Week 1: Developing XR Experiences

1. XR Experiences:

   a. Environment: Physical and Virtual

   b. 3D Characters: Avatars, 3D Models, Data

   c. Interactions: Explicit and Implicit

   a,b: Context

   b,c: Story

   - Context: In VR: design the VR environment to fit in real world

     In AR: design the AR environment to blend with real world.

   - Story: In VR: tell the story at the intersection of the real and virtual (When R matches V)

     In AR: tell the story as the union of the real and virtual (Both R and V)

2. Developing XR Experiences:

   a. Basic XR Experience:

   - XR Designer (scenes)

   b. Advanced XR Experience:

   - XR Designer (scenes)
   - 3D Artist (3D models, animation)
   - XR Designer (lights & materials)
   - XR Designer (sounds)
   - XR Developer (scripts)

   c. Contents and Interactions:

   - Content:

     - 3D models & animations (Basic VR, Immersive VR, Marker-based AR, Marker-less AR)
     - Lights & shadows (Basic VR, Immersive VR, Marker-based AR, Marker-less AR)
     - Environment (Immersive VR, Marker-less AR)
     - Physics (Immersive VR, Marker-less AR)
     - Spatial sound (Immersive VR, Marker-less AR)
     - Menus & HUDs (Immersive VR, Marker-based AR, Marker-less AR)

   - Explicit Interaction

     - Cursor (Basic VR, Marker-based AR)
     - Controller / hand (Immersive VR, Marker-less AR)
     - Speech / voice (Immersive VR, Marker-less AR)

   - Implicit Interaction

     - GPS location (Marker-based AR, Marker-less AR)
     - Fiducial marker (Marker-based AR)
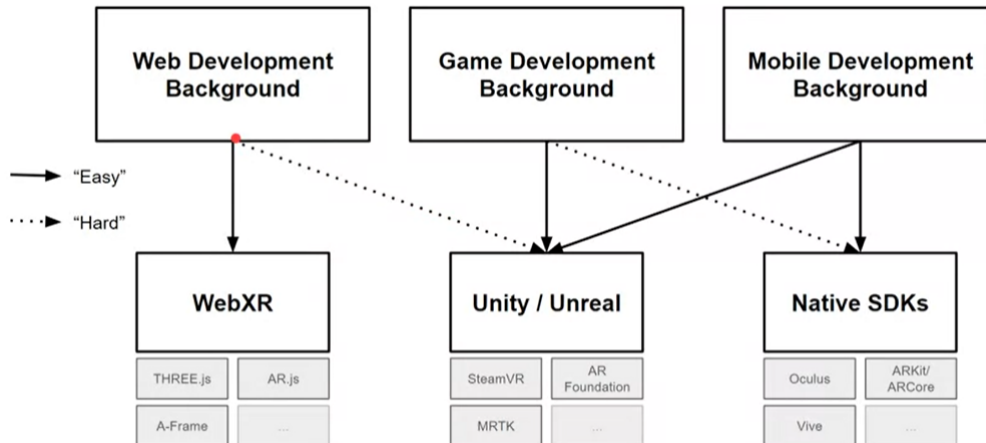   - Environmental features (Marker-less AR)

3. Get Start to design XR:

   a. Design:

   - Design Process
   - Design Methods

- Authoring Tools
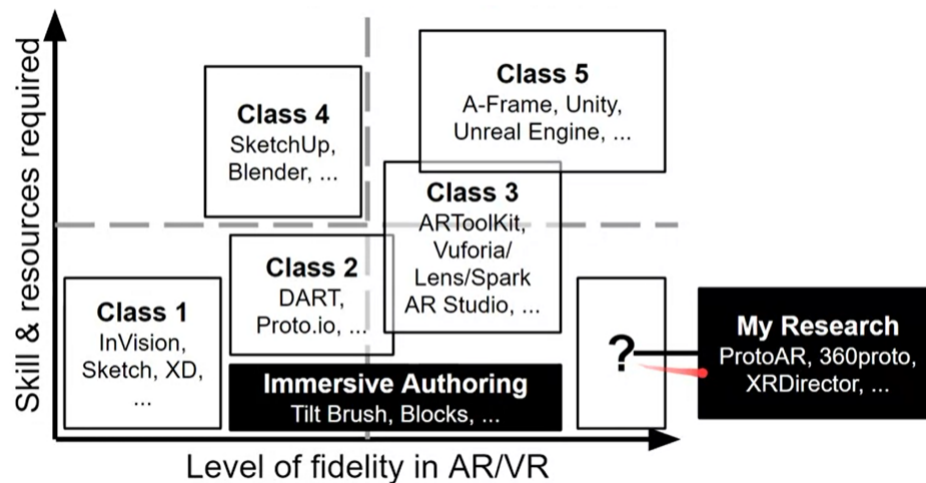- Design Guidelines
- Usability Evaluation

  Development:
- Development Process
- Development Toolkits
- Programming Tools
- Design Patterns
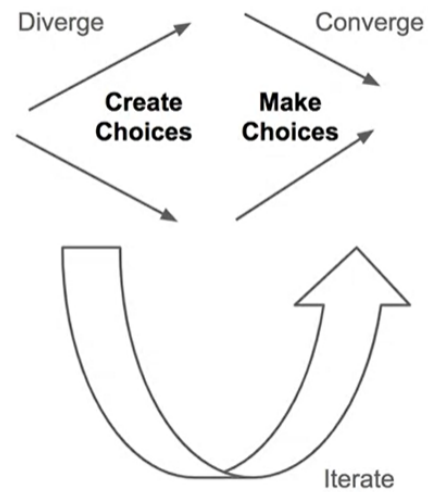- Interaction Tracking
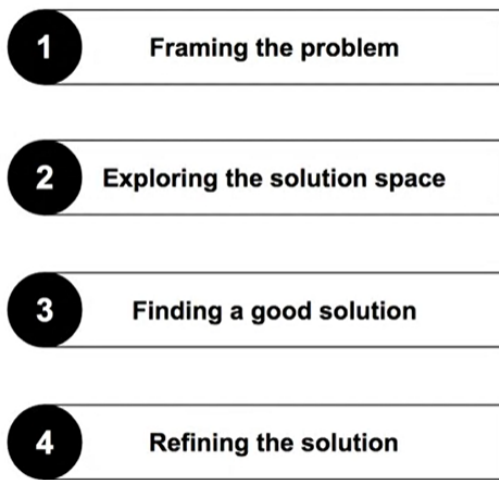
b. Paths of creating XR:

c. Tools of Developing XR:

d. To get started:
- Try out current XR devices
- Make your first XR app
- Join XR communities
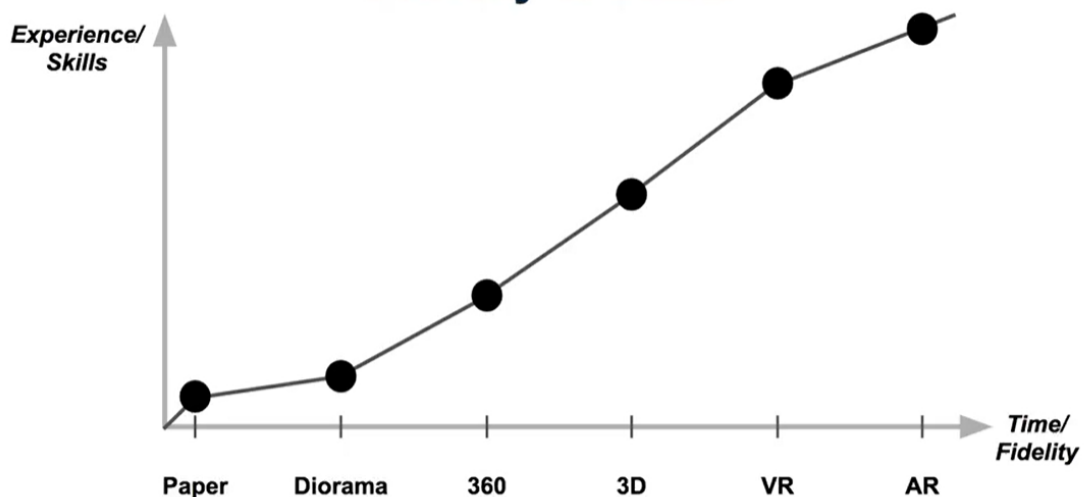- Start reading on XR research

4. Interaction Design Process:

1 Framing the problem	
2 Exploring the solution space	
3 Finding a good solution	
4 Refining the solution	

Diverge → Converge

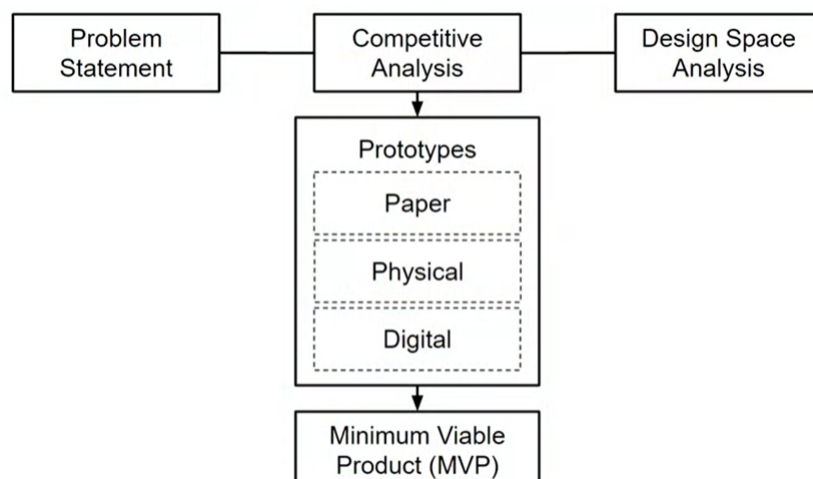Create Choices   Make Choices

Iterate

a. Processes:

- Needfinding & Brainstorming: Framing the problem via scenarios & use cases, personas, and competitors.
- Storyboarding & Prototyping: Creating mockups using paper and digital tools, involving XR devices. (Storyboards, Physical prototype, Digital prototype)
- Development & Testing: Creating application using WebXR, Unity/Unreal or native SDK.
- Development & Analytics: Deploying application on XR device, collecting data during usage.

b. Fidelity and Skills:

Experience/Skills (vertical axis) vs Time/Fidelity (horizontal axis): Paper, Diorama, 360, 3D, VR, AR

5. Intro to XR Design:

a. Design Process:

Problem Statement → Competitive Analysis → Design Space Analysis

Competitive Analysis → Prototypes (Paper, Physical, Digital) → Minimum Viable Product (MVP)

b. Methods & Tools: Needfinding & Brainstorming, Lo-Fi & Hi-Fi Prototyping, Digital & Immersive Authoring, Development Analytics.

c. Guidelines & Evaluation: Usability, Usefulness, Utility, User Acceptance

Week 2: Design Thinking, Ethics, and Guidelines for XR

1. Design Thinking:
    - Design Stages:

    a. Empathize
        - Can't just ask / observe: most people don't use XR frequently

    b. Define
        - Don't just commit to AR/VR

    c. Ideate
        - Ignore technical constraints

    d. Prototype
        - Focus on the unknowns

    e. Test
        - Act as wizard to fill the gaps

    f. Implement
        - Use the bet suited tools
    - Scenarios & Personas:

    a. Scenario Elements:
        - Setting: situation (state of the system, context of use) in which activity occurs.
        - Agent(s): people performing the activity described in scenario.
        - Goals: objectives agents are trying to accomplish.
        - Actions: activities that agents do to accomplish their goals.
        - Events: things that happen to users while doing the activities in the scenario.

    b. Personas: To represent different kinds of potential users.
        - Descriptions:

Descriptive Persona Name template showing: Photo that shows user or context attributes, "Quote that best represents this persona"; Goals (Goal 1, Goal 2); Frustrations/Needs (Frustration/Need 1, Frustration/Need 2); Motivations (Incentive, Fear, Achievement, Growth, Power); XR Experience (Novice 1 □□□□□□ 7 Expert, Designer □□□□□ Developer, ___ months User, ___ times per week User); XR Devices (VR: □Oculus □Vive, AR: □Smartphone □HoloLens, Other: ____); XR Apps (VR: □Tilt Brush □Quill, AR: □Reality Composer □Aero □Torch, Other: ____); Primary: our design completely satisfies their goals/needs; Secondary: our design meets most of their goals/needs; Negative: our design does not target their goals/needs

        - Do's and Don'ts of Personas:

        (1) Do's:
            - Clarify how they were created.

- - - Base them on data / facts.
      - Let them reflect your diverse target user groups.
      - Include negative personas to make it clear who you are not designing for (yet).

    (2) Don'ts:

      - Just make assumptions.
      - Stereotype personas.
      - Confuse negative personas with people you don't want to use your solution.
  - Competitive Analysis & Design Space Analysis:

    a. Competitors:

      - Direct (Overlapping features/services/users)
      - Partial (Subset of features/services/users)
      - Indirect (Similar but in a different way)
      - Parallel (Similar but no direct competition)
      - Analogous (Non-competitor for inspiration).

    b. Competitive Analysis:

      - Include: Strengths, Weaknesses, Opportunities, Threats.
      - Don't try to include all features and do everything better. Rather, look for gaps.

    c. Design Space:

      Questions => Options => Criteria

  - Guiding Principles:

    a. Designer is not User.

    b. Hard to Unlearn.

    c. Be a Senior Designer: find path with different options in one stage and go forward/backward.

    d. Prototype is Product.

2. Design Ethics:

  a. Design Ethics Principles:

    - Autonomy: User is in control.
    - Transparency: User is informed about how to control.
    - Safety: User is protected from danger, risk and injury.

  b. Ethical Review:

    - Define Purpose, Procedure, Duration
    - Maximize Benefits / Limit Trade-offs.
    - Minimize Risks.
    - Empower Participants (How to enable users to stay in control)

  c. Design Ethics Review:

    - Context & Situation:

      Where is the XR device being used?

      Who is around the user?

    - Sensory Information & Data:

      What does the XR device see?

      What kind of data is acquired?

    - Processing Pipeline:

What happens on-device?

What happens in the cloud?

- Data Ownership & Governance:

Who owns and manages the data?

How is the user informed?

d. Design Impact:

- Before: set expectations of users for new reality.
- During: balance realism, comfort and safety.
- After: help users transition back to old reality.

e. Keep in Mind:

- Consider the cost-benefit ratio.
- Stay aware (always learn).
- Be honest (know the tech).

3. Design Guidelines:

a. Existing Design Guidelines:

- By Vendors (Platform driven): Oculus, Google, Mozilla.
- By Designers (User oriented)
- By Practitioners (Experience based)
- By Researchers (Empirically derived)

b. Troubles with Design Guidelines:

- Rapidly evolving best practices:

Still a moving target, lots to learn about XR design.

Slowly emerging design patterns, but often change with OS updates.

Already major differences between device platforms.

- Challenges with scoping guidelines:

Often too high level, like "keep the user safe and comfortable" or too application/device/vendor-specific.

- Best guidelines come from learning by doing:

Test your designs early and often, learn from your own "mistakes". Mind differences between VR and AR, but less so between devices.

c. Do's and Don't of XR Design:

- Do's:
  - Offer alternative ways to travel.
  - Use interface metaphors. (An interface metaphor is a set of user interface visuals, actions and procedures that exploit specific knowledge that users already have of other domains)
  - Use physical affordances. (Physical affordance is a design feature that helps, aids, supports, facilitates, or enables doing something physically)
  - Minimize explicit interactions.
  - Put important elements in the foreground, avoid periphery.
  - Provide visual and audio cues to guide the users through tasks.
- Don'ts:
  - Move user without their input.
  - Reinvent reality.

- Just copy/paste real world.
- Rely on gestures or speech.
- Have users read a lot of text.
- Put lots of content in the HUD.
- Remap controller buttons, break conventions.

4. Design Jams:

   a. Different people gather together to design.

   b. Things to think about:

   Goals, Tasks, Deliverables, Devices, Tools, Time.

   c. Planning Design Jams:

   - Start by thinking about who should participate.
   - What happens before and after the design jam? (pre and post task, data to collect about participants)
   - Make sure everyone is on the same page. (demo, reference)
   - Organize all the materials.

   d. Running Design Jams:

   - Give the problem statement.
   - Have a clear design prompt.
   - Use a consistent structure. (start with a goal, provide a description of each task and specify deliverables)
   - Collect feedback.
   - Have clear roles assigned in your team (keeping time, answering questions, taking notes)
   - Do a dry-run.

   e. Participating in Design Jams:

   - Making a plan is crucial.
   - Sketch to brainstorm.
   - Make use of space.
   - Divide workload.
   - Key deliverables: Scenarios, personas, competitor (inspiration), user flows & wireframes, physical / digital prototype, video.
   - Alternative designs: Document design evolution as much as possible.
   - Practice and solicit feedback from others.

   f. Things you could/should do:

   - Mini project plan (define approach, milestones and roles)
   - Scenarios (spell out settings, agents and activities)
   - Personas (communicate who you are designing for and who not)
   - Storyboards (use AR/VR templates, flesh out promising ideas)
   - Physical prototypes (demo envisioned interface and interactions)
   - Digital prototypes (implement and test key aspects)

5. Design Critiques:

   a. I like / I wish / What if Template:

   - I like: What you like about it.
   - I wish: What could be improved.
   - What if: How could it be improved.

   b. Keep in mind:

   - Design guidelines: Feedback, affordances, mappings, constraints.

- Usability: Learnability, memorability, efficiency, errors, satisfaction.
- Accessibility and equity
- Audience and personal experience
- Concerns and impact: Physical, mental, privacy, social, cultural, environmental, ...

Week 3: Storyboarding & Physical Prototyping for XR Experiences

1. Brainstorming and Problem Framing:

   a. Ideate-Implement-Inspect-Iterate Design Cycle

   b. "Killer-App" Problem:

   - Novelty is not Creativity.
   - Find key app and improve it in one way.
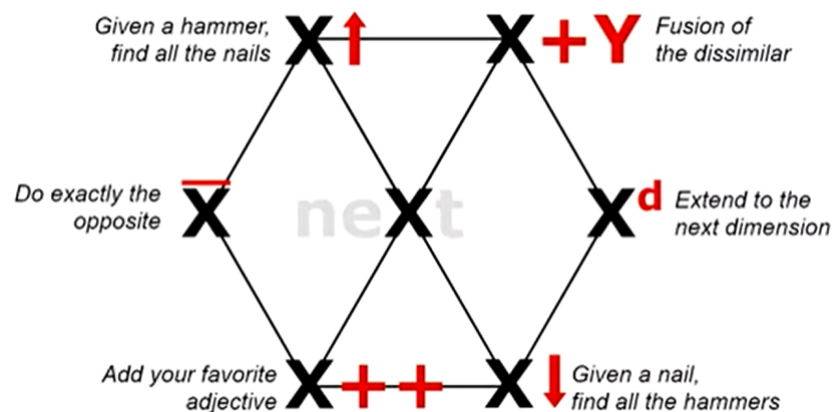
   c. Problem-Promise-Premise Approach:

   - What is the problem tackled by your XR solution? (importance)
   - What is the promise of your XR solution? (value proposition)
   - What is the premise of your XR solution? (assumption)

   d. Write the Intro First Approach:

   - Introduction: problem (in general), obstacle (specifically), solution (in essence), contribution (scientifically).

   e. Idea Hexagon

   f. Object-Control Design Matrix:

   Example:

	Slider	Voice	Pressure	Squeeze	Shake...
Snow shoes					
Gloves					
Hat			10x10 matrix		
Boots			full of rapid sketches		
Ice scraper...					

g. Threshold & Ceiling in Tool Design:
- Threshold: all the members need to learn about the tool.
- Ceiling: the most that can be done with the tool.
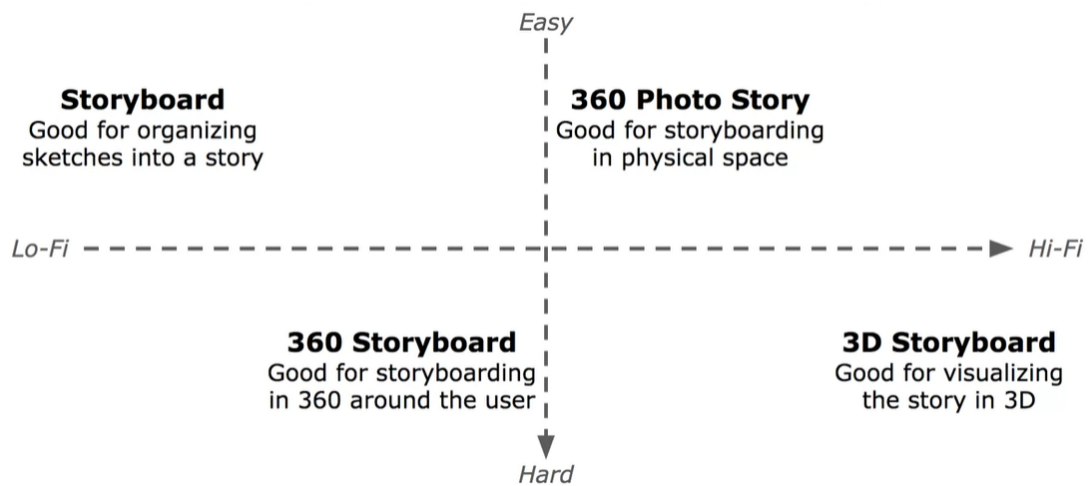
2. Good Ideas and Bad Ideas in XR Innovation:

a. Good Ideas:
- You are adding a significant new feature to an existing solution.
- You are doing a redesign of part of an existing solution.
- You are picking one particular aspect of a problem and try a different solution.
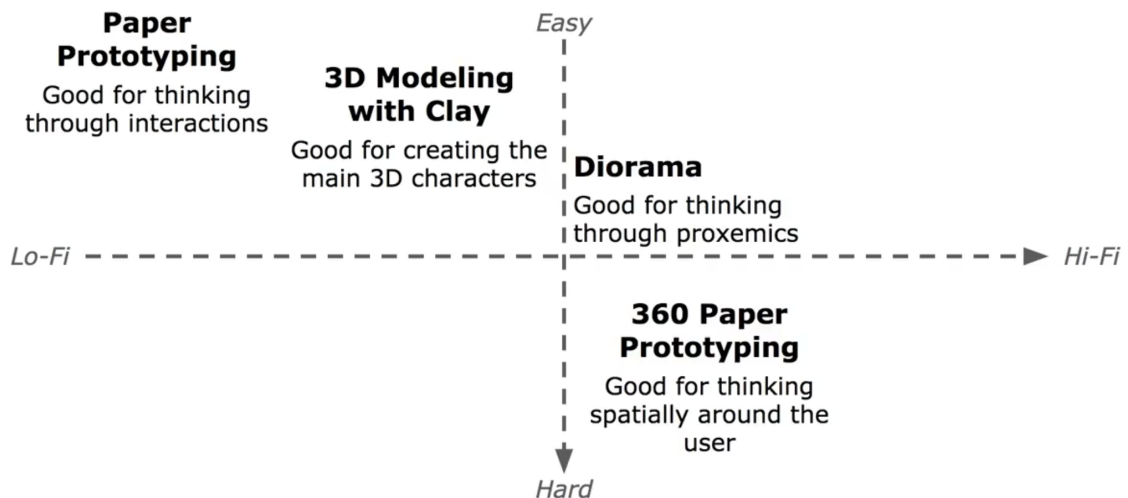- You are looking at an existing problem through a new lens.

b. Bad Ideas:
- You are just adding a small new feature to an existing solution.
- You are trying to make an existing solution "much better".
- You are trying to do what multiple existing solutions do in one new solution.
- You are trying to do what nobody else has done before.

3. Storyboarding & Wireframing:

Easy

Storyboard
Good for organizing
sketches into a story

360 Photo Story
Good for storyboarding
in physical space

Lo-Fi — — — — — — — — — — — — — — — — — — ▶ Hi-Fi

360 Storyboard
Good for storyboarding
in 360 around the user

3D Storyboard
Good for visualizing
the story in 3D

Hard

- Traditional storyboards: organize 2D screens, limited for 3D scenes.
- 360 storyboards: spatial content layout around the user, no depth.
- 3D sketches: detailed content around the user, in 3D space.

4. Physical Prototyping:

Paper Prototyping
Good for thinking
through interactions

3D Modeling with Clay
Good for creating the
main 3D characters

Easy

Diorama
Good for thinking
through proxemics

Lo-Fi — — — — — — — — — — — — — — — — — — ▶ Hi-Fi

360 Paper Prototyping
Good for thinking
spatially around the
user

Hard
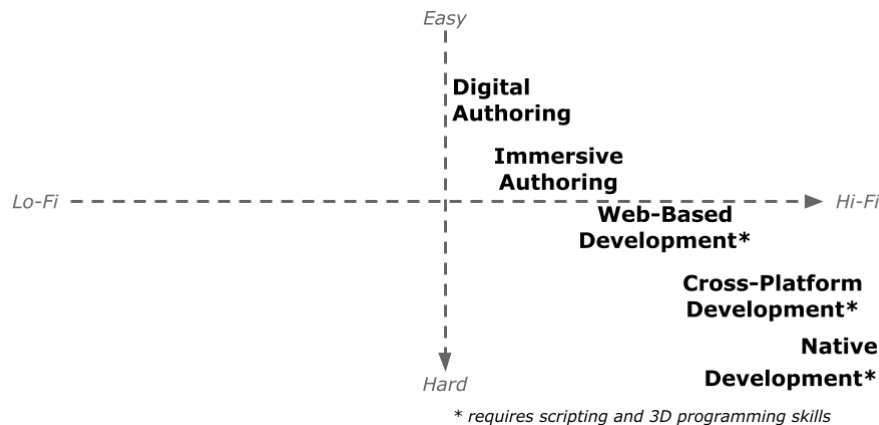
a. paper prototypes:
- bring storyboards to life.

- Demonstrate interactions in front of the user.
  - Put objects on cardboard sticks or toothpicks to animate them.

    b. 360 paper prototypes:

- Lay out content 360 degrees around the user.
- Demonstrate interactions along the gird lines.
  - Use objects of different size to simulate depth.

    c. Diorama:

- Enact the 3D scene in the physical world.
- Play out the 3D characters in response to user interactions.
  - Gauge spatial requirements and understand relationships.

    d. 3D modeling with clay:

- Bring the 3D characters to live.
- Get a scene of the complexity of 3D props.
  - Demonstrate interactions with respect to the 3D models.

Week 4: Digital Prototyping & Evaluation of XR Experiences

1. Digital prototyping:

Easy

Digital Authoring

Immersive Authoring

Lo-Fi — → Hi-Fi

Web-Based Development*

Cross-Platform Development*

Native Development*

Hard

requires scripting and 3D programming skills

    a. Digital Authoring Tools for VR: Amazon Sumerian, Unity Editor

- Support visual authoring of 3D scene graphs with VR previews.
- Basic interactions can be implemented without coding.
- Advanced interactions require JavaScript, C# or C++.

    b. Digital Authoring Tools of AR: Vuforia Studio, Lens Studio.

- Support visual authoring of marker-based or marker-less AR apps.
- Provide default marker and support for custom markers.
- Typically enable AR previews through emulator but need to deploy to AR device for testing.

    c. Immersive Authoring Tools for VR: Google Blocks, Oculus Quill

- Enable visual authoring of 3D content in VR.
- Make it possible to edit while previewing VR experience.
- Focus on 3D modeling rather than animation & scripting.
- Typically support export to common 3D model formats an asset sharing platform like Google Pole, Sketchfab or 3D Warehouse.

    d. Immersive Authoring Tools for AR: Apple Reality Composer, Adobe Aero

- Enable visual authoring of 3D content in AR.
- Make it possible to edit while previewing Ar experience in the environment.
- Provide basic support for interactive behaviors.

- Sometimes support export to WebXR.

e. Web-based Development: A-Frame

- Based on Three.js and WebGL.
- New HTML tags for 3D scenes.
- A-Frame Inspector (not editor).
- Asset management (img, video, audio, 3D models)
- ECS architecture with many open-source components.
- Cross-platform XR.

f. Cross-platform Development: Unity

- Started out as game engine.
- Has integrated support for many XR platforms.
- Powerful scene editor.
- Asset management & store.
- Basically all XR device vendors provide Unity SDKs.

2. Evaluating XR Experiences:

a. Evaluation format:

- Usability Test: efficiency, effectiveness, satisfaction.
- User Study: presence and immersions, hypothesis testing, performance benchmarking
- Playtest: gameplay, fun, user retention

b. Evaluation Criteria:

- Usability: How well can they use it?
- Usefulness: Do they find it helpful?
- Utility: Does it solve their problem?
- User Acceptance: Will they continue to use it?

c. Selected Evaluation Methods:

- System Walkthroughts

- Paper Prototype Testing:

OBSERVER "COMPUTER" FACILITATOR VR USER AR USER

Facilitator guides the process (introduces tasks, help if user gets stuck)

"Computer" simulates system dialogue (both system input and output)

Observer documents the study (what did user do, how did facilitator help)

- Heuristic Evaluation

- Usability Testing

d. Usability Evaluation:

Different types of usability evaluation:

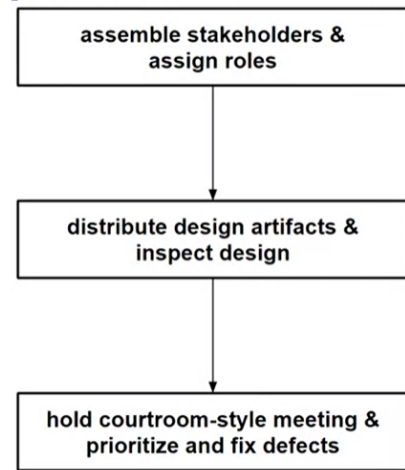- Usability Inspection:

Heuristic Evaluation
Informal, holistic
Checks against established
principles and guidelines

Cognitive Walkthrough
Procedural, task-oriented
Checks if simulated user goals
and memory can be assumed to
lead to correct action

Formal Usability Inspection
Combines above methods

- Usability Testing:

  Usability metrics: Effectiveness, Efficiency, Satisfaction, Learnability, Memorability, Predictability.

  Usability metrics specific to XR: Realism, Safety, Flexibility, Comfort.

- Automated Usability Testing:

  Custom Logging & Log Analysis, Platform-based Analytics Tools, Dedicated XR Analytics Tools.

e. Designing a User Study:
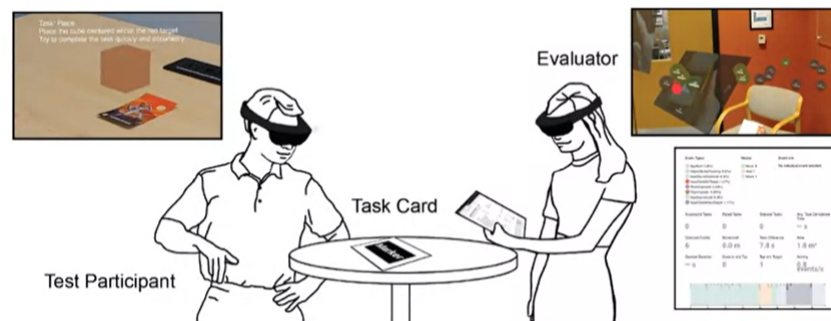
- Choose a scenario: Describe verbally to tester as part of test script, Illustrate through a combination of physical and virtual objects.
- Decide on set of tasks: Think of the usability test in terms of clearly defined testing units, Consider stopping criteria correct path, possible issues.
- Perform walkthroughs: Run pilot tests of your usability test.
- Revise study plan: But still only use minimal instructions.
- Run the usability test: Prepare forms for taking observations not es & user feedback.

f. Designing Tasks:

A good task:

- based on a goal that matters to you and the users.
- based on questions important to the success of the product or research.
- not too broad or too narrow.
- finite and predictable set of possible solutions.
- clear end point that user can recognize and reach.
- looking for actionable feedback and not just opinion.

A bad task:

- no real task.

- looks to confirm rather than test your design choice.
- too vague.
- too small/short.
- only involves trivial interactions.
- uses exact labels from interface.

g. Usability Matrix