

SWE 632 - Design & Development of User Interfaces



George Mason
University

Instructor:
Dr. Kevin Moran

Teaching Assistant:
David Gonzalez Samudio

Class will start in:
20:00

SWE 632 - Design & Development of User Interfaces



George Mason
University

Instructor:
Dr. Kevin Moran

Teaching Assistant:
David Gonzalez Samudio

Class will start in:
20:00

SWE 632 - Design & Development of User Interfaces

Fall 2020



George Mason
University

Dr. Kevin Moran

Week 5:
Sketching &
Prototyping





Administrivia

- Project Checkpoint 3 due Next Week
- In-class Midterm Exam in Two Weeks



Expectations for Midterm Exam

- Free response, essay questions
- Will include definitions, key ideas & concepts, how to use methods
 - May link multiple ideas together in applying them to a scenario
- Lectures, assigned readings, tech talks
- Sample questions:
 - What's one advantage of using React over Flutter? What's one advantage of Flutter?
 - Given UI image w/ description, conduct a heuristic evaluation to identify at least 3 issues
 - Define an empirical usability evaluation with an example



Project Checkpoint 3

In this HW assignment, you will improve the interaction design of your web app by making changes to fix at least 9 potential usability issues that have been identified with your app.

1. Each project group will be provided a document listing at least 15 potential usability issues identified with your app. If your document is missing usability issues or you do not understand any of the issues, please contact the TA.
2. You should provide separate URLs for (1) the original HW1 version of your web app and (2) the new, updated version of your web app addressing the usability issues.
3. You should select 9 of the reported potential usability issues to address. For each issue, (1) copy the text and original screenshot(s) describing the issue from the provided document, (2) describe in a short paragraph how the reported issue has been addressed, (3) include a new screenshot(s) depicting the new behavior of your web app
4. If two (or more) of the usability issues that were reported are similar or identical in nature, you can count the fix that you make multiple times for each of the reported usability issues it addresses.
5. In grading your assignment, we will evaluate the effectiveness and thoroughness of each change in addressing the reported usability issue.
6. All of your submitted documents should include your name and the names of the other group members (if applicable).



Class Overview

- *Part 1 -Sketching & Storyboards:* Working through & linking ideas
- *Part 2 - Wireframes & Design Critques:* Contextualizing ideas to a UI
- *Part 3 - Prototyping:* Building (some) of the ideas
- *Part 4 -In Class Activity:* Sketching an Example
- *7 Minute Break*
- *Part 5: React Tech Talk:* Joseph, Weifeng & Han
- *Part 6: Flutter Tech Talk:* Obeyd & Eliane

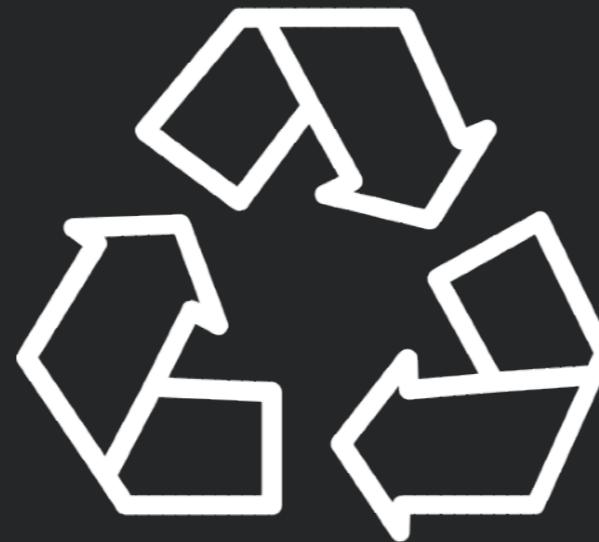
Iterative Model of User-Centered Design

Observation

(Re)Define the Problem
Understand User Needs

Idea Generation

Brainstorm
what to build



Test

Evaluate what
you have built

Prototype

Build

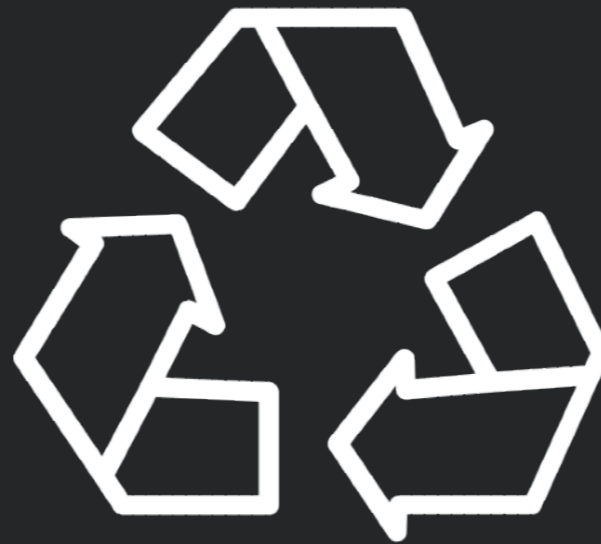
Iterative Model of User-Centered Design

Observation

(Re)Define the Problem
Understand User Needs

Test

Evaluate what
you have built



Idea Generation

Brainstorm
what to build

Prototype

Build

Sketching & Storyboards



How do You Brainstorm?



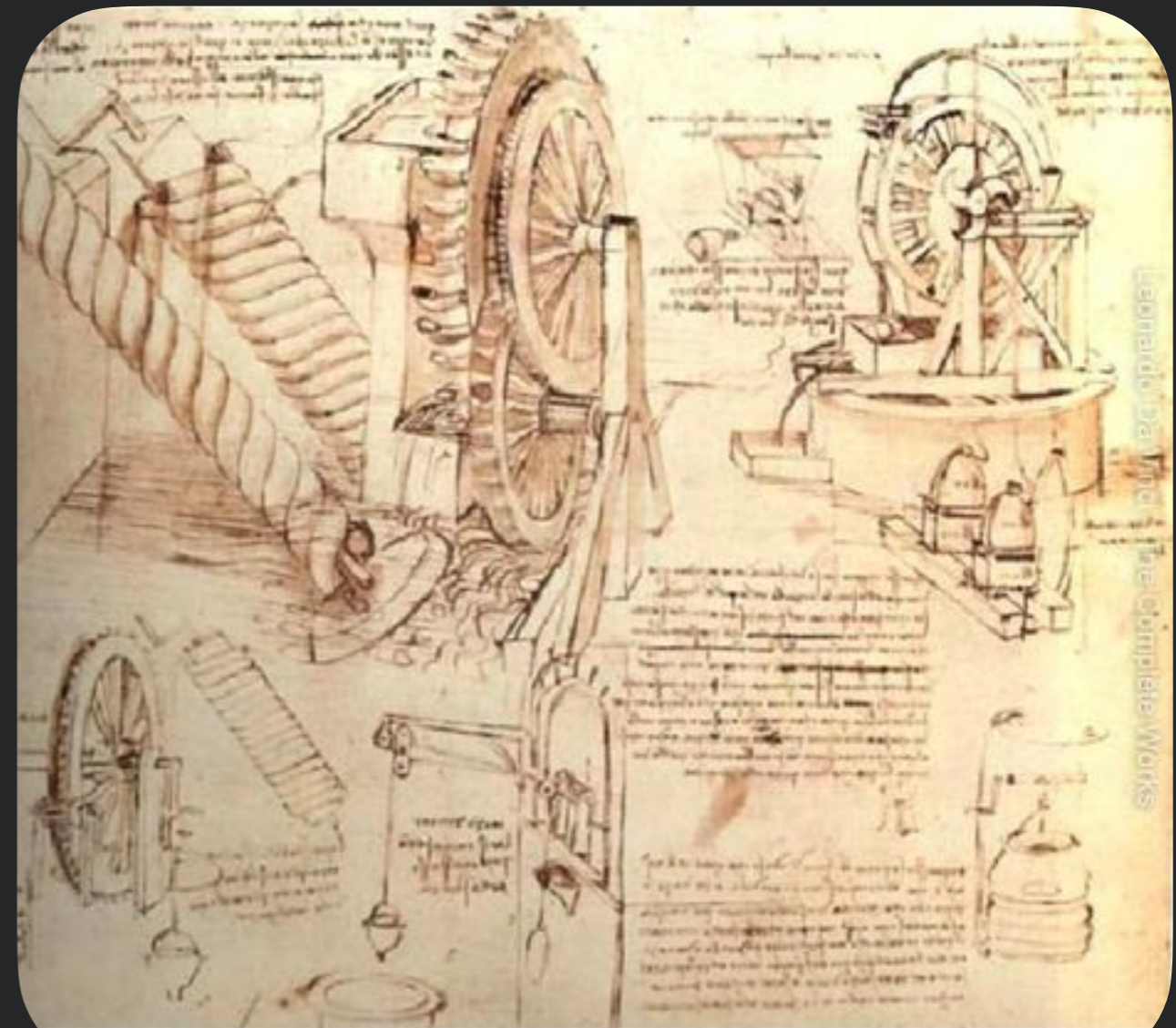


What is a Sketch?

“A conversation between the sketcher or designer and the artifact”

Why Sketch?

- Sketching offers visual medium for exploration, offering cognitive scaffolding to externalize cognition



courtesy of www.leonardoda-vinci.org

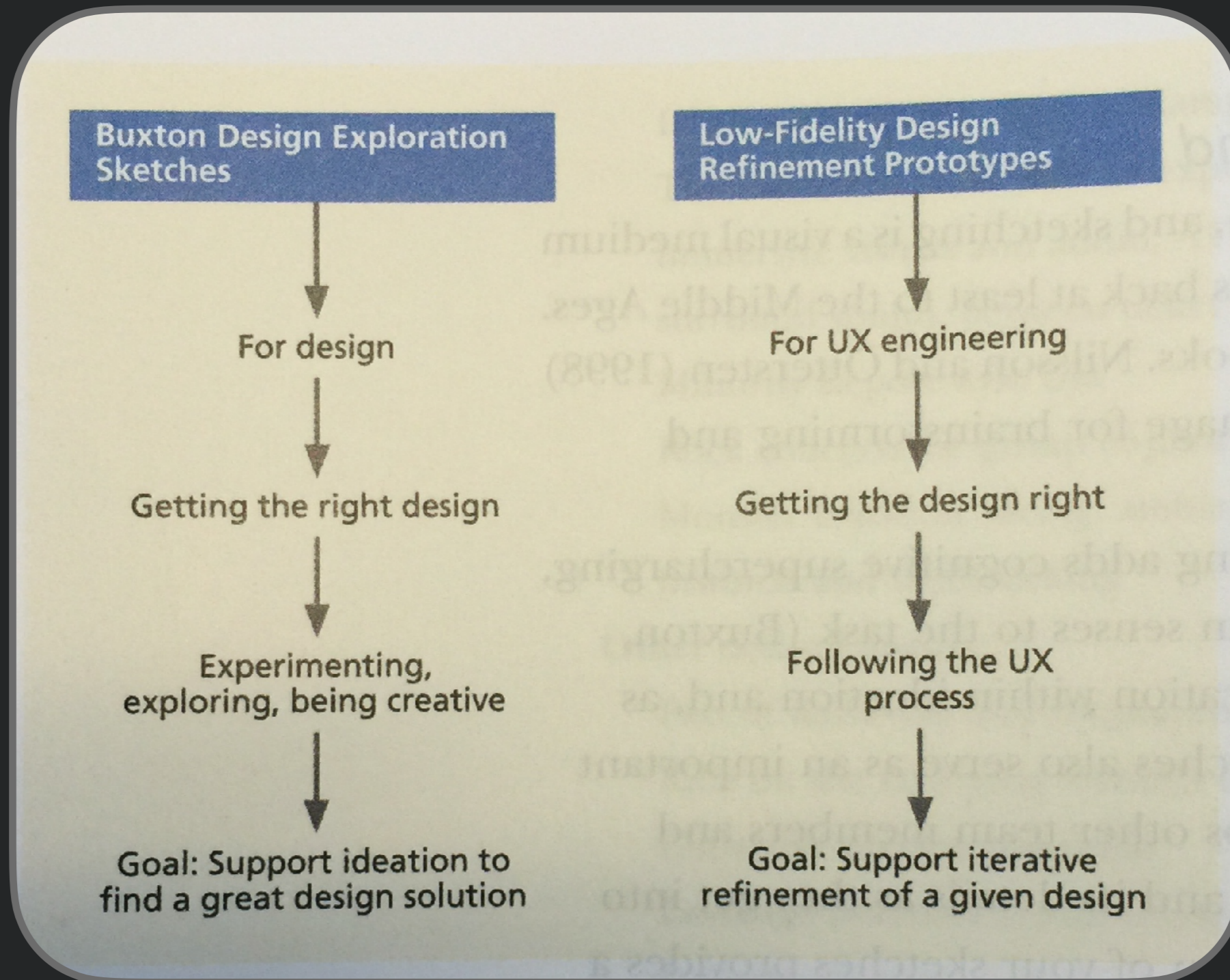
Leonardo da Vinci: The Complete Works



Being Creative with Sketches

- How do you come up with a great idea?
 - Generate lots of ideas
 - Work through ideas through externalization in sketch
 - Critique the ideas
 - Refine them to make them better
- Sketching offers a low-cost medium for working with early ideas **before** committing to one
- Design is process of creation & **exploration**

Sketching vs. Prototyping





Physical Sketches

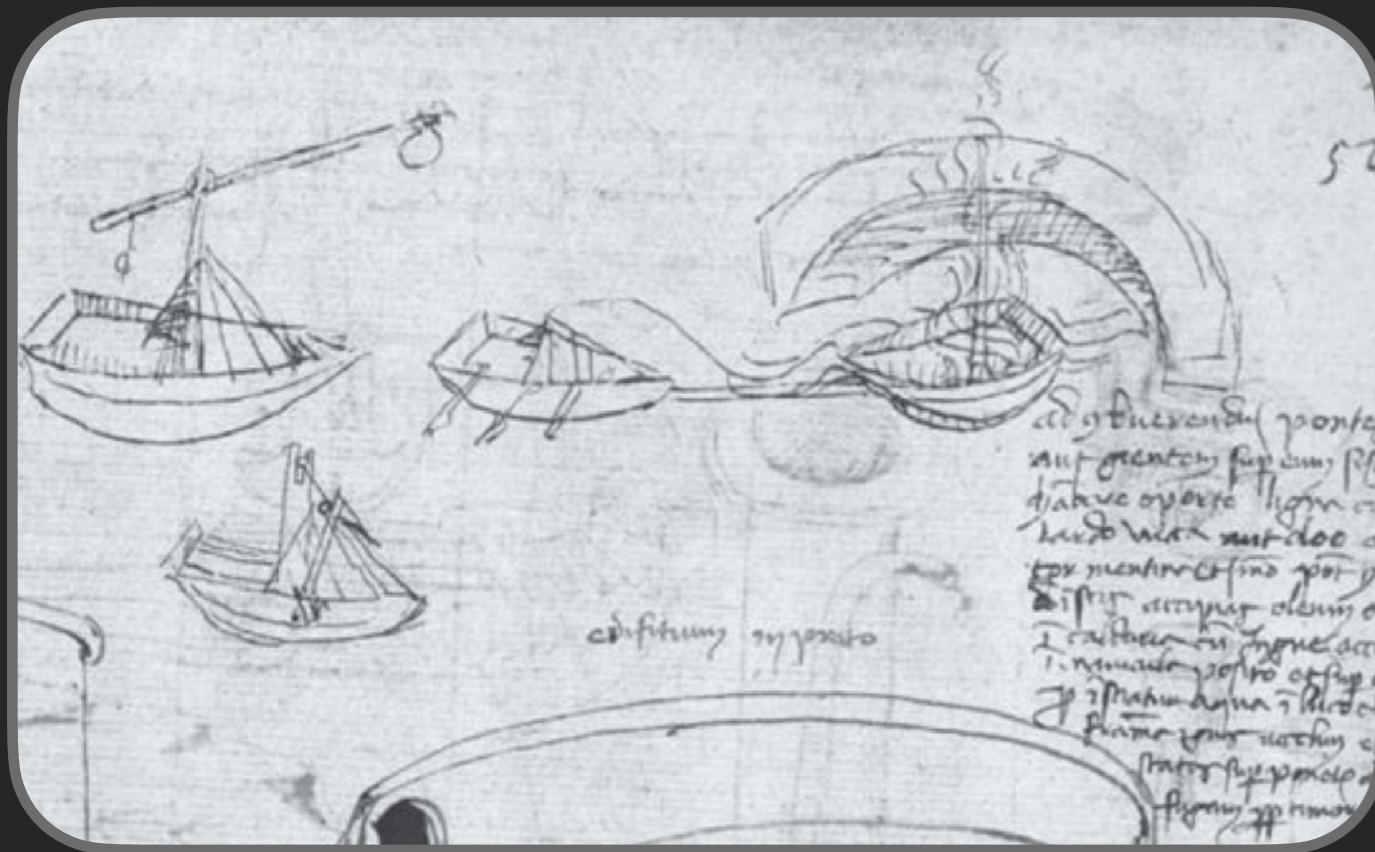
- Production tools for sketching:
 - whiteboards, blackboards, cork boards, flip chart easels
 - post it notes
 - duct tape, scotch tape, push pins, staples
 - marking pens, crayons, spray paint
 - scissors, hobby knives, foam core board
 - duct tape
 - bits of cloth, rubber

The Space Remembers

- Covering walls, whiteboards, etc. w/ materials is extremely useful
- Provides fast access for revisiting and remixing old ideas
- Facilitates group discussion of designs



Sketches are Sketchy



- Not mechanically correct and perfectly straight lines
- **Freehand**, open gestures
- Strokes may miss connections
- Resolution & detail **low** enough to suggest is concept
- Deliberately **ambiguous** & abstract, leaving “holes” for imagination



Rules for Sketching

- **Everyone** can sketch; you do not have to be artistic
- Most ideas conveyed more effectively with sketch than words.
- Sketches are **quick** and inexpensive to create; do not inhibit early exploration
- Sketches are **disposable**; no investment in sketch itself
- Sketches are **timely**; made in-the-moment, just-in-time
- Sketches are **plentiful**; entertain large # of ideas w/ multiple sketches of each

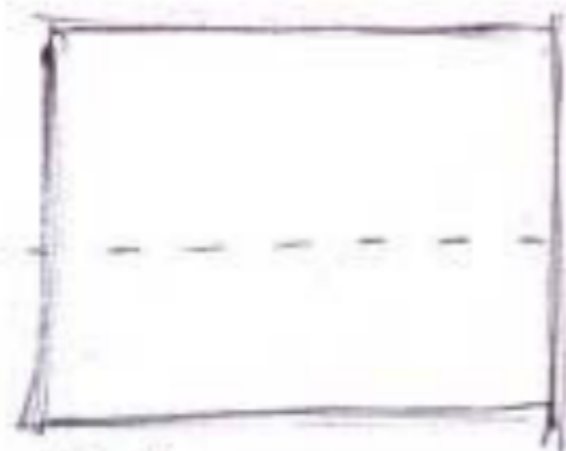
Sketches Include Annotations

- Annotations explain what is going on in each part of sketch & how

Revisiting the helium project



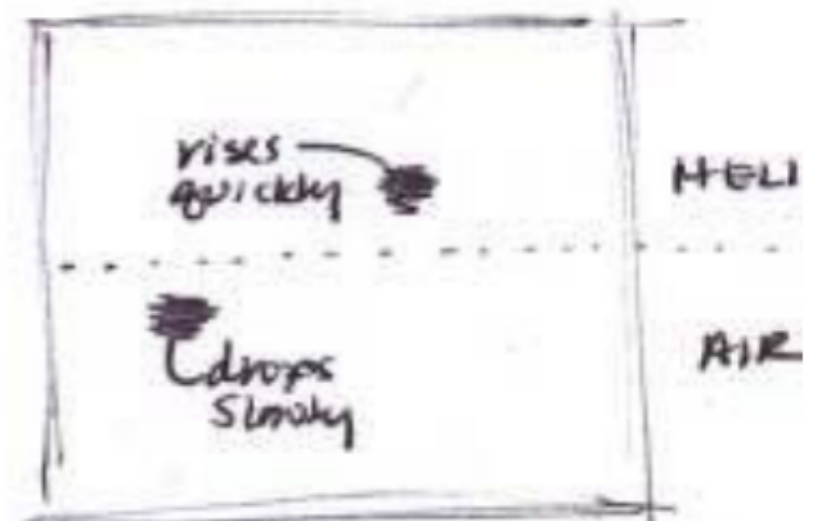
CURSOR AREA
FADES IN



If the cursor moves
above the line or
"up" it (the cursor)
changes to helium.
If it moves down
it changes to air.
Speed is matched

CAN THE
SPLIT BE
TOP AND
BOTTOM?

OR



Single image used.
Black rectangle appears
when entering the
opposite area? or
blurred cursor circle
just behaves differently
in one versus the other.

Sketches part of design exploration



NOVICE → INTERESTED → ADVANCED → EXPERTS
 May stop anywhere on this line, which is fine!
 Go through this

object: Physical interactions: Mouse, keyboard, touch, laptop...
 Physical Software interactions: What things are on screen, where things are, States...

LEARNING THE BASICS
 Navigation: Right/left click, Backwards, forwards, opening, closing, saving, undoing.
 REGIONS: Toolbar, toolbar, Taskbar. *This is a TASKBAR! I'm not a novice!*

WAYS TO TEACH THEM STUFF.
 LEARN AS YOU GO
 LEARN BY EXAMPLE
 HOW DO USERS GET CONFIDENT? Confidence meter.
 How do you ask someone "Is this your first time using a pc?" without asking anything?
 What about OEMs overriding everything...? *pointing*
 If you need to know one thing it's this... PSST... *(shades of the office assistant)*
 THANKS USERS ARE WORRIED ABOUT. SHOW ME

Is there any way of establishing a user experience?
 Ask them → Amazing → Try and guess → unpredictable

- Do you need help with a concept?
 - Do you need help from a friend? → Network of friends. New user support group

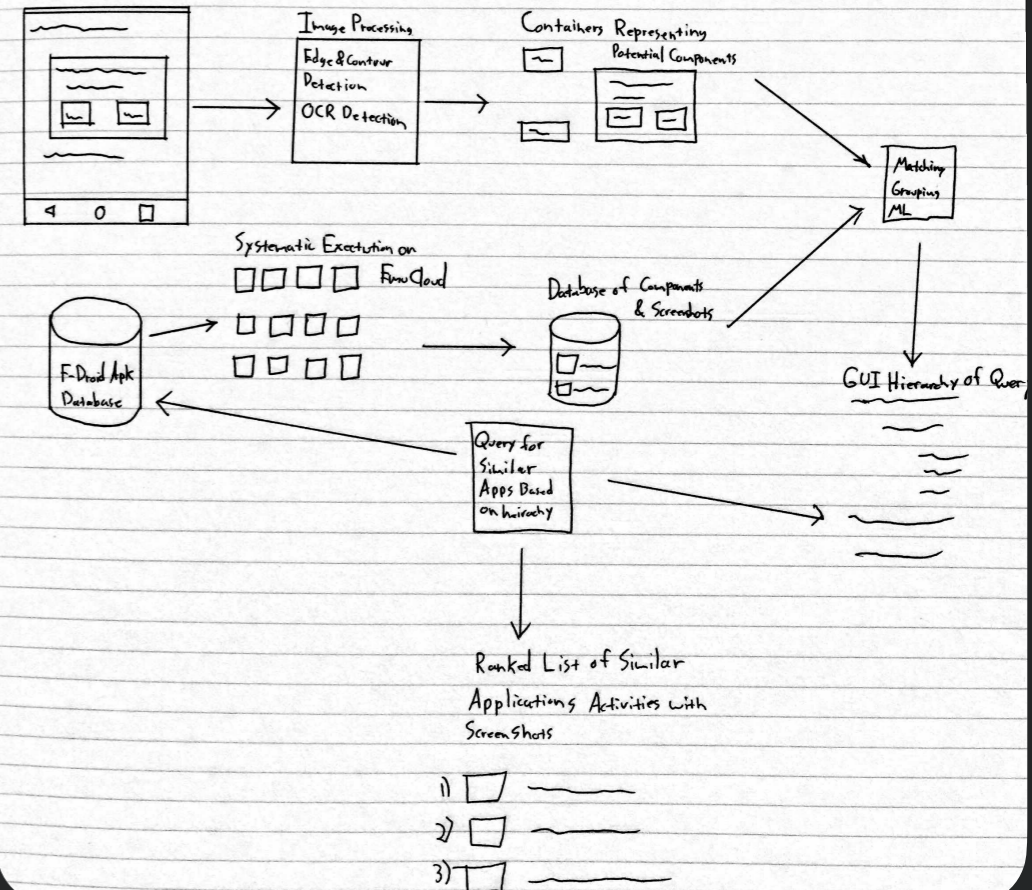
Not knowing the basics
 Not knowing how to set something up → Not online ::: problem. Ignoring warnings

Problem 1: figuring out the expertise of someone.
 Problem 2: knowing what they need help with.
 Problem 3: Building a UI that grows as they go.

B. Buxton. Sketching User Experiences.

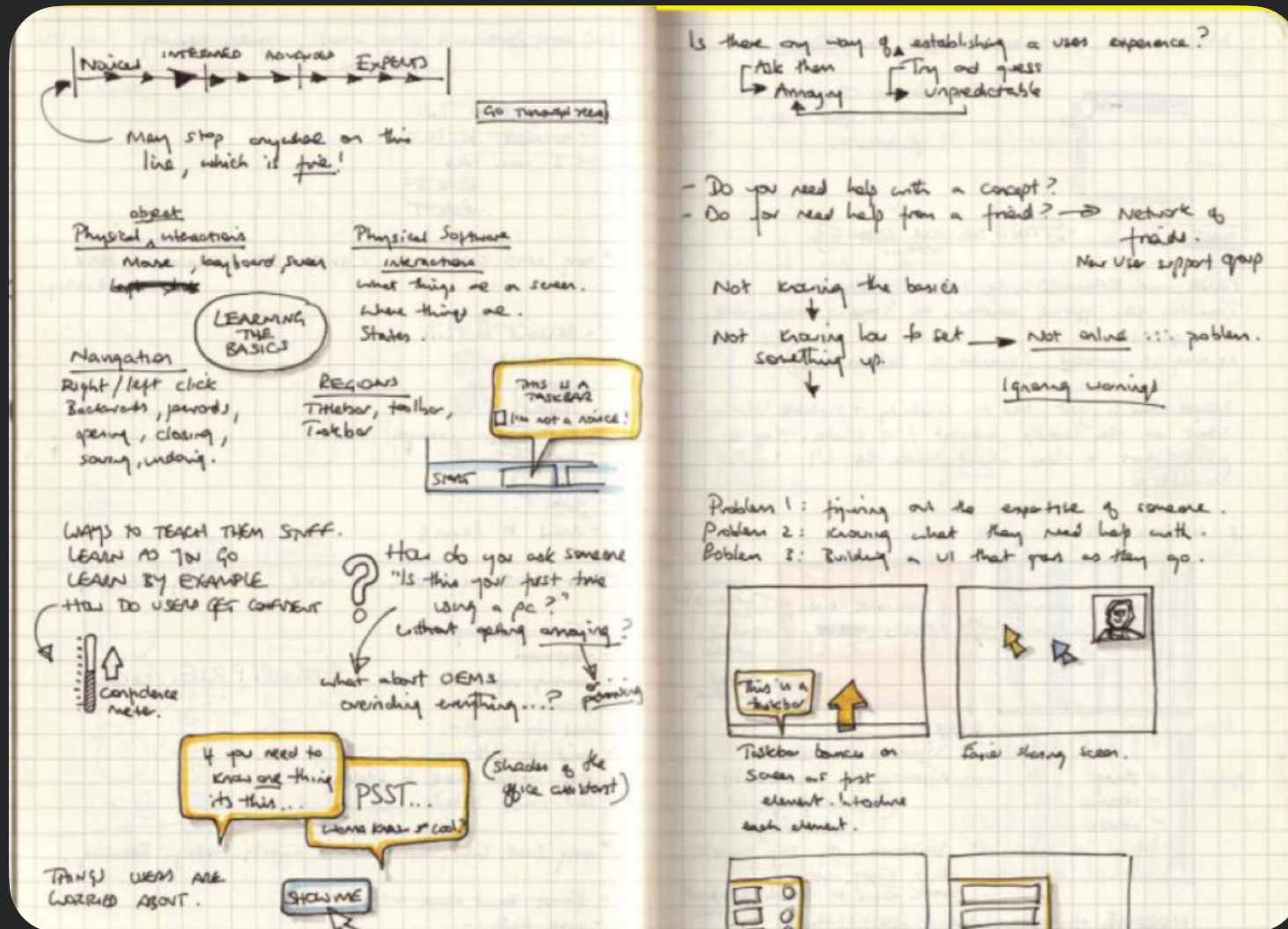
GUI Based Code Search

Screenshot/Sketch Query



K. Moran, ReDraw Project Sketch

Sketches part of design exploration



B. Buxton. Sketching User Experiences.

Machine Learning-Based Prototyping of Graphical User Interfaces for Mobile Apps

Kevin Moran, Member, IEEE, Carlos Bernal-Cárdenas, Student Member, IEEE, Michael Curcio, Student Member, IEEE, Richard Bonett, Student Member, IEEE, and Denys Poshvanyk, Member, IEEE

Abstract—It is common practice for developers of user-facing software to transform a mock-up of a graphical user interface (GUI) into code. This process takes place both at an application's inception and in an evolutionary context as GUI changes keep pace with evolving features. Unfortunately, this practice is challenging and time-consuming. In this paper, we present an approach that automates this process by enabling accurate prototyping of GUIs via three tasks: *detection*, *classification*, and *assembly*. First, logical components of a GUI are *detected* from a mock-up artifact using either computer vision techniques or mock-up metadata. Then, software repository mining, automated dynamic analysis, and deep convolutional neural networks are utilized to accurately *classify* GUI-components into domain-specific types (e.g., toggle-button). Finally, a data-driven, K-nearest-neighbors algorithm generates a suitable hierarchical GUI structure from which a prototype application can be automatically *assembled*. We implemented this approach for Android in a system called ReDraw. Our evaluation illustrates that ReDraw achieves an average GUI-component classification accuracy of 91% and assembles prototype applications that closely mirror target mock-ups in terms of visual affinity while exhibiting reasonable code structure. Interviews with industrial practitioners illustrate ReDraw's potential to improve real development workflows.

Index Terms—GUI, CNN, Mobile, Prototyping, Machine-Learning, Mining Software Repositories.

1 INTRODUCTION

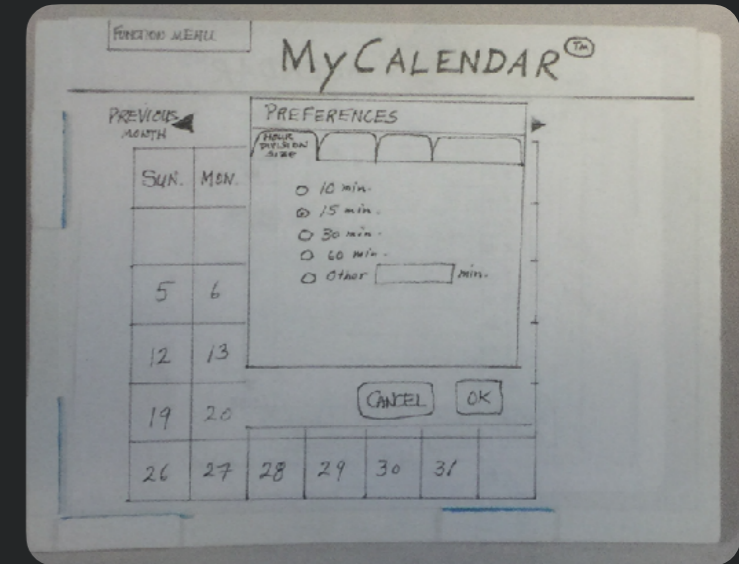
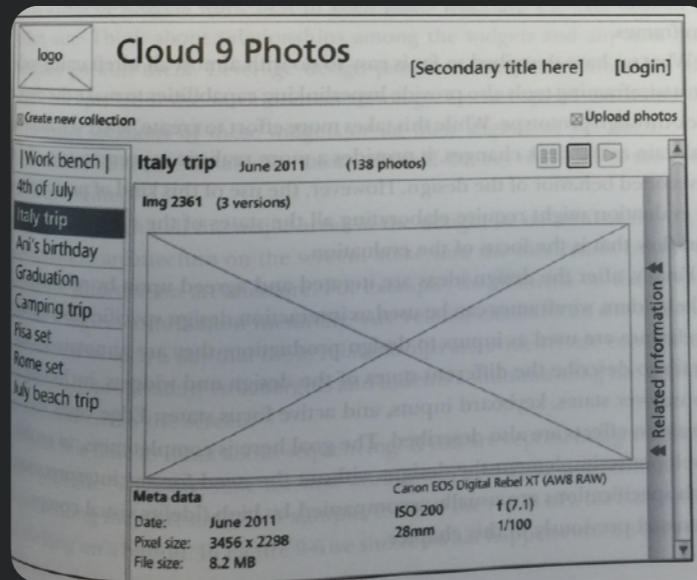
MOST modern user-facing software applications are GUI-centric, and rely on attractive user interfaces (UI) and intuitive user experiences (UX) to attract customers, facilitate the effective completion of computing tasks, and engage users. Software with cumbersome or aesthetically displeasing UIs are far less likely to succeed, particularly as companies look to differentiate their applications from competitors with similar functionality. This phenomena can be readily observed in mobile application marketplaces such as the App Store [1], or Google Play [2], where many competing applications (also known as *apps*) offering similar functionality (e.g., task managers, weather apps) largely distinguish themselves via UI/UX [3]. Thus, an important step in developing any GUI-based application is drafting and prototyping design mock-ups, which facilitates the in-

committing to spending development resources implementing them. After these initial design drafts are created it is critical that they are faithfully translated into code in order for the end-user to experience the design and user interface in its intended form.

This process (which often involves multiple iterations) has been shown by past work and empirical studies to be challenging, time-consuming, and error prone [6], [7], [8], [9], [10] particularly if the design and implementation are carried out by different teams (which is often the case in industrial settings [10]). Additionally, UI/UX teams often practice an iterative design process, where feedback is collected regarding the effectiveness of GUIs at early stages. Using prototypes would be preferred, as more detailed feedback could be collected; however, with current practices

K. Moran, ReDraw Project Sketch

Fidelity of Sketches & Mockups



Storyboard ————— Wireframe ————— Prototype

low

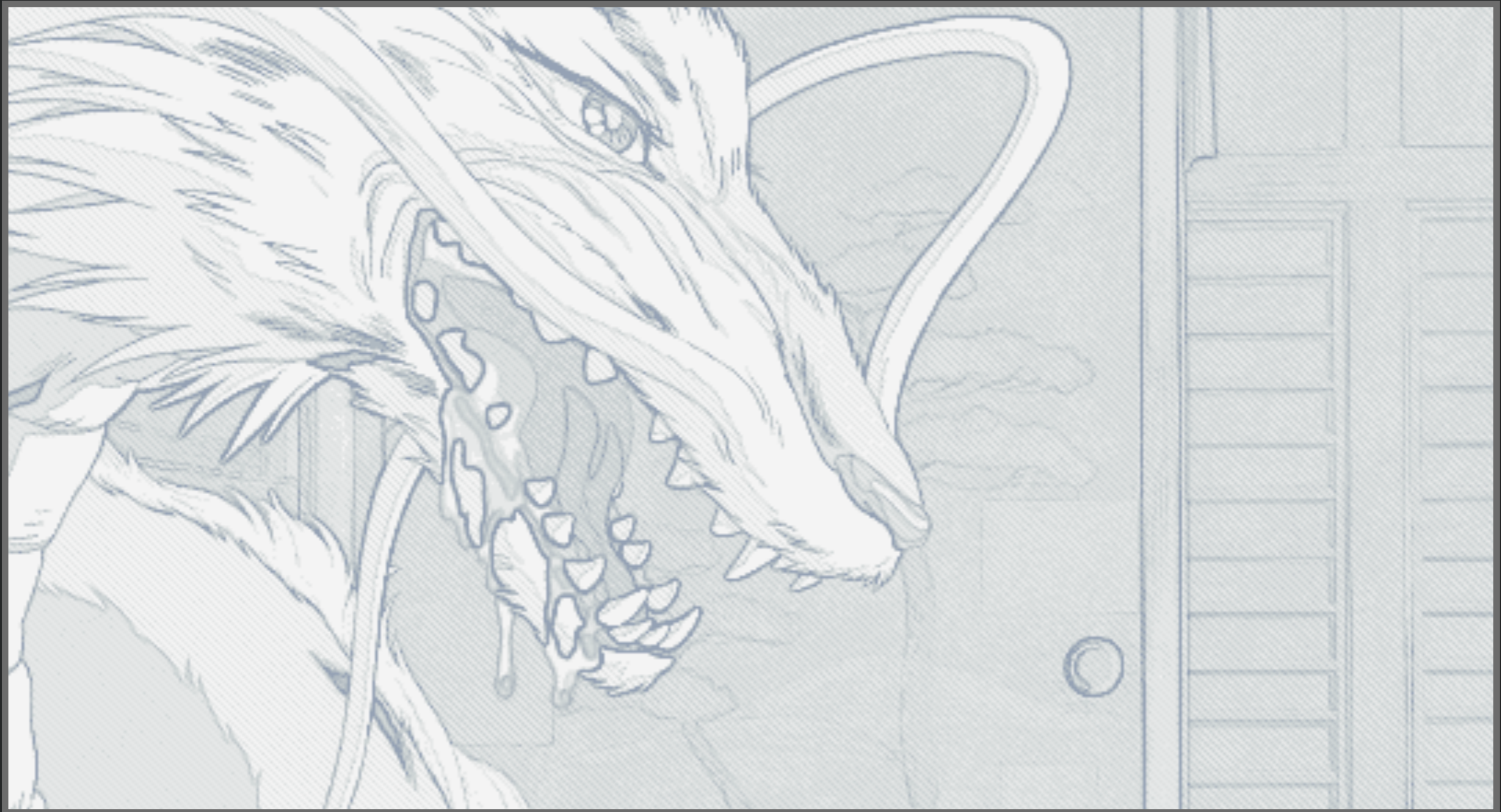
(many details left unspecified)

Fidelity

high

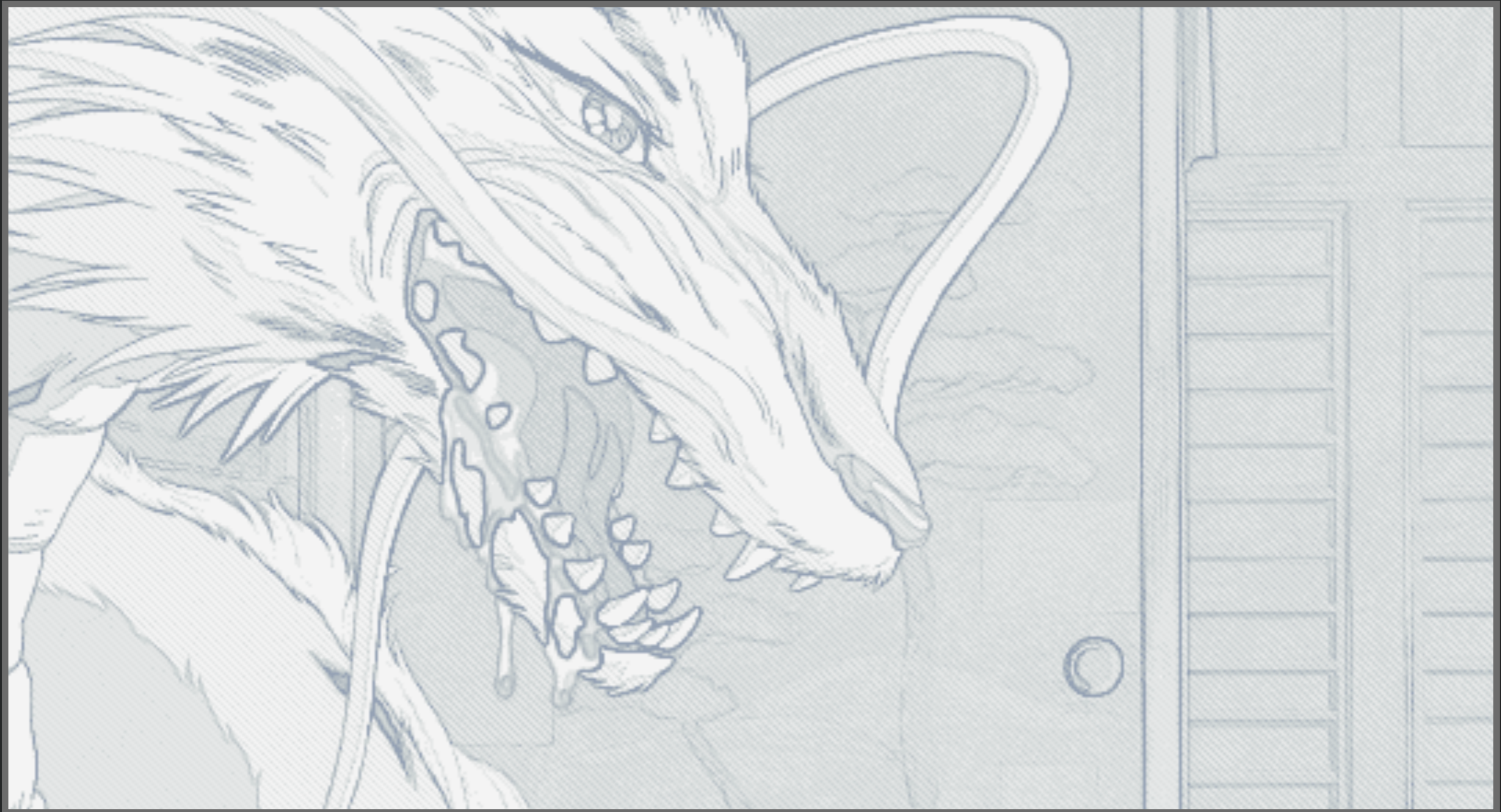
(more polished & detailed)

Classic Storyboards



Credit Studio Ghibli: "Spirited Away"

Classic Storyboards



Credit Studio Ghibli: "Spirited Away"

Storyboards for UI Design

- Sequence of visual “frames” illustrating *interplay* between user & envisioned system
- Explains how app fits into a larger *context* through a single scenario / story
- Bring design to *life* in graphical clips - freeze frame sketches of user interactions
- “Comic-book” style *illustration* of a scenario, with actors, screens, interaction, & dialog



Crafting a Storyboard

- Set the stage:
 - Who? What Where? Why? When?
- Show key interactions with application
- Show consequences of taking actions
- May also think about errors

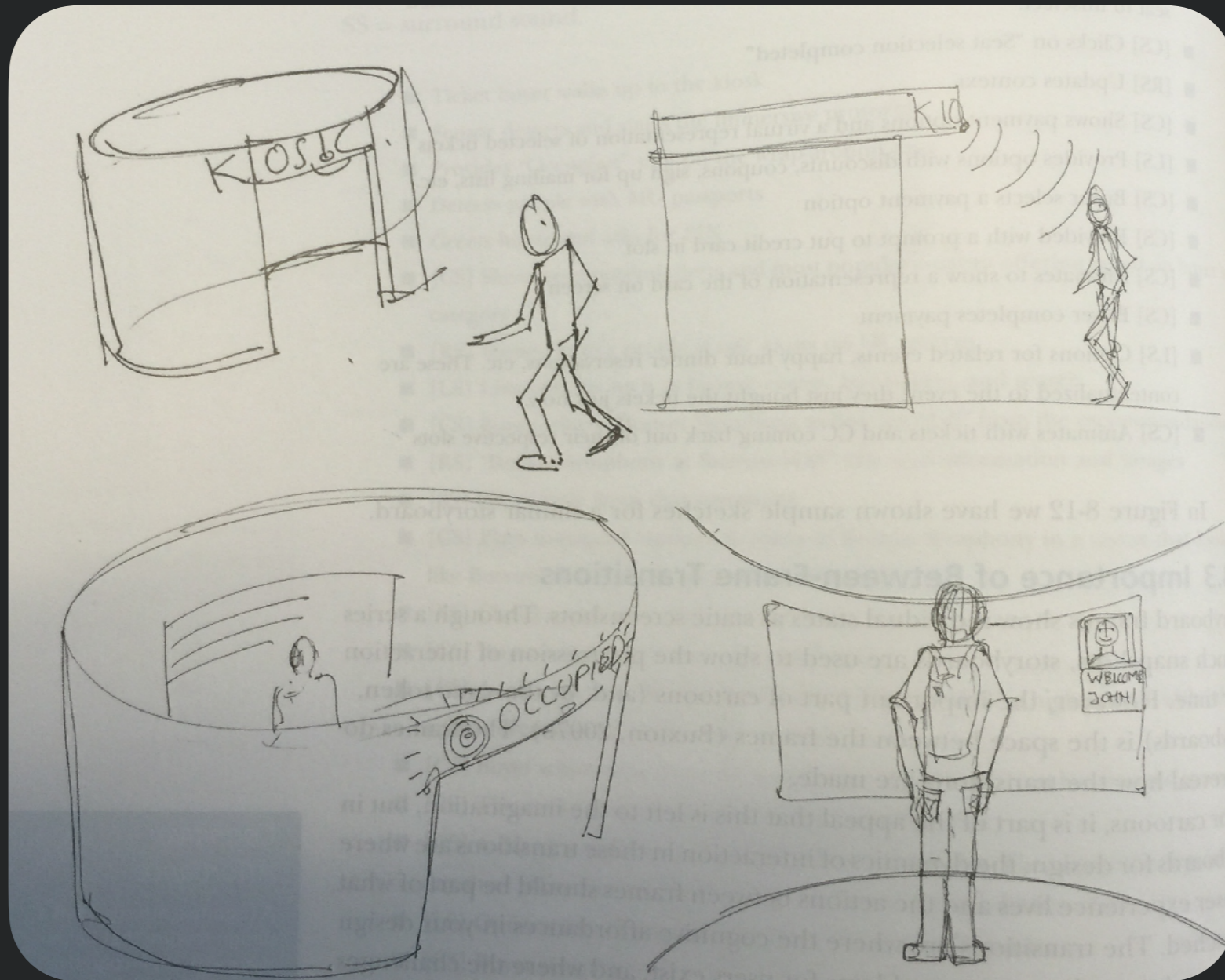


Example Elements of a UI Storyboard

- Hand-sketched pictures annotated with a few words
- Sketch of user activity before or after interacting w/ system
- Sketches of devices & screens
- Connections with system (e.g., database connection)
- Physical user actions
- Cognitive user action in “thought balloons”

Example: Ticket Kiosk

Ticket buyer walks up to the kiosk



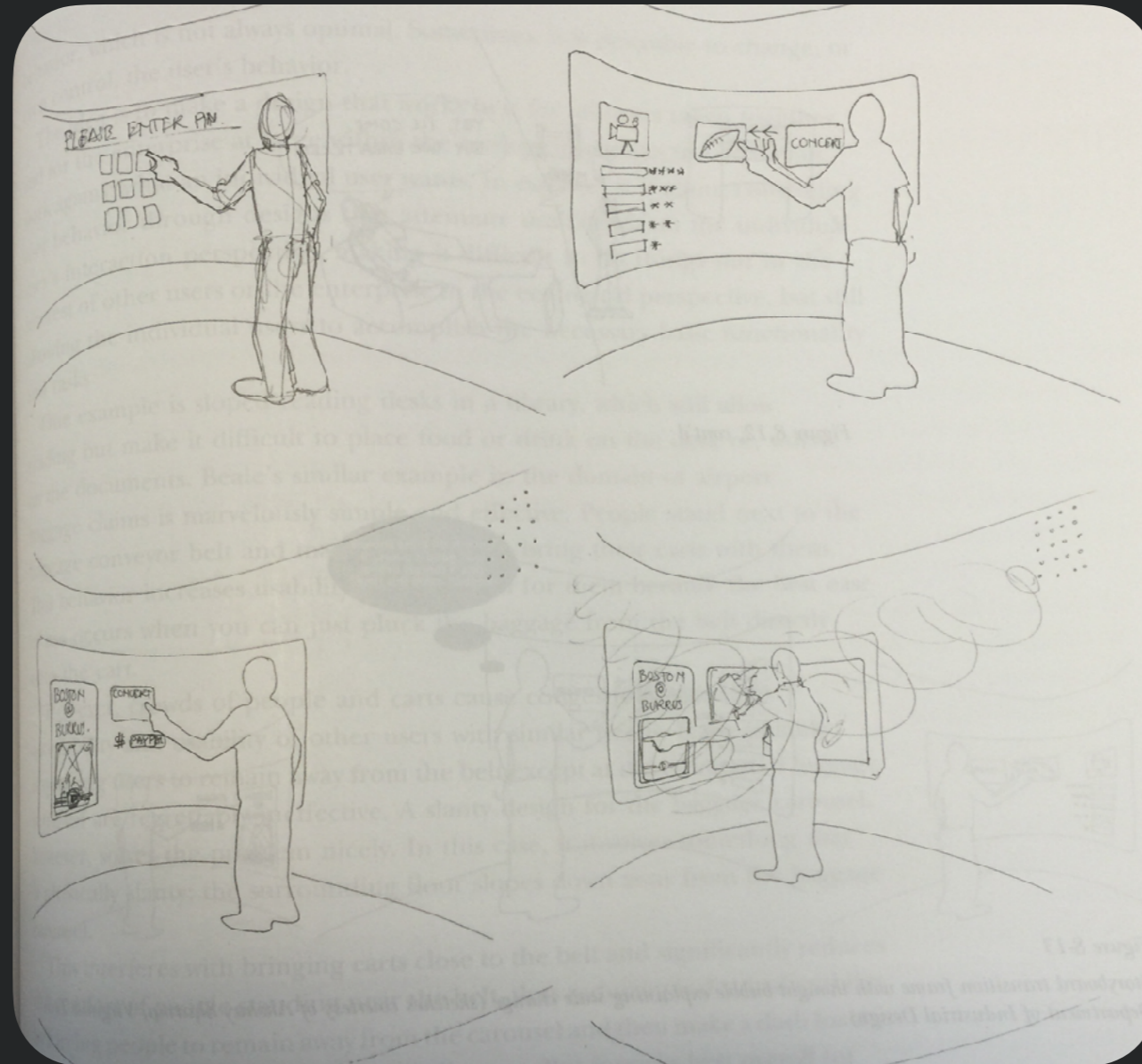
Sensor detects user & starts immersive process

Displays “Occupied” sign on wraparound case

Detects people with ID card

Example: Ticket Kiosk

Greets buyer and asks for PIN



Shows recommendations & most popular categories

Buyer selects “Boston symphony at Burruss Hall”

Plays music from symphony, shows date & time picker



Frame Transitions

- Transitions between frames particularly important
- What users think, how users choose actions
- Many problems can occur here (e.g., gulfs of execution & evaluation)
- Useful to think about how these work, can add thought bubbles to describe

Wireframes & Design Critiques

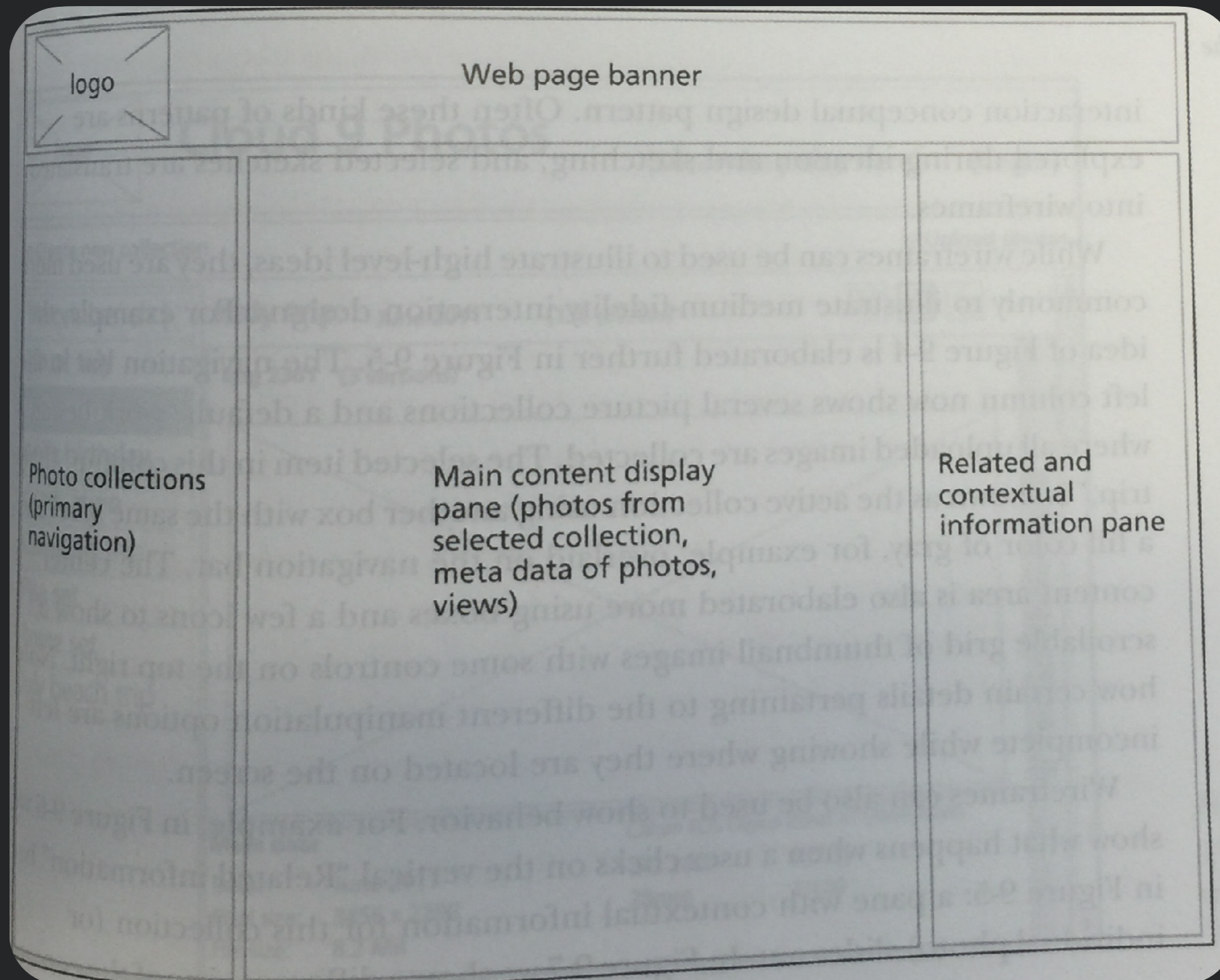




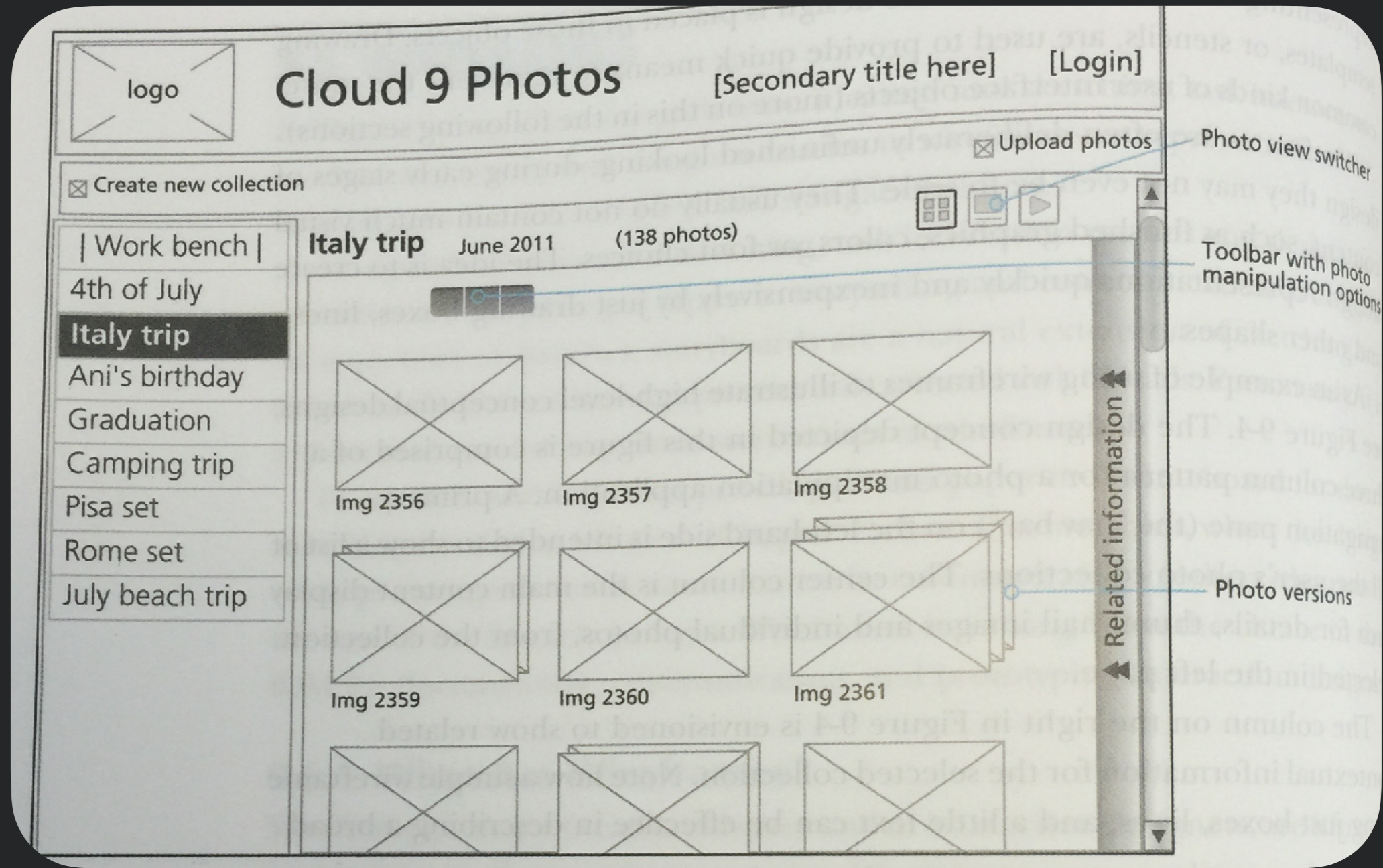
Wireframes

- Lines & outlines (“wireframes”) of boxes & other shapes
- Capturing emerging interaction designs
- Schematic designs to define screen content & visual flow
- Illustrate approximate visual layout, behavior, transitions emerging from task flows
- Deliberate unfinished: do not contain finished graphics, colors, or fonts

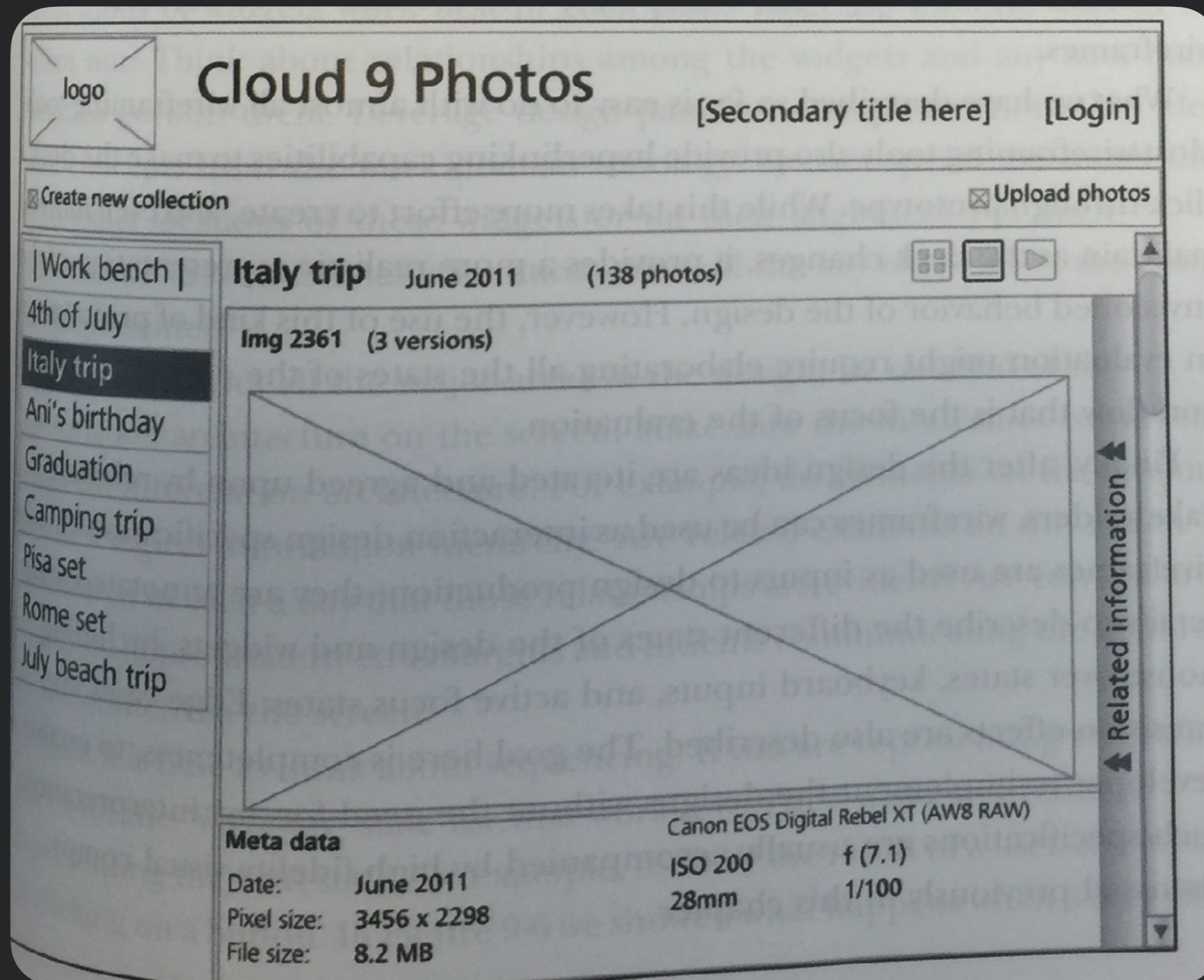
Example



Example



Example





Wireframes

- Can be used to step through a particular scenario
- Focus on key screens rather than every screen
- Tools can help
 - Can be made clickable
 - Can use stencils & templates; copy & edit similar screens



Creating a Wireframe - (I)

- What are the key interactions needed to support design?
- What widgets support these interactions?
- What are the best ways to lay them out?
- How do these relate to conceptual design & user's mental model?



Creating a Wireframe - (2)

- What are all of the items: toolbars, scrollbars, windows, ...?
- Are there too many widgets on the screen?
- What happens when data is larger than available space? Will entire page scroll, or individual panel?
- How much detail of items to show?

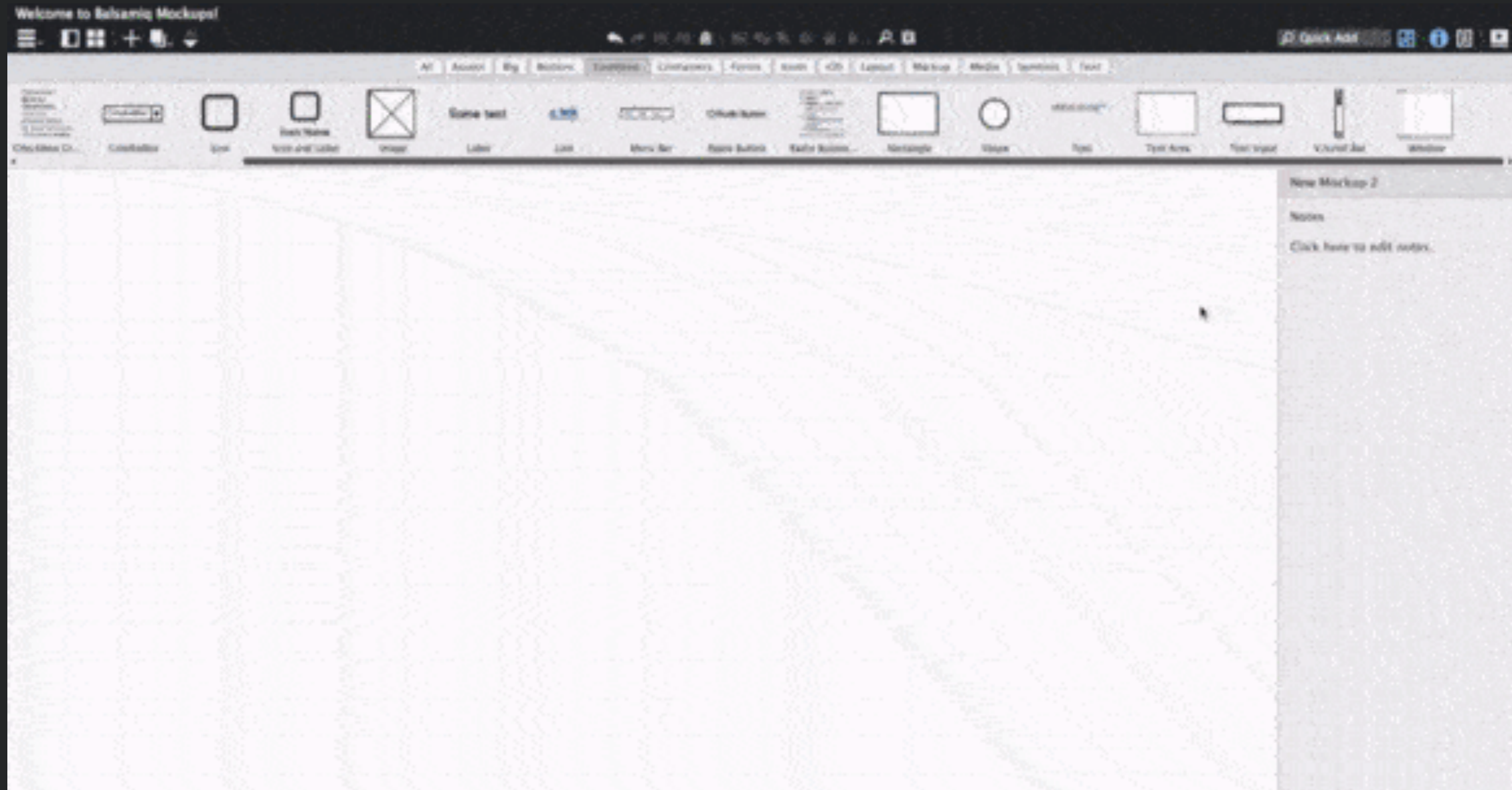


Example Tool - Balsamiq





Example Tool - Balsamiq



Design Critiques

- Stylized meeting for getting feedback on design sketches & prototypes
- Solicit feedback from peers
- History: studio art education



<http://www.flickr.com/photos/pjchmiel/2972140234/>



Designer: Frame the Discussion

- State ***explicitly***: What would you like comments on?
 - Overall idea?
 - Usability?
 - Specific interaction design?
 - Visual design?
- Take a **dispassionate** stance (this is hard!)
 - Show alternatives where possible



Critic: How to Avoid Deaf Ears

- Comments about the *design*, not the designer
- Point out positive aspects - be *specific*
 - Not: “I like this, but...”
 - “The layout effectively communicate the hierarchical nature of the data. However...”
- Ask for *alternatives* instead of offering solutions
 - Not: “You should really change X”
 - Instead “Have you considered alternatives for X?”

Prototyping



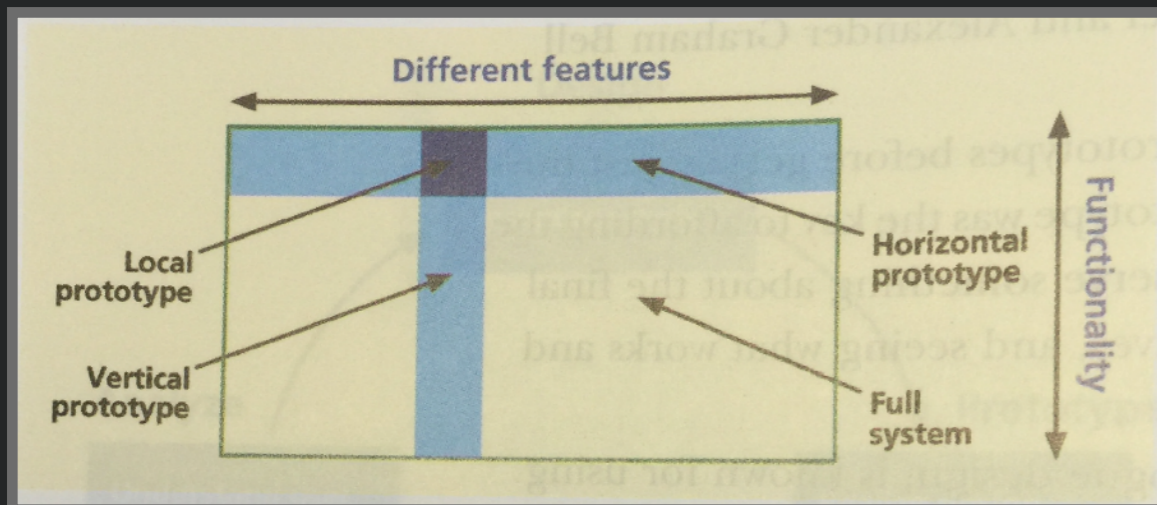


Prototyping

- How do you know your system design is right before you invest the time to build it?
- Answer: prototyping!
 - Evaluation performed **before** investing resources in building finished product
 - Early version of system constructed much **faster** & with less expense used to evaluate & **refine** design ideas

Types of Prototypes

- Which details do you leave out?
- **Horizontal**: *broad* in features, less depth
 - Explore overall concept of app, but not specific workflows
- **Vertical**: lots of *depth*, but only for a few features
 - Enables testing limited range of features w/ realistic user evals
- **T**: most of UI realized at low depth, few parts realized in depth
 - Combination of vertical & horizontal
- **Local**: focused prototype on *specific* interaction detail





Interactivity of Prototypes

- Scripted, click through prototypes
 - Prototype w/ **clickable** links to move between screens
 - Live action storyboard of screens
 - Simulates real **task flow**, but w/ static content
- Fully-implemented prototypes
 - Usually **expensive** to implement actual system
 - But can build key piece of system first to evaluate

Wizard of Oz

- Goal: *simulate* actual system w/ out building it
 - Want user to interact *as if* they were interacting w/ real system
 - Helps explore how users would interact w/ novel interaction if it were to exist
- Example: natural command line (Good et al 1984)
 - Users typed in commands to interact w/ computer
 - Commands intercepted by hidden human who interpreted commands & executed them

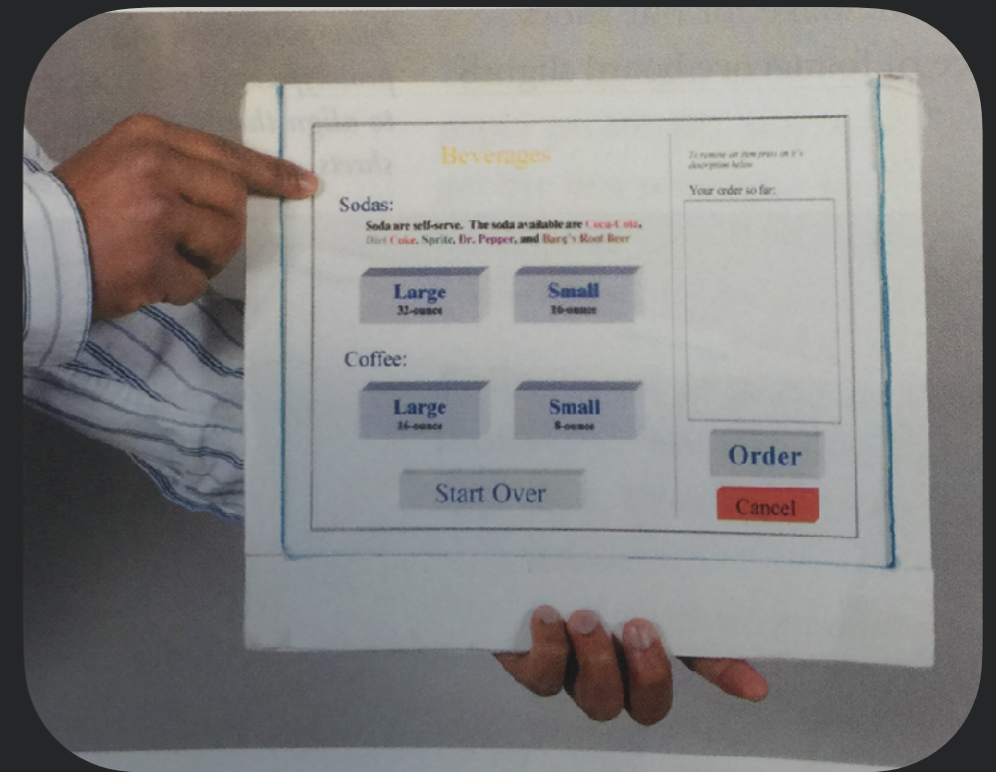


Paper Prototypes

- *Low fidelity* prototype w/ paper mockups
- **Goal:** get feedback from users early w/ very low cost interactive prototype of envisioned interaction design

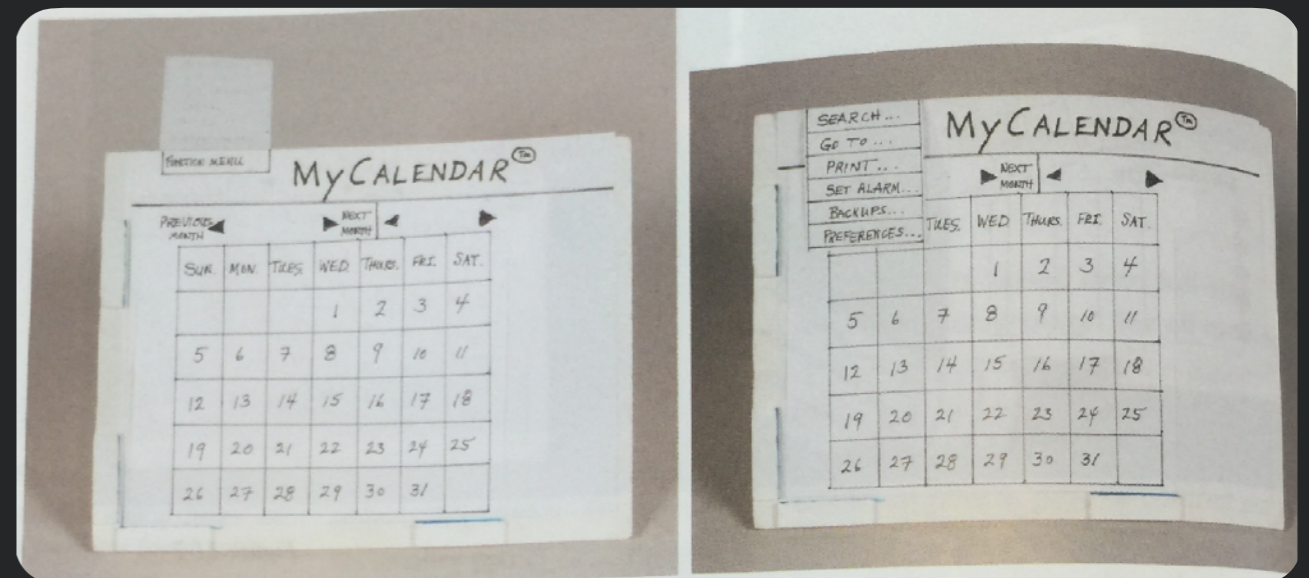
Paper Prototyping (I)

- Set a realistic deadline
- Gather set of paper prototyping materials
- Work ***fast*** & do not color within the lines
- Reuse existing sketches & mockups
- Make underlying paper mockups of key screens

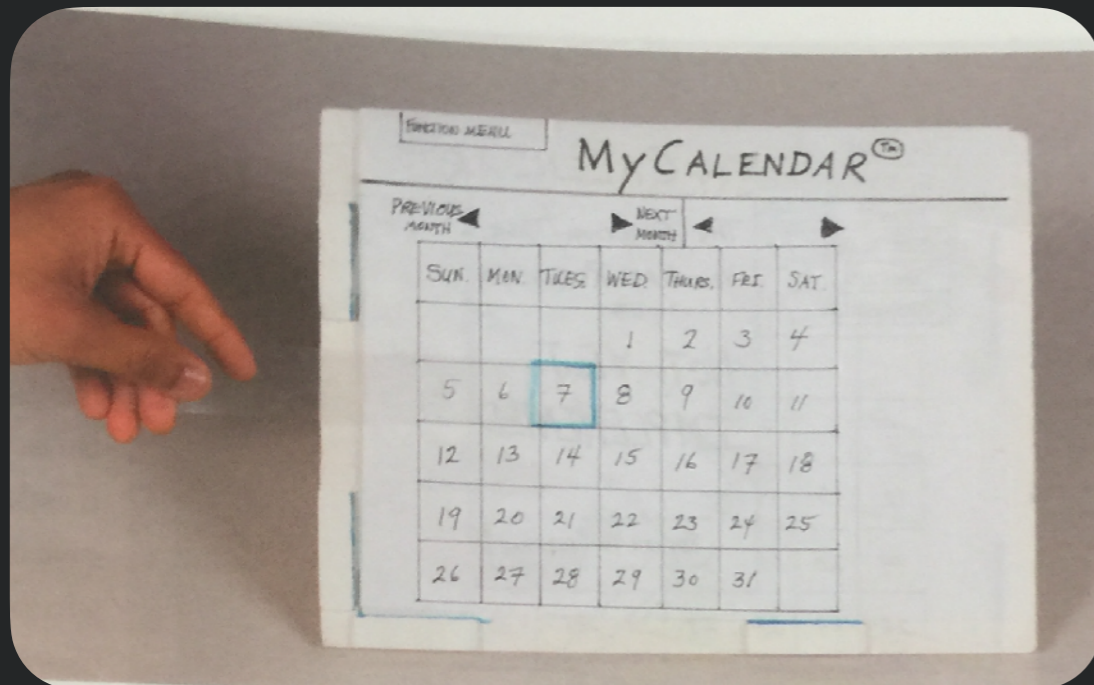


Paper Prototyping (2)

- Use ***paper cutouts*** & tape onto full-size transparencies as “interaction sheets” for moving parts, making modular by including only a small amount
- Do not write or mark on interaction sheets
- Be ***creative***
- ***Reuse*** at every level
- Cut corners wherever possible (trade accuracy against efficiency)
- Make a “this feature not implemented” message



Paper prototyping (3)



- Include “**decoy**” user interface objects not needed for expected tasks
- Accommodate data value entry by users w/ blank transparencies
- **Organize** materials to manage complex task threads
- **Pilot** test thoroughly

In Class Activity





Group activity

- In groups of 2:
 - Pick one of the 2 scenarios from last time
 - Start with a specific set of user needs identified
 - Create Wireframe design of a new system that better addresses the users' needs
 - Build storyboards w/ separate wireframe “pages” for at least 2 separate scenarios
 - 25-30 minutes



Scenario: User

- You want to book a safe trip to the movies for you and your two roommates during COVID-19. You know that you want to be safe due to the virus, but need to understand the precautions movies are taking with aspects such as seating arrangements and concessions.



Scenario: Interviewer

- You work for ACM, one of the nation's largest movie theater chains. You have made several updates to the site to help with COVID-19 logistics, but you want to understand customers' needs.
- Focus: understand the factors that users consider when identifying whether a movie-going experience would be safe or not
- Note: you should **not** tell users to use a specific website, as you're interested in how they search for a movie showing more broadly.
- Take notes on user actions & think aloud



Some Findings from Contextual Inquiry

- Ease of access is important
- Booking is typically done closer to the showing time than normal.
- Certain COVID-19 precautions were unclear, making the user more hesitant about booking.
- Mental mapping of seating selection is important.



Scenario: User

- You want to grab dinner with a friend tomorrow night and need to find a restaurant. You know your friend likes fun and unique spots, but you don't want something really expensive and you're a little worried that trendy spots might not have any outdoor tables (COVID). Find a dinner suggestion for your friend.

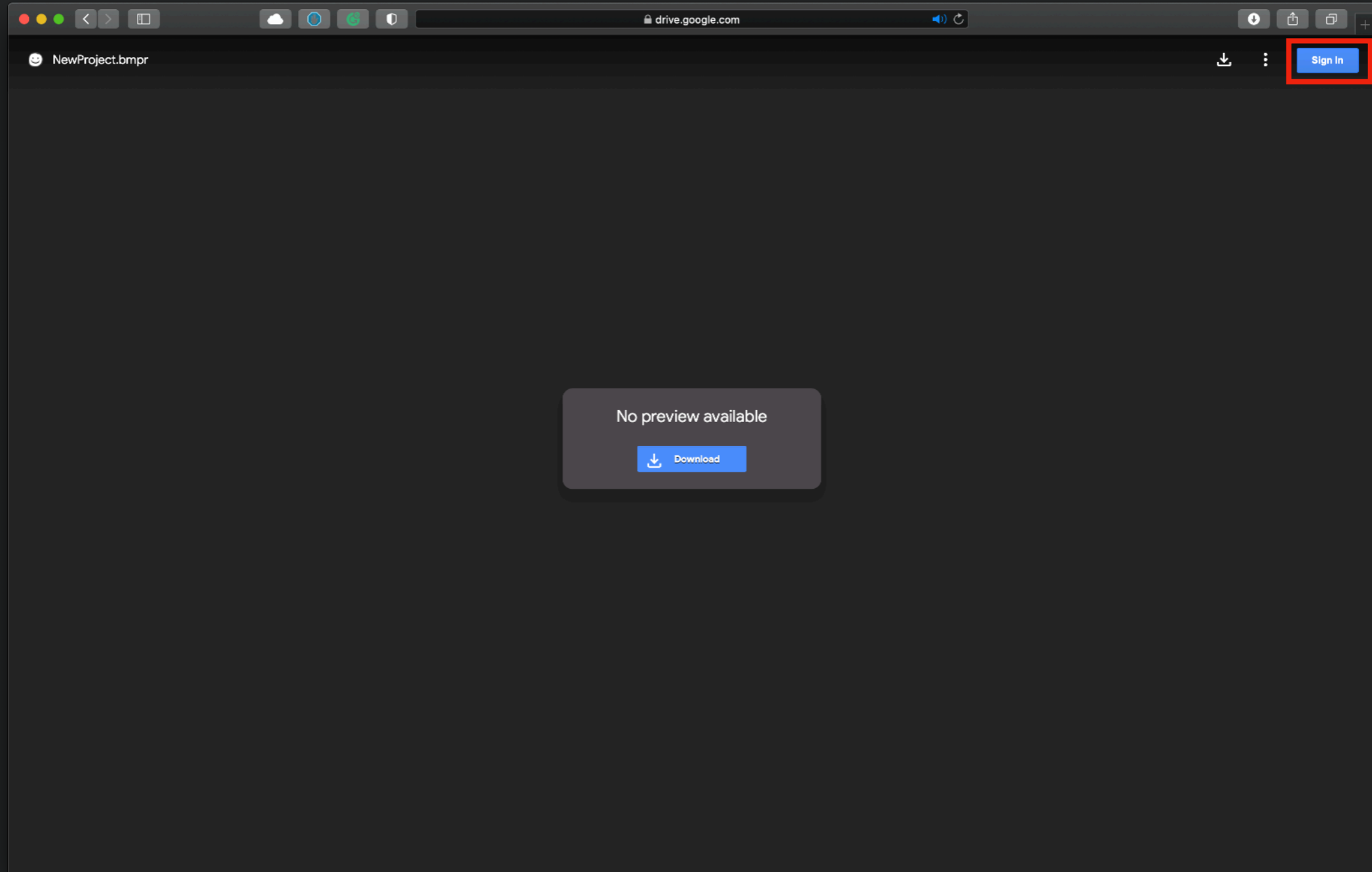


Scenario: Interviewer

- You work for Yelp. Given the many options and features, your team has become concerned that users are getting overwhelmed and are looking to streamline the process for choosing a restaurant.
- Focus: what is the process by which users identify restaurant choices and what factors influence these decisions (particularly during COVID)
- Note: You should ***not*** instruct users to use Yelp, as you're interested in understanding how they use such recommendation sites in general, not only yours.
- Take notes on user actions & think aloud

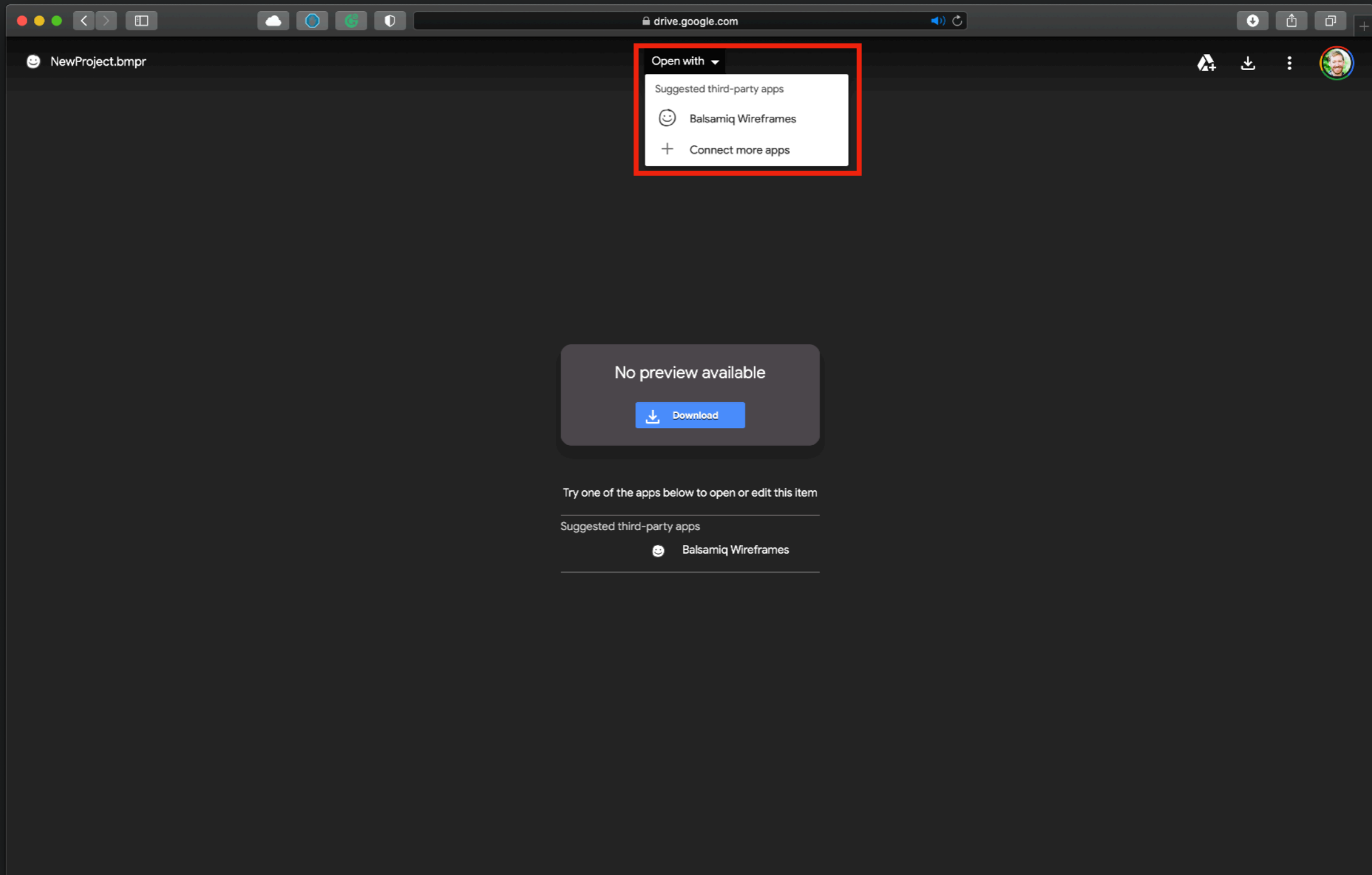


Using Balsamiq in G-Drive

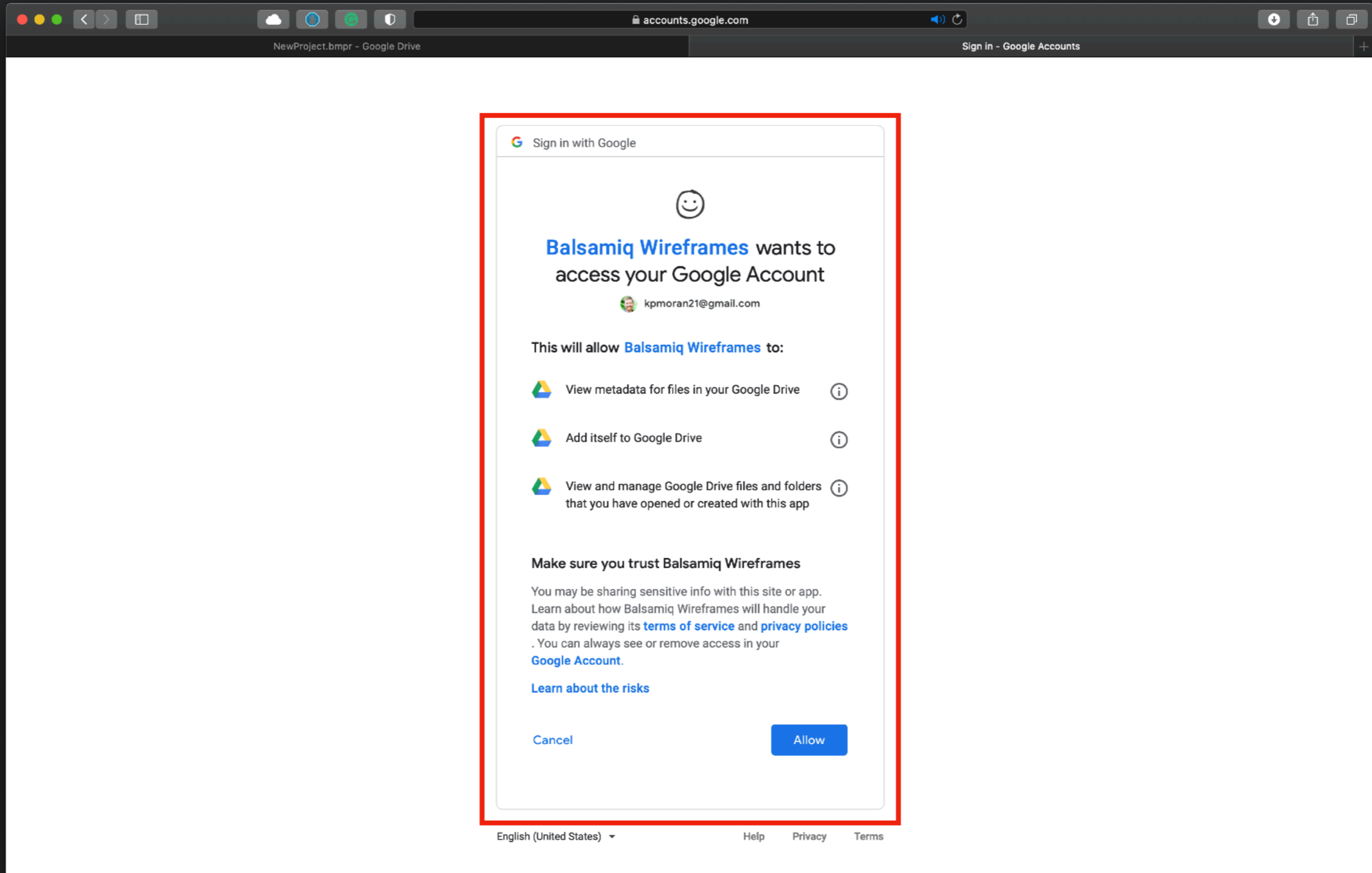




Using Balsamiq in G-Drive



Using Balsamiq in G-Drive



7 Minute Break



SWE 632 - Design & Development of User Interfaces



George Mason
University

Instructor:
Dr. Kevin Moran

Teaching Assistant:
David Gonzalez Samudio

Class will start in:

07:00

In the Chat, describe
how you **brainstorm**
ideas for a project.

SWE 632 - Design & Development of User Interfaces



George Mason
University

Instructor:
Dr. Kevin Moran

Teaching Assistant:
David Gonzalez Samudio

Class will start in:

07:00

In the Chat, describe
how you **brainstorm**
ideas for a project.



Acknowledgements

- Slides adapted from Dr. Thomas Latoza's SWE 632 course