

Communicating with Data— Prototyping

Dr. Ab Mosca (they/them)

Slides based off slides courtesy of Jordan Crouser (<https://jcrouser.github.io/>)

Coming Up



Statistical & Data Sciences

Presentation of the Majors

**Monday, November 3
Campus Center, Carroll Room
12:15-1:30pm**

**Undeclared? Come hear about the SDS
and MST Majors! All current SDS & MST
Majors/Minors are welcome.**

Lunch will be provided!

Coming Up

- Final Project!
- Instructions will be released next week on Thursday
- Yes, you do need to work with a group.
 - If you'd like help finding a group, send me a message on Slack **before next Tuesday**
- Start thinking about a topic area you're interested in

Plan for Today

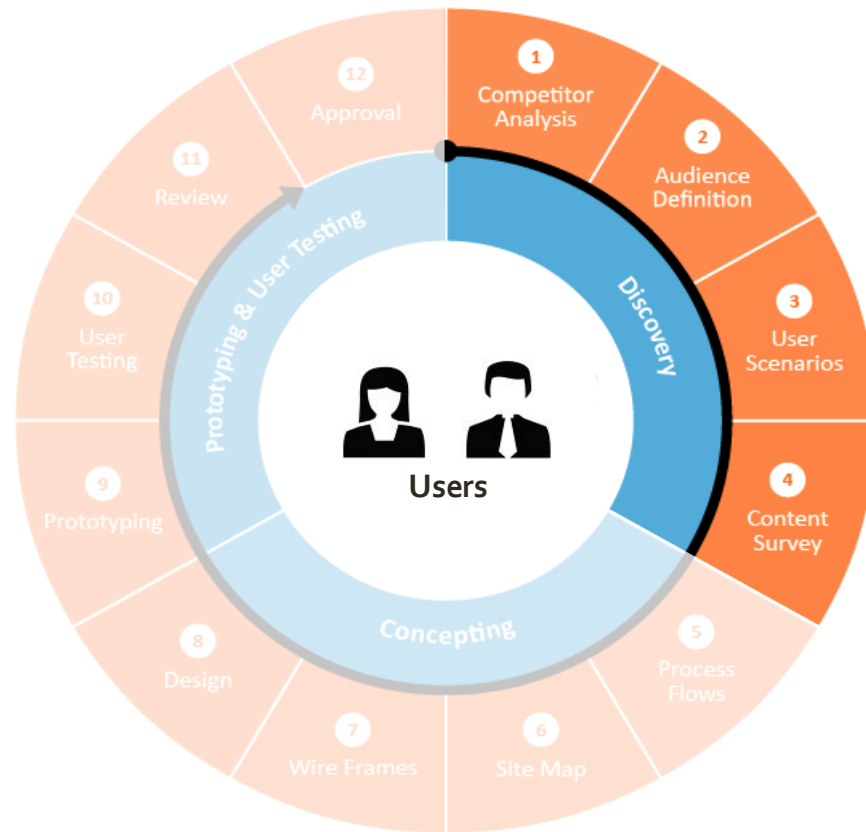
- User-centered design
 - What it is
 - Why do it
 - Ways to do it
- Paper prototypes

Hypothetical Example

Admissions asks you, a visualization expert, to **make a visualization summarizing Smith College for prospective students.**

How do you start?

User-centered design framework



1) Discovery

- Learning about your users
- Modeling your users
- Analyzing your users' tasks
- Eliciting and defining clear product requirements

2) Concepting Phase

- Developing conceptual models
- Solving design problems through ideation
- Detailed design activities

3) Prototyping + User Testing

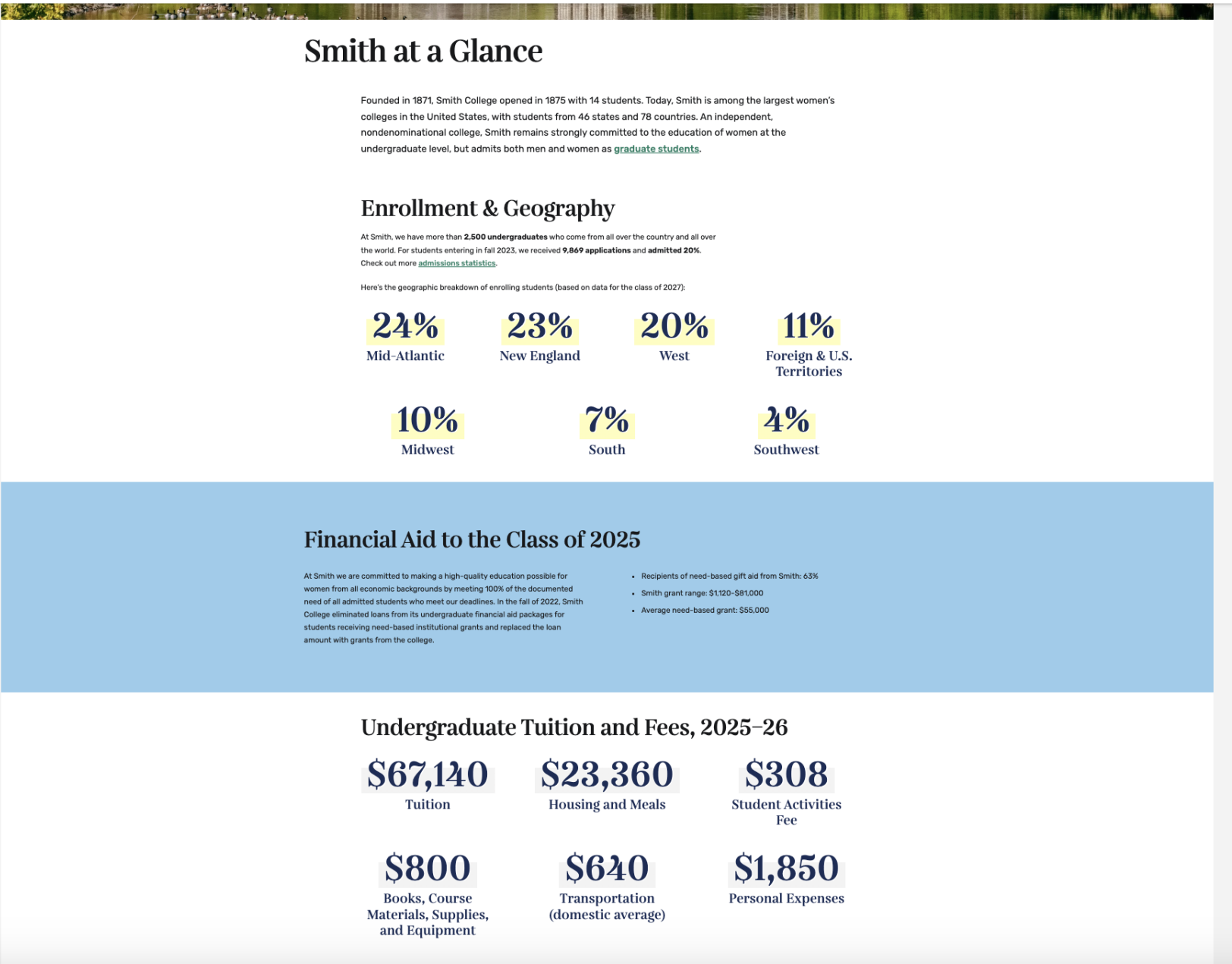
- Delivery of a high-quality product that meets users' needs and is easy to learn and use

Discovery: Competitor Analysis

- **Why?**
 - If you look at what already exists, you might be able to identify potential issues in advance
 - Also helps establish your unique contribution
- **How?**
 - Literature or product review
 - Analysis
 - What are the existing tools?
 - What is their purpose?
 - What audience are they aiming for?
 - What kinds of strategies are they using?
 - What functionality do they contain?
 - What are their strengths and shortcomings?
 - Identify opportunities and design constraints

Discovery:
Competitor
Analysis

https://www.smith.edu/discover-smith/smith-glance



Discovery: Audience Definition

- Learning about their problem
 - Semi-structured interview
- Analyzing their tasks
 - Hierarchical task analysis
- Modeling users
 - Personas

Semi-structured interviews

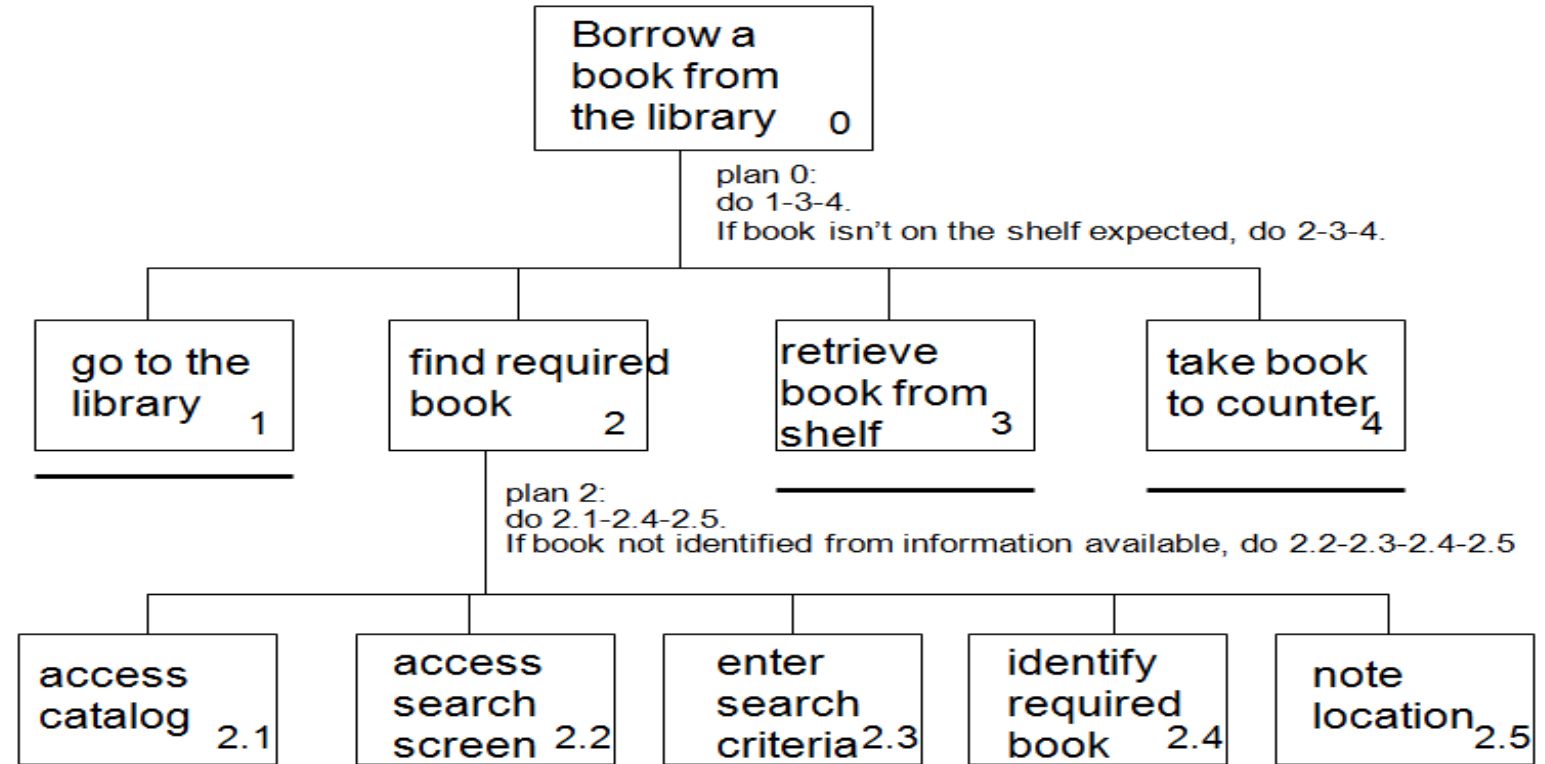
- **Why?**
 - gather qualitative data about users to understand the problem
 - can help identify key differences between designer and target user
- **How?**
 - ask open-ended questions
 - bring along a “cheat sheet” to ensure that you gather all the information you need
- **Some tips:**
 - establish trust at the beginning
 - participant engagement will vary
 - be flexible, but make sure you get what you came for
 - consider recording or note-taking to help with recall



Hierarchical task analysis

- **Why?**
 - Understand user workflow
 - Identify pain points and areas for optimization
- **How?**
 - Decompose tasks into 4-8 sequential steps
 - Identify patterns, sequences and skips in the tasks
 - An example:

Task analysis example



Personas

- **Why?**
 - mechanism for reasoning about user needs
 - model behavioral characteristics of target users
 - doesn't require access to ACTUAL users
- **How?**
 - fictionalization
 - narrative, goals, needs, "pain points"
 - attributes specific to the problem space
 - data-driven method* using info from interviews
 - mapping persona to software features

Personas

Example for a Smith College dining app

- Ellis is a first year Smithie who lives in Wilson. They go cross country and have 6am practice before their 9:25am art history class in the art museum on MWF. On TR, they have engineering 101 in Ford Hall until 12:05, and their work study job at the campus school that starts at 12:45pm sharp. Ellis is a “live to eat” kind of person, and likes to optimize the “yum” factor in their meals. In addition, they are vegan. They are comfortable using apps on their phone like GoogleMaps, but are struggling to find the time to survey dining hall menus and get to the appropriate dining hall for the meal they want between their various other activities.

Pain Point:
Limited time

Goal/Need: Eating
what they like

Pain Point:
Limited time

Pain Point: Distilling
current menu list

Technical skills: Comfortable
with basic apps

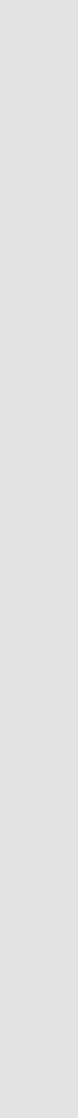

Pain Point:
Walking far & fast

Goal/Need:
Minimize walk
distance

Activity: personas

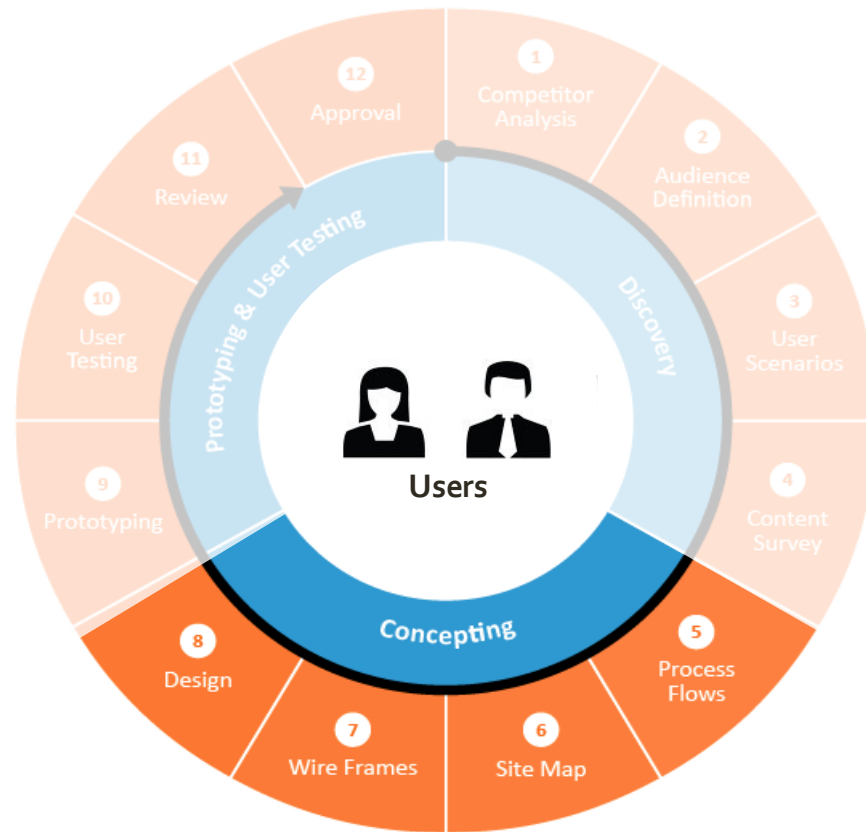
Goal: come up with a **persona** that characterizes a user of the visualization admissions asked you to create.





Now that we've got some end users in mind,
what would the system look like?

User-centered design framework



1) Discovery

- Learning about your users
- Modeling your users
- Analyzing your users' tasks
- Eliciting and defining clear product requirements

2) Concepting Phase

- Developing conceptual models
- Solving design problems through ideation
- Detailed design activities

3) Prototyping + User Testing

- Delivery of a high-quality product that meets users' needs and is easy to learn and use

Conceptual Models: Why?

- What *concepts* do your users need to be aware of to use your visualization?
- Consider:
 - What data will they interact with?
 - How will they interact with it?

Conceptual Models: Why?

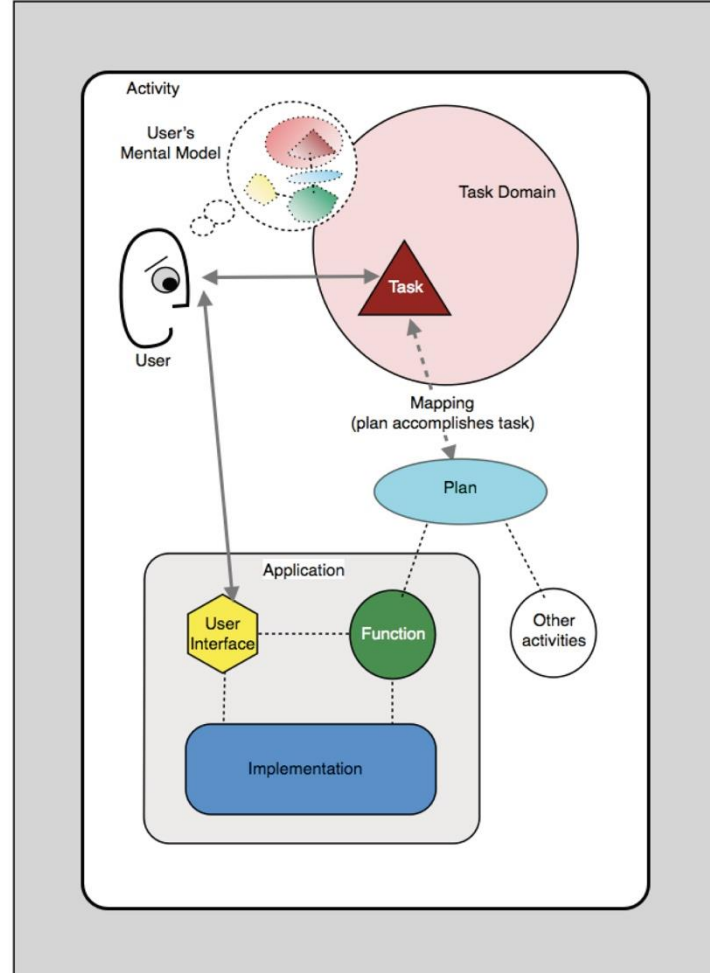


Figure 1.1: Using a tool (an application).

Johnson, J. and Henderson, A. (2011) *Conceptual Models: Core to Good Design*, Morgan & Claypool Publishers, DOI: 10.2200/S00391ED1V01Y2011HCI012

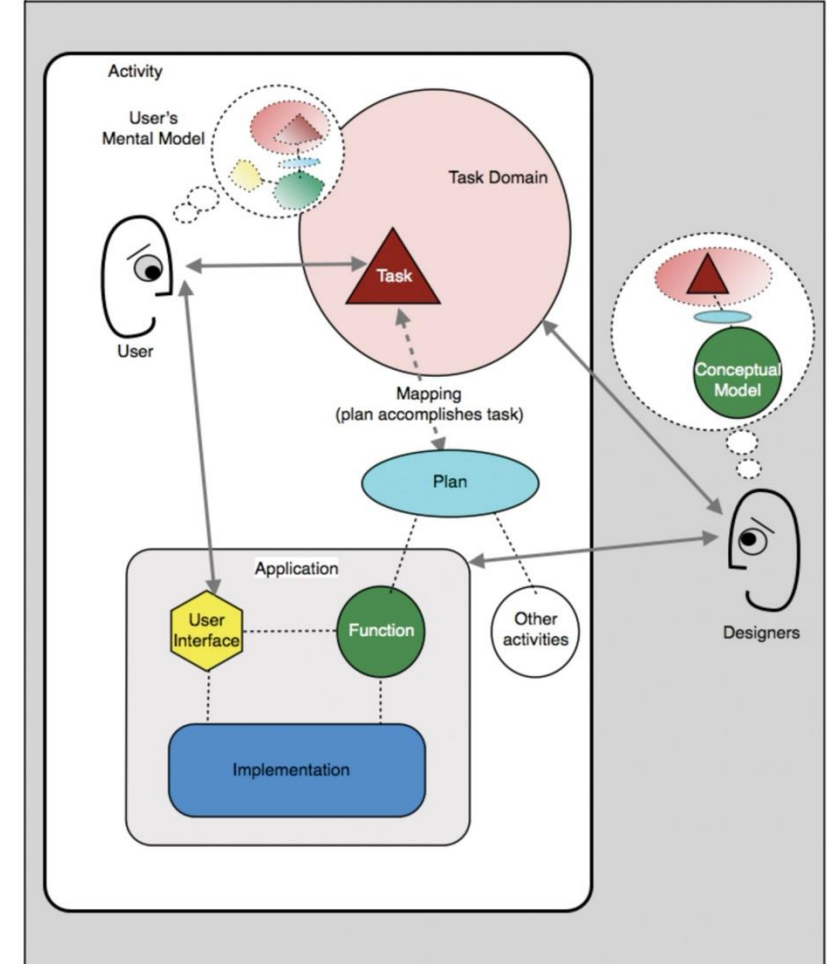
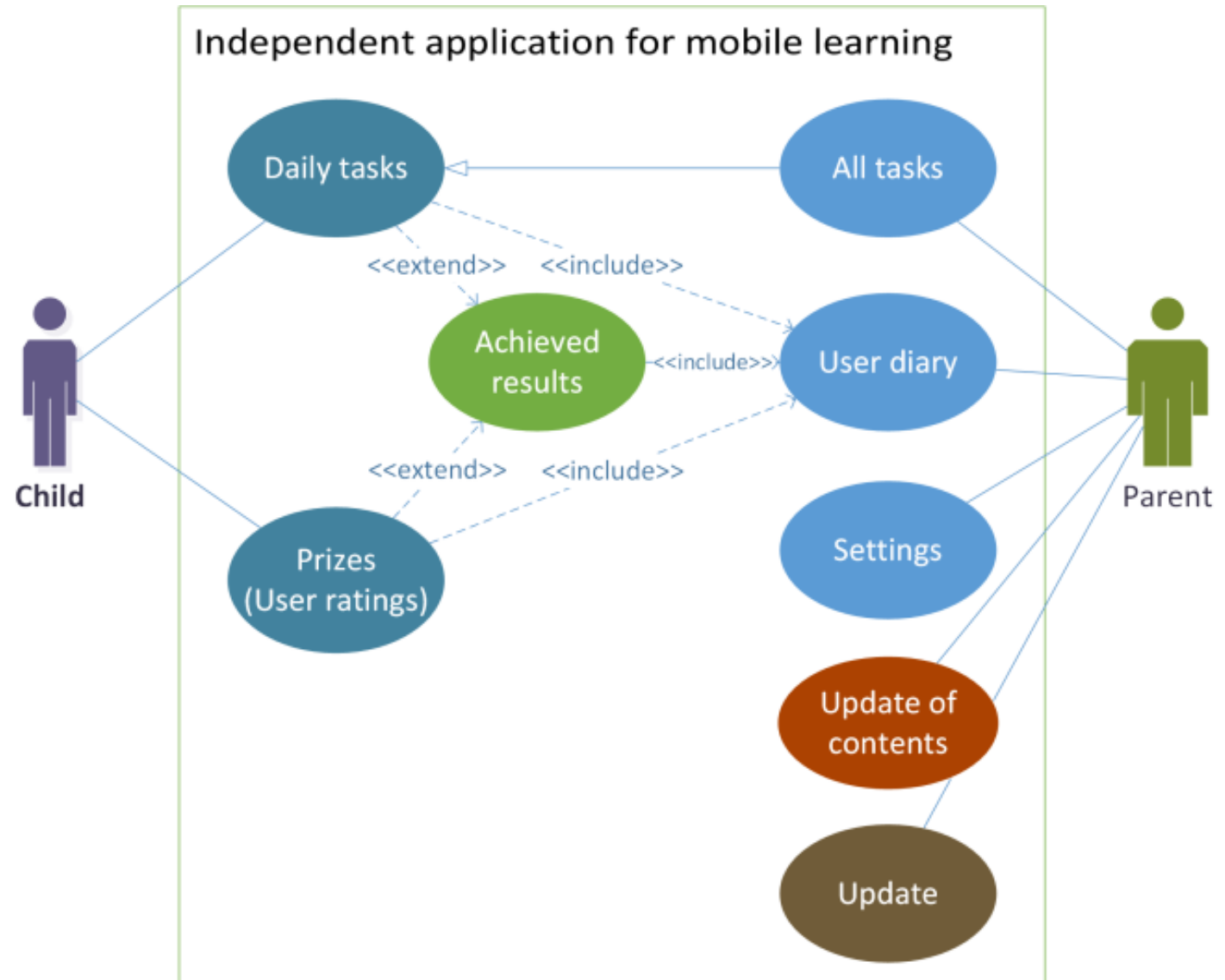


Figure 1.2: Designers' model of a user using an application.

Johnson, J. and Henderson, A. (2011) *Conceptual Models: Core to Good Design*, Morgan & Claypool Publishers, DOI: 10.2200/S00391ED1V01Y2011HCI012

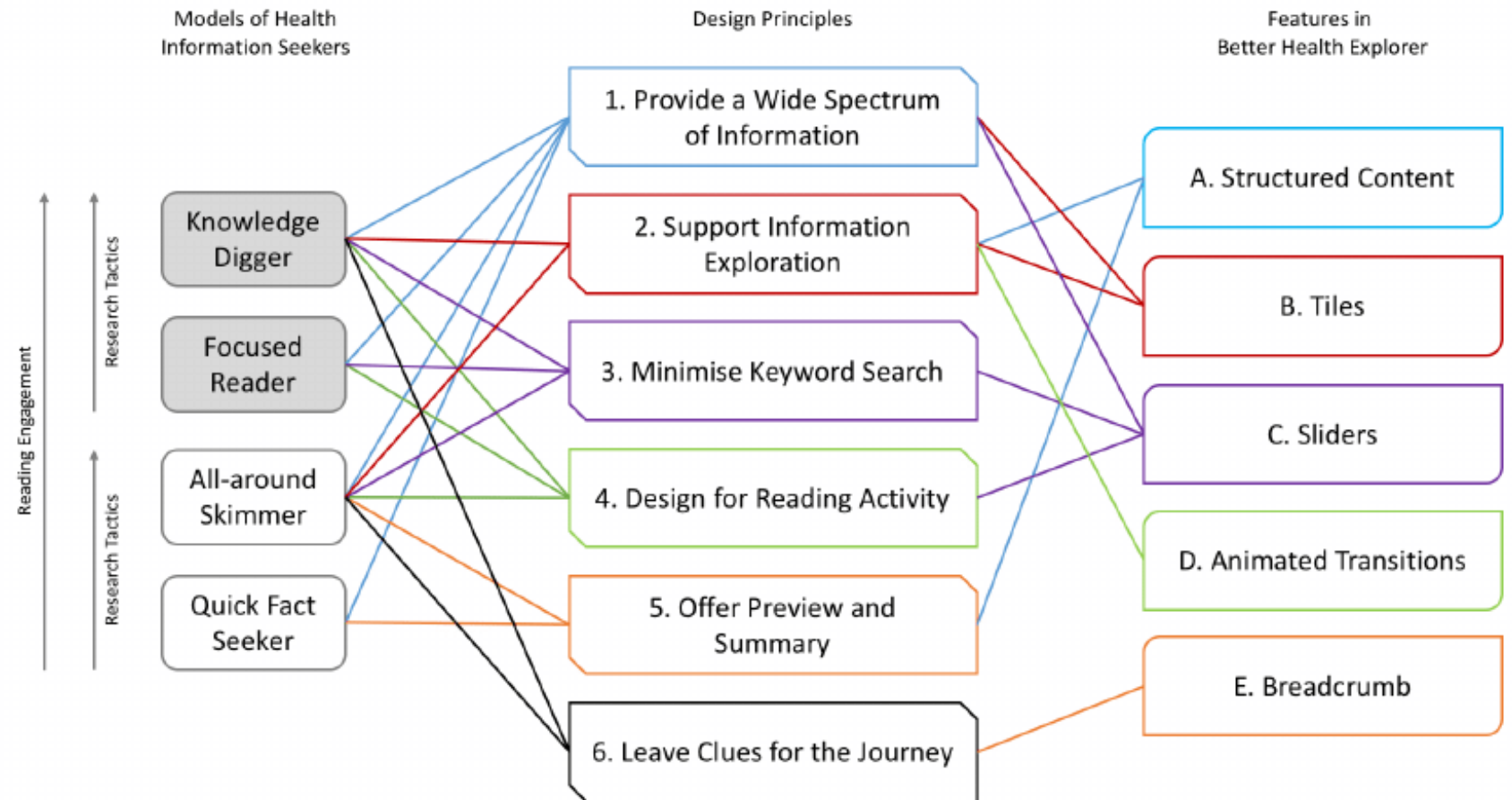
Conceptual Models: How?

- Focus on key components and content areas
- Describe all essential tasks for your visualization



Conceptual Models: How?

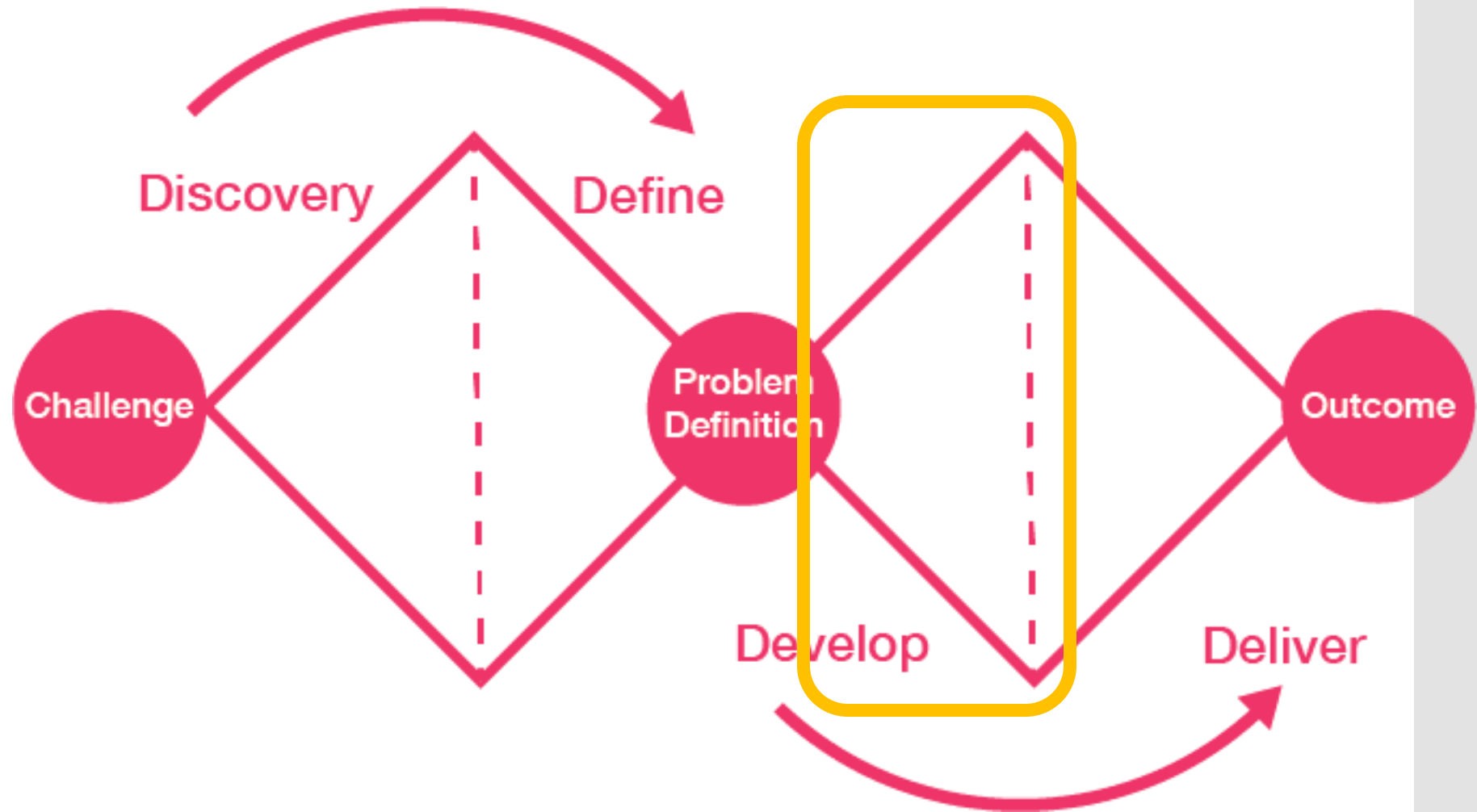
- Focus on key components and content areas
- Describe all essential tasks for your visualization



Activity: Conceptual Models:

- Draw a conceptual model for the visualization admissions asked you to make

Ideation and Detailed Design



Sketching

- What will it look like?

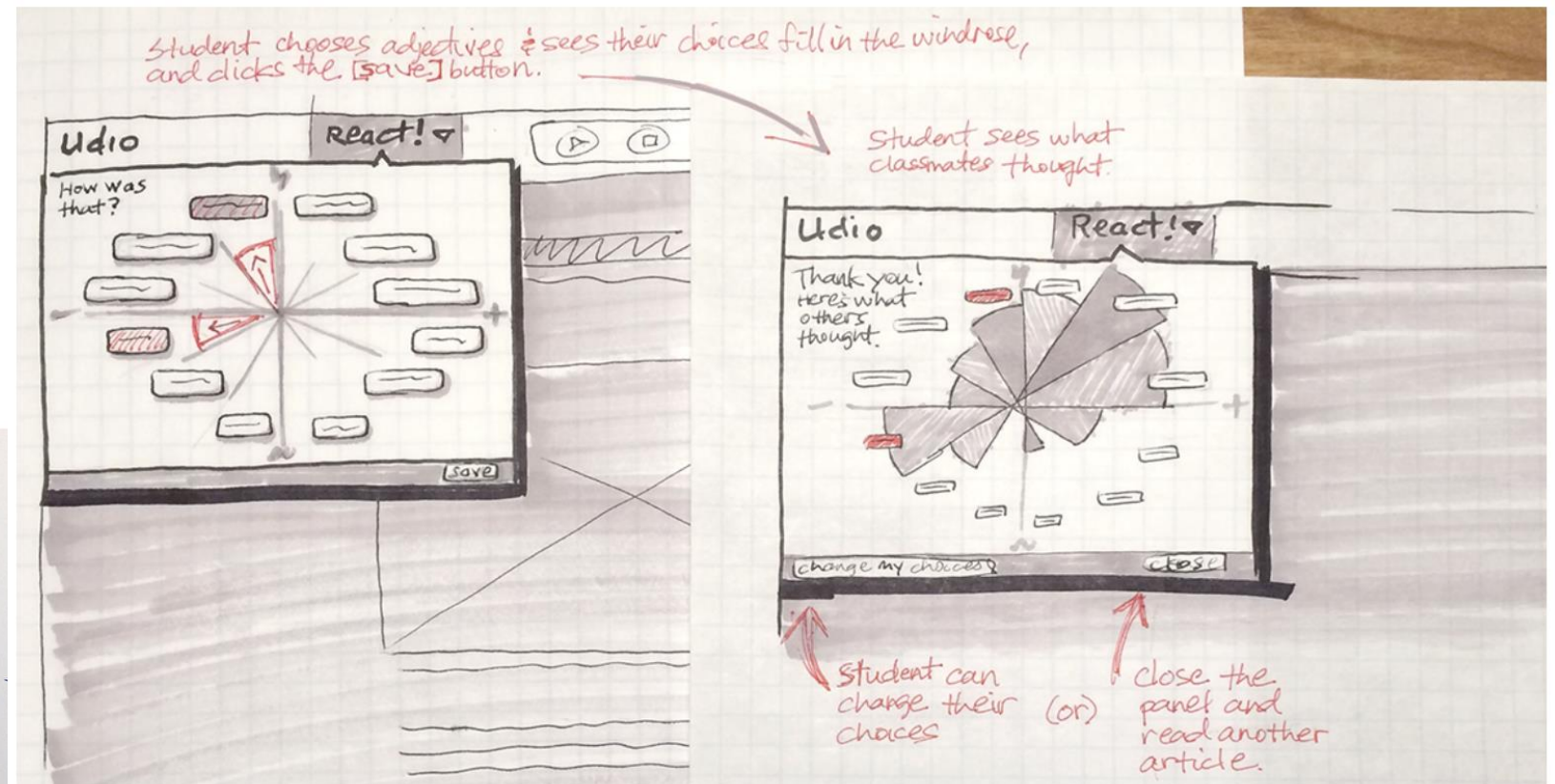
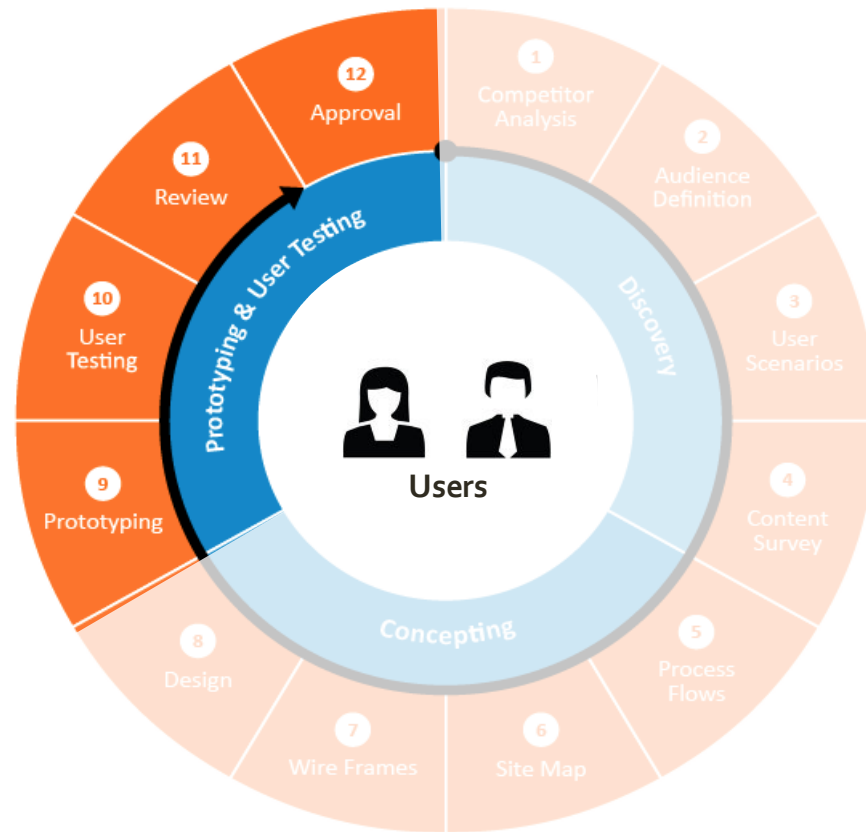


Figure 3. Example of refined sketch of “React” data visualization.

Sketching

- Roughly sketch the visualization you would make for admissions
- Pair up with 2 other teams and compare rough sketches
- Based on feedback / input draw a more polished sketch

User-centered design framework



1) Discovery

- Learning about your users
- Modeling your users
- Analyzing your users' tasks
- Eliciting and defining clear product requirements

2) Concepting Phase

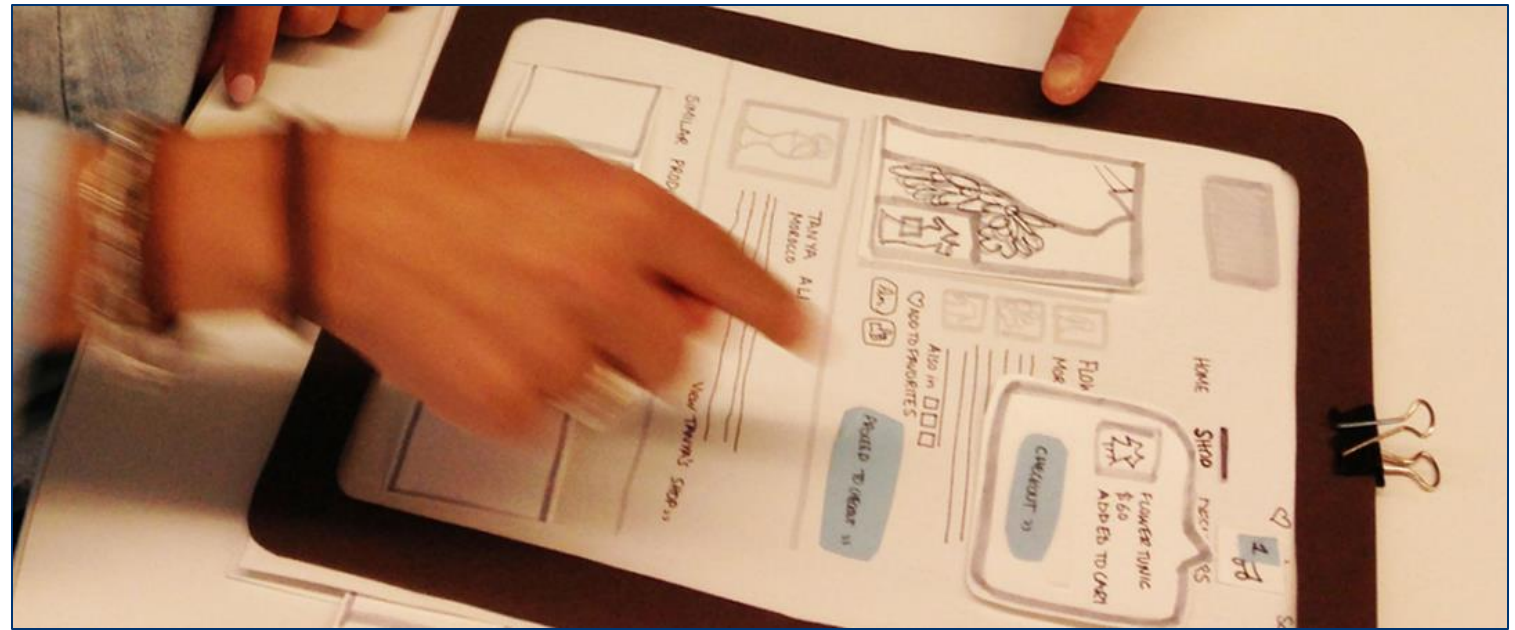
- Developing conceptual models
- Solving design problems through ideation
- Detailed design activities

3) Prototyping + User Testing

- Delivery of a high-quality product that meets users' needs and is easy to learn and use

Prototyping and Testing: Low-fidelity paper prototyping

- **Big idea:**
 - Not sure yet whether or not an **idea** will work?
 - Making a **paper version** of an interface is a lot faster and easier than coding a working prototype – start there!



Prototyping and Testing: Low-fidelity paper prototyping

- Generate **lots of ideas**
- Engage **other people** in the design process
- Identify **potential problems** before you waste time coding
- Get **feedback** quickly, from lots of different people
- Some tips:
 - Focus on the **big picture**, don't worry about the details
 - **Think about what you want it to do**, rather than what you know how to implement (we'll worry about that later)
 - Not so into arts and crafts? It doesn't have to be **actual paper**... Whiteboard / PowerPoint / Keynote will also do the trick!

Prototyping and Testing: Low-fidelity paper prototyping

Examples:

- <https://www.youtube.com/watch?v=nAgQP9lkl2o>
- <https://www.youtube.com/watch?v=y2oE3qBmHpg>
- <https://www.youtube.com/watch?v=yafaGNFu8Eg>

Prototyping and Testing: Soliciting feedback

- One purpose of a prototype is to get feedback on your design idea. We do this through **user testing**:
 - Choose specific tasks your end user should be able to do with your app
 - Ask someone to perform those tasks with your prototype
 - Do not give clues or help while they perform each task
 - Ask the tester to “think out loud” i.e. narrate what they are doing / why
 - Observe where they get stuck and what they like
 - Modify your design based on testing
- What tasks could we ask a user to perform to test the admissions visualization?

Prototyping and Testing: Medium & High Fidelity

- **Medium-fidelity prototype**
 - Typically made once design is more solidified
 - Usually does not involve coding
 - Could be made with:
 - Powerpoint
 - Canva
 - Figma
 - Etc..
 - Same idea -- build, test, adapt
- **High-fidelity prototype**
 - Once you have a solid idea of the design, code your app
 - Test and adapt
 - You might implement one feature at a time

Your turn!



- Build a paper prototype for your admissions visualization
- Choose 3 tasks to have another group perform with your prototype
 - Pay attention to where they run into trouble

Takeaways

- Think about your end user early → you're more likely to **build something that actually solves the problem**
- “**Low-fidelity**” **prototyping** saves time and energy by helping identify problems before you commit to code
- Also, the process is **kinda fun...**