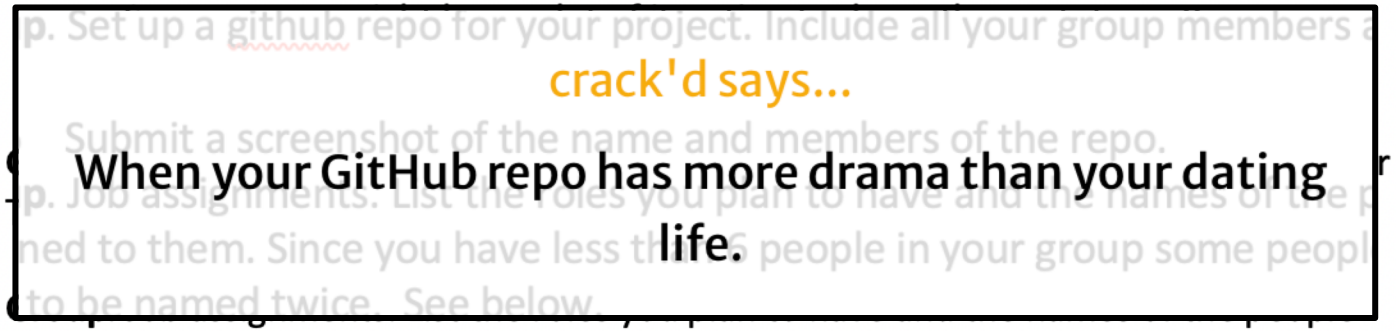
HW10: Technical Prototype

Warm up due Friday at 11:59pm (grace period until 8am the following day)
Main due Wednesday at 1pm (or otherwise before your section starts) on Courseworks (no grace period – feedback will be given in class)

Note: if your TA feedback meetings are not Wednesdays 1-2:30, then you may have slightly different deadlines. Please turn in assignments according to instructions given by your TA.

Warm-up:

1. **Group**. Which prototype will you continue to iterate on for the rest of the semester? Note: You must get this approved by your TA.
2. **Group**. Do any final iterations on your prototype that you need to before you start implementing it. Make sure you have all the media/data you need to complete it.
    - Include a PDF of the final Google Slides



3.
4.
assigned to them. Since you have less than 6 people in your group some people will have to be named twice. See below.

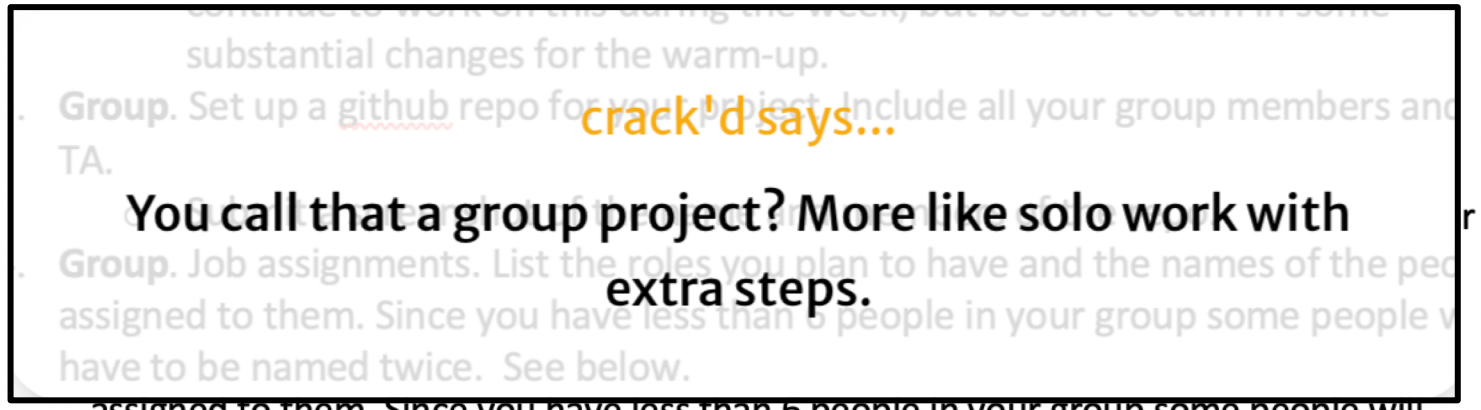
HW10: Technical Prototype

Warm up due Friday at 11:59pm (grace period until 8am the following day)
Main due Wednesday at 1pm (or otherwise before your section starts) on Courseworks (no grace period – feedback will be given in class)

Note: if your TA feedback meetings are not Wednesdays 1-2:30, then you may have slightly different deadlines. Please turn in assignments according to instructions given by your TA.

Warm-up:

1. **Group**. Which prototype will you continue to iterate on for the rest of the semester? Note: You must get this approved by your TA.
2. **Group**. Do any final iterations on your prototype that you need to before you start implementing it. Make sure you have all the media/data you need to complete it.



assigned to them. Since you have less than 6 people in your group some people will have to be named twice. See below.

HW10: Technical Prototype

Warm up due Friday at 11:59pm (grace period until 8am the following day)
Main due Wednesday at 1pm (or otherwise before your section starts) on Courseworks (no grace period – feedback will be given in class)

Note: if your TA feedback meetings are not Wednesdays 1-2:30, then you may have slightly different deadlines. Please turn in assignments according to instructions given by your TA.

Warm-up:

1. **Group**. Which prototype will you continue to iterate on for the rest of the semester? Note: You must get this approved by your TA.
2. **Group**. Do any final iterations on your prototype that you need to before you start implementing it. Make sure you have all the media/data you need to complete it.

crack'd says...

Your document is so organized, it must be hiding a dark secret.

assigned to them. Since you have less than 6 people in your group some people will have to be named twice. See below.

HW10: Technical Prototype

Warm up due Friday at 11:59pm (grace period until 8am the following day)
Main due Wednesday at 1pm (or otherwise before your section starts) on Courseworks (no grace period – feedback will be given in class)

Note: if your TA feedback meetings are not Wednesdays 1-2:30, then you may have slightly different deadlines. Please turn in assignments according to instructions given by your TA.

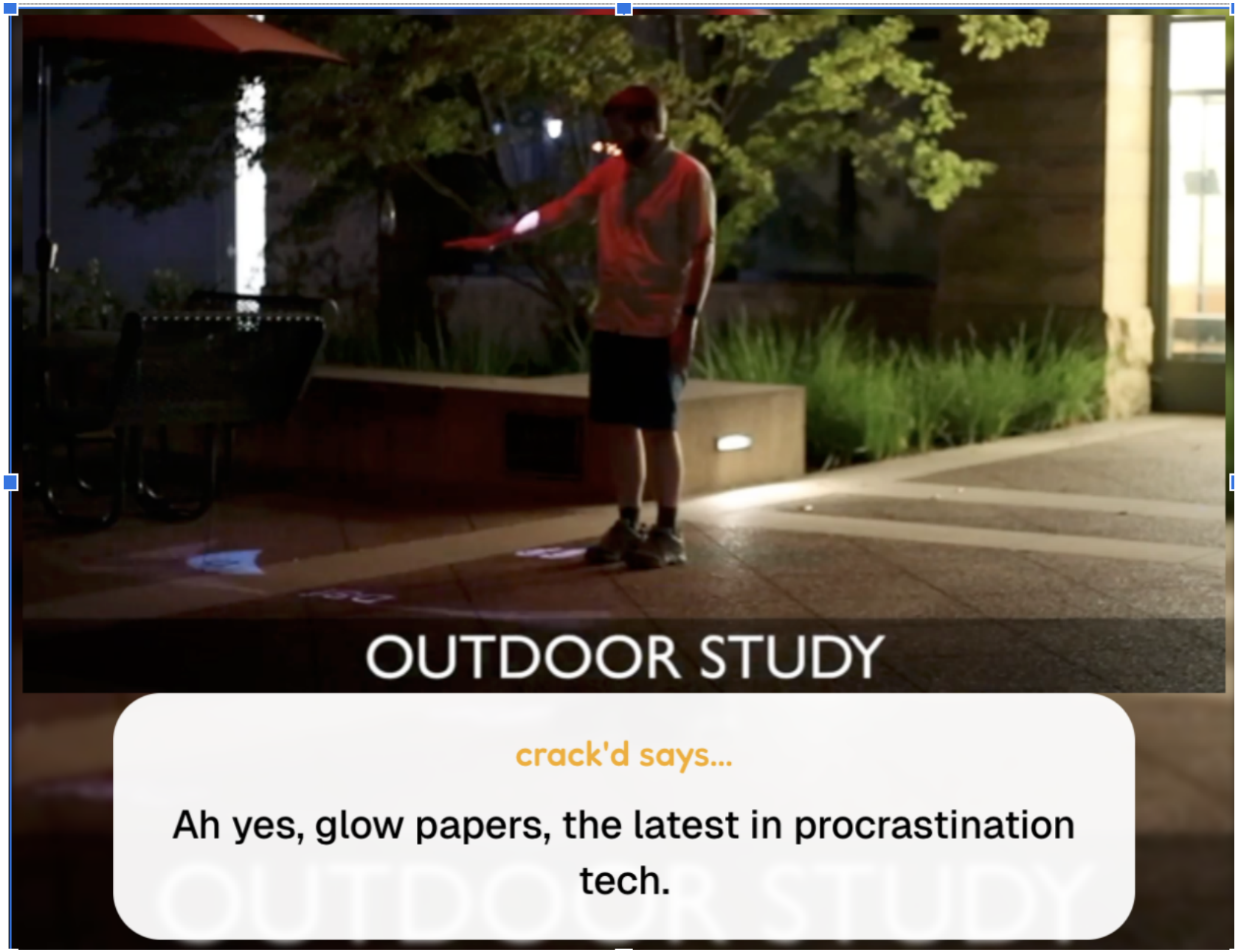
Warm-up:

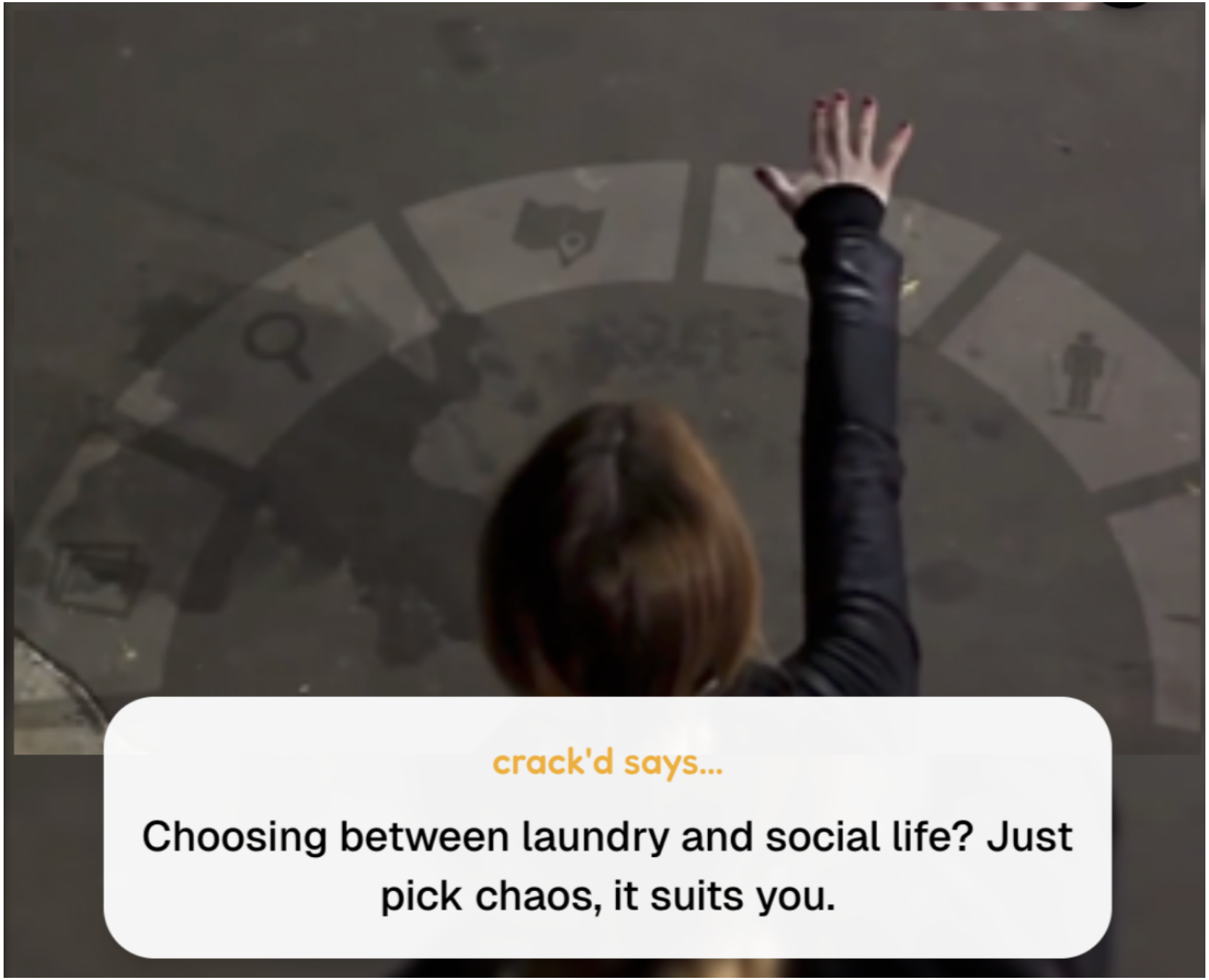
1. **Group**. Which prototype will you continue to iterate on for the rest of the semester? Note: You must continue to work on this during the week, but be sure to turn in some substantial changes for the warm-up.
2. **Group**. Set up a github repo for your project. Include all your group members and y TA.
3. **Group**. Job assignments. List the roles you plan to have and the names of the peop assigned to them. Since you have less than 6 people in your group some people wi have to be named twice. See below.
   - Submit a screenshot of the name and members of the repo.
4. **Group**. Job assignments. List the roles you plan to have and the names of the people assigned to them. Since you have less than 6 people in your group some people will have to be named twice. See below.

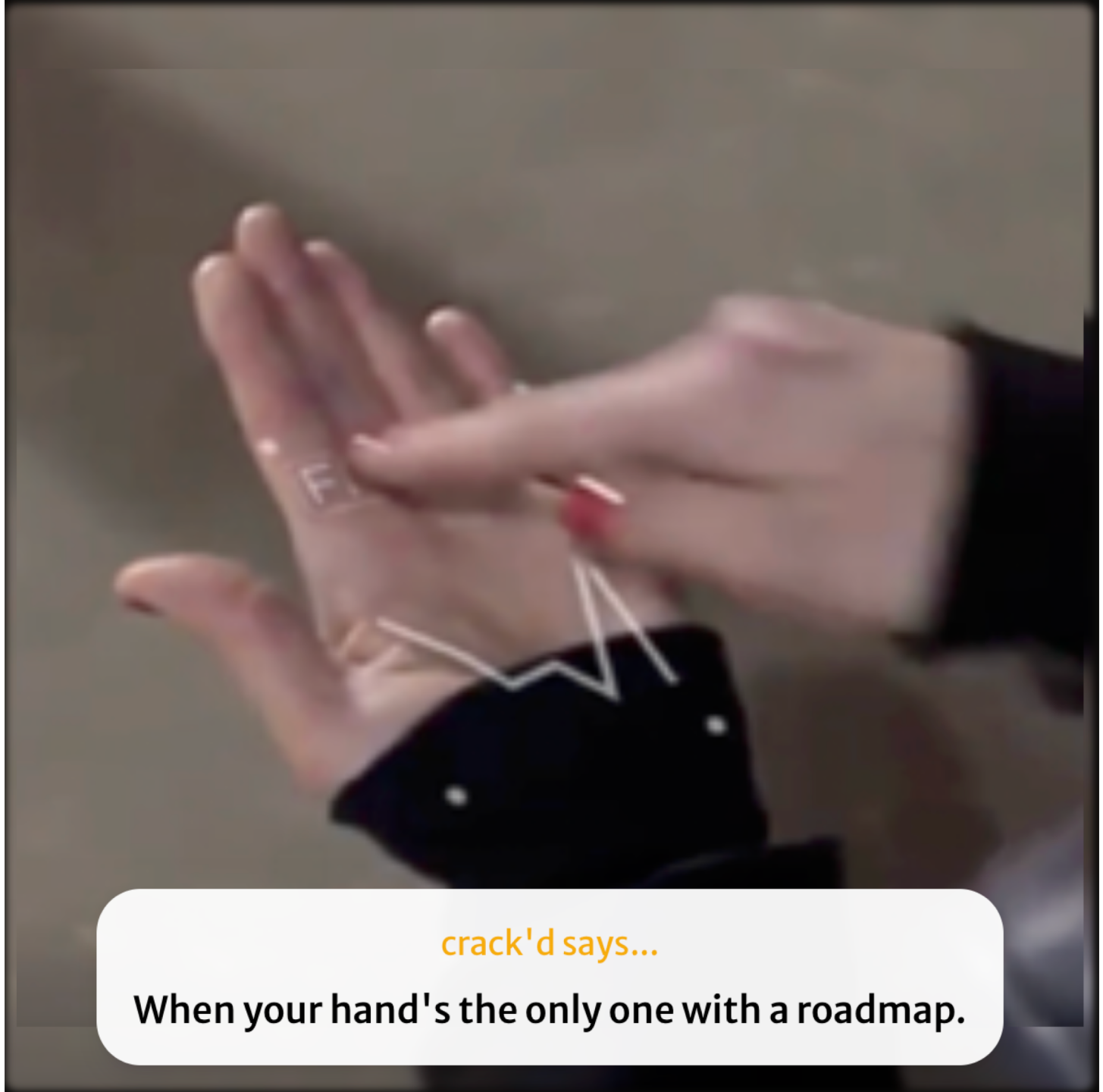
crack'd says...

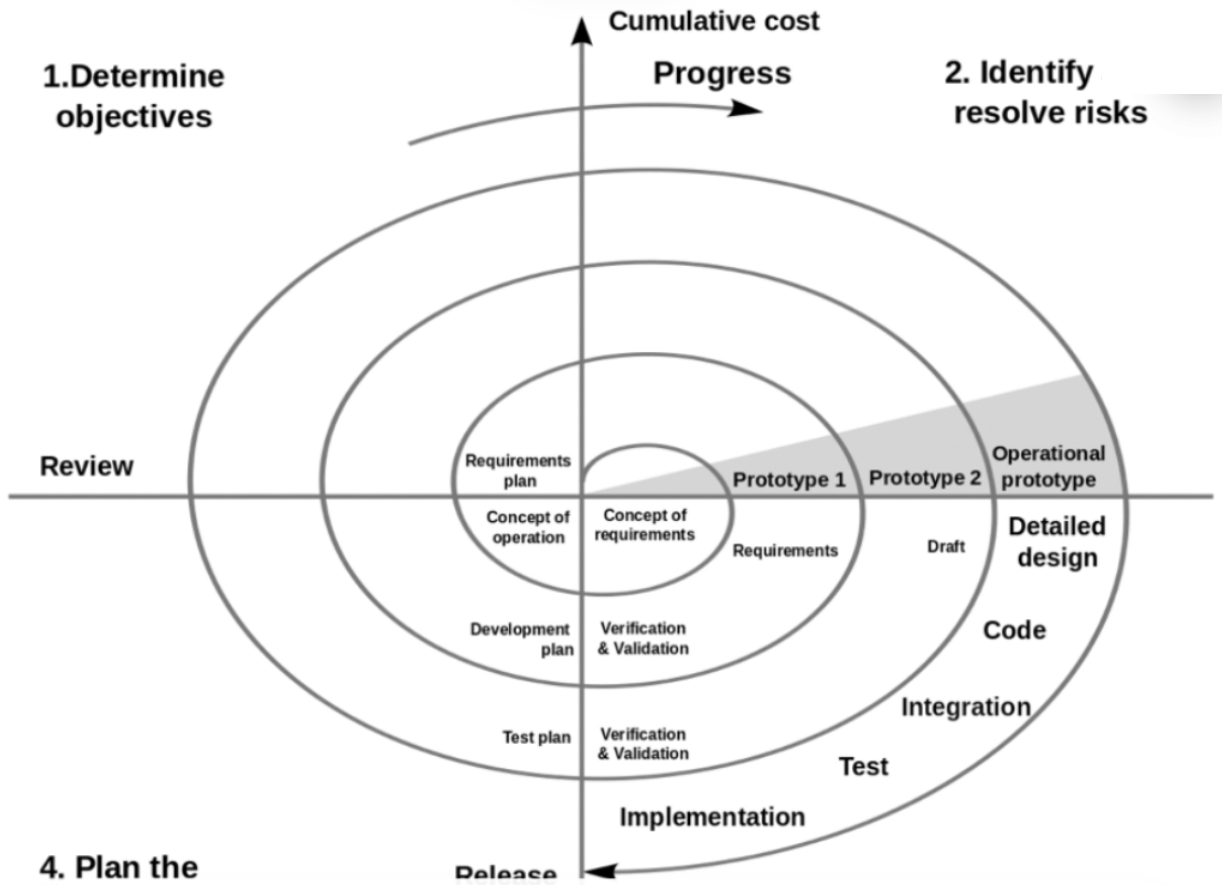
That warm-up section is just a sneak peek into your panic mode.

LAB USER STUDY
crack'd says...
Your phone deserves better friends than you.


OUTDOOR STUDY
crack'd says...
Ah yes, glow papers, the latest in procrastination tech.

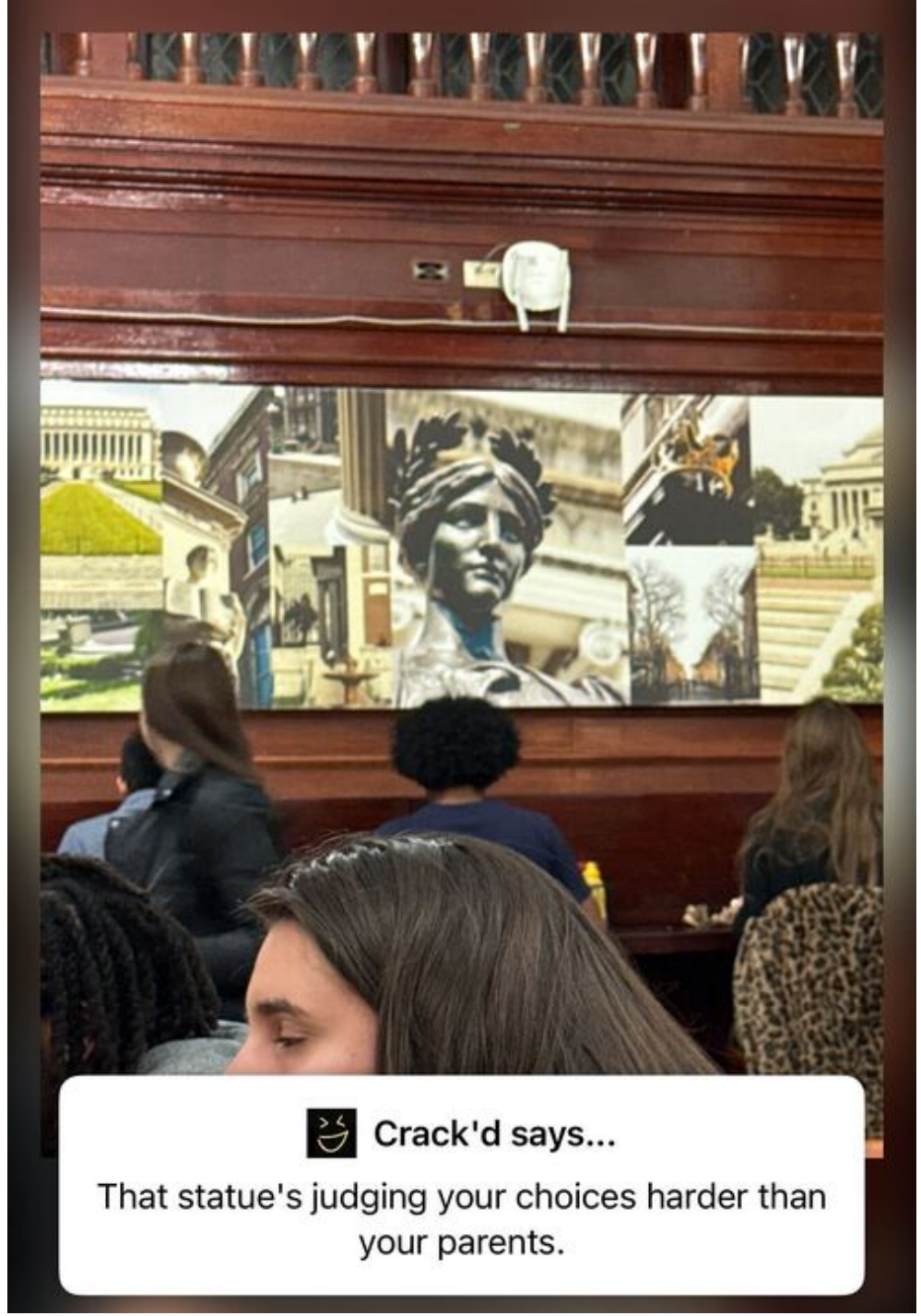

crack'd says...
Choosing between laundry and social life? Just pick chaos, it suits you.


crack'd says...
When your hand's the only one with a roadmap.



crack'd says...

Your project management skills are as useful as a screen door on a submarine.


Crack'd says...
That statue's judging your choices harder than your parents.

How to Build our Future Robot Overlords

Going from 0→1 with Robotics at the Edge