

FRONTIER

A GRAPHIC DESIGN
EDUCATION READER



Contents

- 7 **Preface**
Colin Frazer
- 9 **Introduction**
Bryn Smith
-
- 13 **PART I**
Dear Design Educator: Six Questions
Juliette Cezzar
-
- 17 **Graphic Design History as a Critical Studio Tool**
Matt Greenwell and Aggie Toppins
- 21 **WORKSHOP**
Type and Media: A Typography Course Based on Computer Programming
Dae In Chung and Jiwon Lee
- 33 **WORKSHOP**
Desirable Difficulty in Design Education: Cultivating Discomfort for Deep Learning
Sarah Rutherford
-
- 47 **PART II**
Dear Design Educator: Funding
Juliette Cezzar
-
- 52 **Books Kill Buildings, Internet, Capital, Selves**
Other Forms: Jack Henrie Fisher and Alan Smart
- 55 **Design Essay: Meditation on Defining a New Genre**
Dimitry Tetin
- 79 **PART III**
Dear Design Educator: Boundaries
Juliette Cezzar
-
- 83 **WORKSHOP**
Teaching a Healthier Creative Process
Unsold Studio: Meaghan Barry and Lilian Crum
- 90 **Disruptive Apps: The Awesome Power of the Middleman**
Jonathan Hanahan
- 108 **You Can Touch This: An Introduction to Physical Computing and Design**
Jay Margalus and LeAnne Wagner
-
- 113 **PART IV**
Dear Design Educator: Now What?
Juliette Cezzar
-
- 121 **WORKSHOP**
Dreaming in VR
Helen Armstrong and Derek Ham
- 129 **WORKSHOP**
Illegitimate Worlds: A Workshop on Water Fictions
Dan McCafferty, Rachele Riley, and Joshua Singer
- 138 **Our Program (2016)**
Zach Kaiser
- 142 **Contributors**
- 145 **Acknowledgments**

Disruptive Apps: The Power of the Middleman

Jonathan Hanahan



As design educators, our first task is to establish for designers a zone of value and ownership in creative

practice. I propose that we do that by elevating the role of the “middleman” out of the depths of negativity into a place of true power and influence — vibrating back and forth between form and function, and taking advantage of the designer’s familiarity in both formal experimentation and commercial production.

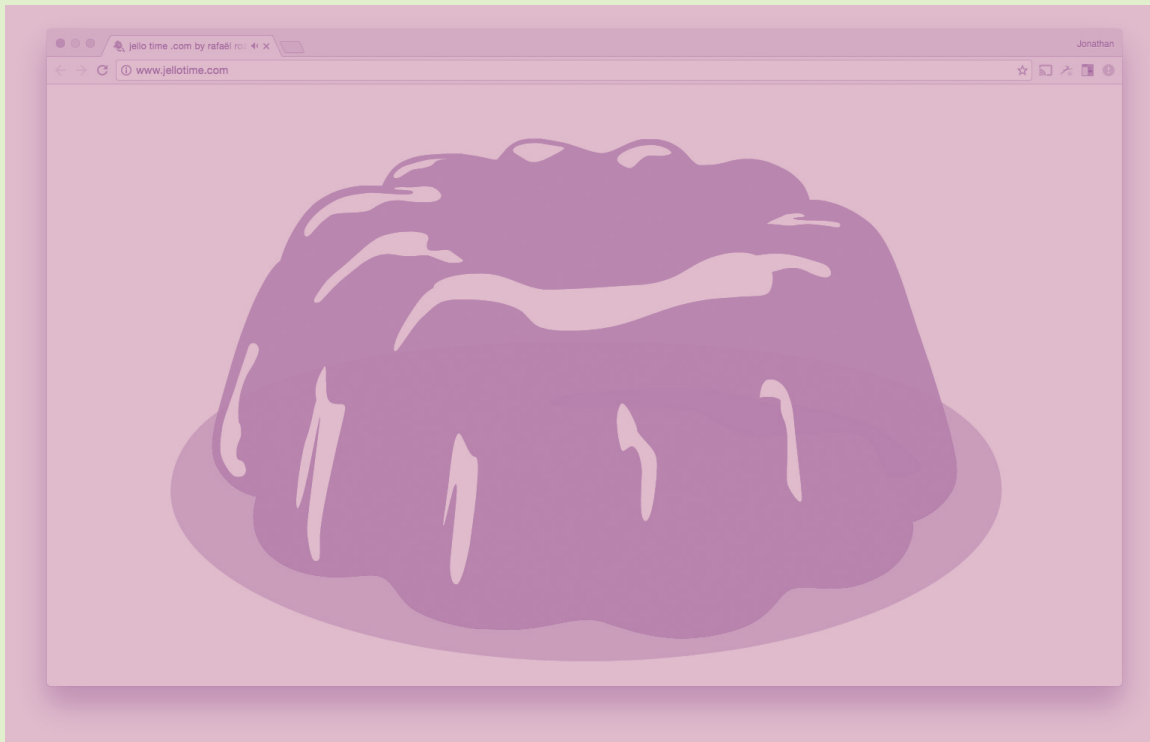
The designer is a generalist, someone who can see the big picture and connect the dots. This perfectly reflects the definition of a middleman or someone who acts as an intermediary between two entities. Here we can pull from both our artistic and engineered counterparts to produce specialized works that not only solve an individual ‘problem’ but also comment on the larger system of experience.

Unfortunately, much of design practice and education is still linked to Beatrice Ward’s “Crystal Goblet” metaphor of design practice. This leads us to associate this middleman as a negative stereotype that’s redundant and inefficient. Yet in contemporary practice this metaphor is not apt. In the act of translation, the middleman’s work is made visible. Rather than cutting out the middleman, design should be celebrating and building upon it. This moment of transition from creative experimentation to product development establishes a hugely important zone of

translation. Rather than minimize the visibility of that translation in our work, what would happen if the designer celebrates ownership of this middle space?

Not design first code later, but design and code together

Reframing the designer as middleman requires a reevaluation of how we, as educators, introduce technology. The familiar model of design education is to frontload the curriculum with type and image manipulation using authorship-based platforms like Adobe’s Creative Cloud. This traditional model provides foundational understanding and technical development in a student’s early years, and creates space to experiment later on. Once students learn how to design they are then introduced to algorithmic or procedural operations of code and markup. Unfortunately, this is simply too late for a number of reasons. First, this structure establishes a distinction between how we produce for traditional media versus how we produce for dynamic media. It also shrinks the timeframe that students can explore through coding (and its potential as an alternative output to authorship), hinders experimentation and creates a classroom environment focused solely on learning new skills. In this model, students struggle through learning each step, solve a problem, and then move on to the next with little time for reflection.



← Rafael Rozendaal,
Jello Time

Experimentation is greater than skill acquisition

If the middleman is less focused on being fluent in any one particular technical skillset and more on understanding many, is filling up class time with instruction on advanced processes, languages, and procedures really serving the interest of the collective student body? I would argue that this negatively affects the student experience in three distinct ways.

First, when interaction courses focus on learning tools and not on exploring the potential experimental results of those tools, they reiterate the stereotype that interaction equals technology and code, and produces digital outputs. We know that this isn't the case. Interaction design is about developing processes for the generation of adaptive experiences, but it's not limited to only the digital experience. Physical media is increasingly included in the same interactive dialogue as digital, and this type of thinking should be encouraged. Design curator and critic Andrew Blauvelt describes this shift in the essay "Towards Relational Design," as "the nature of design itself has broadened from giving form to discrete objects to the creation of systems and more open-ended frame-works for engagement: designs for making designs."

Second, the transition into a code-based design process is a huge hurdle for many students to handle. Because they're trying to digest this new procedure and learn the details of language and syntax for multiple new tools, the focus of the learning is on the tool, and not what can be done with the tool.

Finally, the skill acquisition model lends itself to pushing designers who excel in code down a track of training to be developers. Those that latch on to the operations and syntax immediately learn more and more code. Yet when advanced interaction courses are purely focused on continuing to learn more advanced languages and technologies, students who struggle with the initial translation to a design process based on code are left behind. Simultaneously, this model also puts those students who excel at code at a disadvantage. They start training as if they're developers and their education gets watered down. Their design skills lag behind while they polish up their development skills, and they're left as mediocre in both the design and development space.

Middleman Products

To counteract this trajectory I am proposing two tactics for infusion into interaction-based curriculum that will open up spaces of ownership and experimentation for all

students and designers. The first is to establish a series of products that designers can own. Rather than being purely formal (artist) or purely product-oriented (developer), these products follow the model of tools which add functionality to *existing* platforms and daily experiences. Examples are tools like Browser Extensions, Bookmarklets, and Twitter Bots (though there's much greater potential). These products are small and compact elements that we engage with everyday as designers and users. They affect, change, and distort experiences rather than making new ones from scratch. This process reflects what interaction designer Ralph Ammer refers to as "Natural Design." In this model, we "make ourselves smarter rather than the things around us."¹ These tools inject themselves into our daily experiences with technology, and create opportunities for advanced customization, while creating moments of meta-reflection. They can encourage users to pause momentarily and reflect on their patterns and activities — what they're doing and where they're spending their time. The goal is for these products to both serve a specific purpose and reflect on a larger dialogue at hand. While I'm not echoing the biological-based aspect of Ammer's concept, I'm in support of the dialogue that a designer's output should support and enhance existing life, rather than proposing new forms.

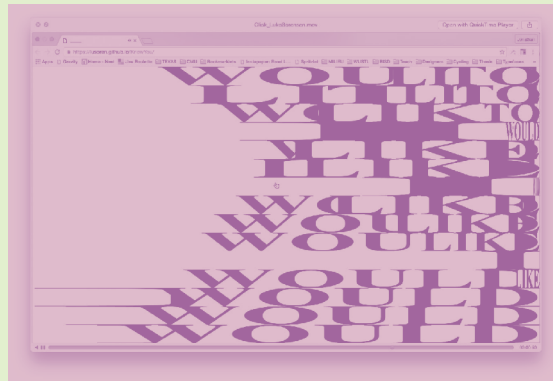
In the classroom, the constraints of these output paradigms allow students to focus on refining experiences and not building full systems. Rather than the concern of building an entire website from scratch, these tools are simple (usually only a few lines of code). Students are able to spend time tweaking and evaluating a simple operation and how it affects an overall experience, rather than building the entire platform. This provides ample time for formal experimentation, and time to reflect on the realities and the idiosyncrasies of digital experiences.

Experimentation with simple tools

The second tactic, is to adjust the classroom to be a space for experimentation using simple functional interactions. I've been practicing this tactic through a senior-level studio taught at Washington University in St. Louis titled "Form and Interaction."

1 — blog.prototypr.io/interaction-design-is-dead-what-now-db09d77cadae

2 — microinteractions.com/



← Student work, Form & Interaction



← Student work, Form & Interaction

The premise of this course is based on four rules. First, the projects are all action-based. The assignment provides the class with a micro-interaction,² for example a click, a scroll, a drag, etc. Students must develop their entire projects around this singular experience. To exacerbate this restriction, projects aren't allowed to have any real-world variables or expectations. There's no client, no content, no budget. Students then develop multi-sensory experiences that are dictated and controlled by the action. In this model, the action is not in support of a narrative, it is the narrative. The goal of this is to free the students from the burden to solve a problem. Instead, they're simply encouraged to make an immersive experience.

When simple and common actions are elevated to a space of great value, it can be hard to get students to defy the traditional

experience or expectations these actions indicate. Interaction design relies heavily on expectations of functionality to produce efficient and clear experiences. Following these expectations is a must for the developer, but the new middleman designer has the opportunity to defy expectation. In fact, challenging expectations may create unexpected yet deeper immersion in an experience. To encourage students to break through and challenge themselves, I start each project with a one-week charrette. In this charrette, students must develop 100 individual results using the given action/interaction. This framework, though excessive, is extremely efficient in getting students to break down the established expectations of what a particular action means, and develop alternative, tangential experiences. It's easy to develop the first 20, but the remaining 80 can prove deeply challenging. The charrette encourages students to stop thinking logically and this is where exciting things begin to unfold. Students begin asking dumb, naive questions of their projects. They start doing silly things simply to fulfill the requirement and along the way often find profound results that become the foundation of their full-fledged projects. Many times this involves linking atypical sensory experiences, drastically shifting scales, defying spatial restrictions, and stretching activity times between extreme highs and lows.

The last and perhaps most important component of this framework is that students are to use only the skills they learned in their Introduction to Web Design course, meaning all that's expected is an understanding of HTML, CSS, and maybe some Javascript. Students are welcome to add as many other components as they wish, but no new skills will be taught or expected. The studio is about spending time reflecting on the foundational structure of their known skills and pushing them into unexpected territories.

In the end, the projects that have come from this class have been profoundly valuable for both myself and the students. Students start to realize the deep and powerful oppor-

tunity of simple algorithmic procedures in creating immersive experiences. They break down the expectation that interactive projects must be something complex, robust, and high-tech. And ultimately, students begin to take ownership over a space that isn't simply formal and not purely productive, but experimental, two-way, and dynamic. It's this in between or middleman space, that's more deeply interesting and powerful than the space that either the artist or developer resides in alone.



The Colstrip plant's coal ash waste is being disposed of in wet storage.

Reservoir
of
WHAT?

Disruptive Apps: The Power of the Middleman

Jonathan Hanahan

Capstone 1: Form & Interaction

Course Description

By now in your code/interaction education you have been introduced to a rich assortment of processes, languages, best practices, etc. that serve as the foundation of designing for interactive media. That foundation is full of actions and artifacts that have been established after years and years of development and use. This constant use has established a set of expectations in both users and designer/developers for what simple actions will result in; expectations which have incredibly powerful implications when designing for the web. They allow us to build complex websites which users can nimbly navigate to reach their designated content. Conversely, these common practices can — and often do — result in repetitive/stale aesthetics and experiences on the web.

This class is about navigating that delicate line between expectation and surprise. Students will analyze the current use of actions and challenge their common behavior by creating critical, unexpected, and profound action-driven digital experiences. Rather than use tools to transmit or navigate to content, students will create environments from these actions that encourage play, exploration, participation, and contemplation.

Objectives

- Establish a space of critical discourse of common web design practices and aesthetics
- Explore the opportunities of sensory and spatially-driven web projects
- Expand on current knowledge to develop deeper expertise in designing for interactive platforms
- Learn to work analytically and intuitively through code
- Develop a practice of authorship for web-based experiences

Assignments

This class will be broken up into three prompt-based projects. Each assignment will result in a single-serve website. To begin each project, students will embark on a one-week charrette which will explore as many possible representations of a provided digital action. From these explorations students will select one or more results to build out a full-fledged immersive digital experience. Each project should be developed at a desktop resolution. You are welcome to expand beyond to create responsive frameworks and customize experiences for handheld or large displays, but these are not requirements. At minimum, your project should display correctly on any of your classmates laptops.

Expectations and ground rules

This class is 100 percent about expectations, or more specifically defying them. That said, there are a few expectations and ground rules.

1. No content; content = action

Projects will be prompt/action-based. You will not be assigned any content, narrative goals, or story to tell. You are free to do anything as long as you execute the action dictated by the prompt. In this sense, your content is the action and the experience that action results in. You will not be telling a story but encouraging exploration and play in your audience. Your projects should strive to be less passive and more active, less like reading environments and more like gaming environments. Your goal is to dive deep into the prompt and create an unexpected and immersive experience. Successful projects will be rich in user experience, not necessarily knowledge consumption. Explore as you see fit.

2. No client, no money, no real world

For this class you are freed from the inevitable burden of a business and commerce-driven output. For the hours we spend together and those you put into your projects, I want you to forget about the real world (where design is dictated by commerce, bounce rates, loading times, legibility, etc). For this class focus on pushing the limits of what we can do in the browser. You have no one to answer to but yourself. Your projects do not need a purpose or a commercially viable outcome. You are not developers seeking to make the most functional application, you are artists and designers creating rich and profound multi-sensory artifacts. They can be stupid, slow, scary, antagonistic, playful, pointless, annoying, frustrating, functionless, or ugly. Push yourself, go crazy, take risks, and have fun!

3. More code ≠ better code (or better projects)

While you'll be working on screens and in code, this class is less about learning new tricks and more about expanding and deepening your existing knowledge in unexpected ways. The

focus will be on dissecting, analyzing, and pushing the limits of many simple and common digital actions you already have great familiarity with. While we'll conduct several tutorials and introduce some plug-in opportunities for specific projects, there is no expectation for advanced or difficult coding languages and the number of plug-ins or fancy tricks implemented will not directly correlate to a good grade. Your goal is to make the most profound experience possible; for some that may come from very simple programs or languages implemented in unexpected ways. Don't think you need to be a super-tech to do well in this class. All you need is to be willing to analyze and critically approach each project. Strong projects will have conceptual, experiential, and critical results and may potentially be very code-light. Remember, at the heart of any successful single-serve site is a simple and profound action.

4. Versioning

Working in code means making thousands of alternative versions of individual programs in a quick and efficient design process. This means that for each meeting I expect a great deal of work and a multitude of iterations. Versioning, testing, experimenting, expanding, breaking, remixing, and reevaluating are of the utmost importance. Use this process to your advantage as your grade will be dependent on both your process and final outcome. As a rule of thumb you should come to each class with at least ten different variations of your project, though I expect much more than that. If you have any less than ten you'll forfeit your critique time for that day. Push yourself and do not think too much, if you ask your project a question, make a version as evidence of experimentation. It's also suggested that you set up your projects in Git. If you have not used it before, git is a version control system which allows you to share, collaborate, change, and store your code. When used correctly it will also make sure you never lose your work or make a change that

breaks the whole project. Git is also very helpful for taking code from others and customizing it, as many in the creative coding world post their work there regularly.

Integrity policy

When working with code it is quite common and encouraged to use code/programs/plugin-ins developed by other people. I highly encourage you to cut and paste other people's code and turn their work into your own. That said, if you use anything developed by someone else and published as such, be sure to credit them in your code. Plug-ins often have credits but if they don't, be sure to indicate where you got them from and who did the work. If you have any questions about how to credit, don't hesitate to ask.

Evaluation

Your performance in this class will be evaluated based on your participation, effort, and the quality of your work.

Project 1 – Mouse Events

The act of clicking is one of the most fundamental experiences on digital platforms. It's how a user chooses what they wish to inquire upon. The role of the mouse or trackpad also is invaluable in digital space. It allows us to move freely in any direction quickly. While the single action of pushing a button may be incredibly simplistic, it's surrounded by a multitude of events leading up to and after this action (mouseenter, mouseleave, onclick, oncontextclick, mousemove, mouseout, mouseup). For this project you'll develop a single-serve website completely driven and only using mouse events. This site can be composed of multiple click events or a single one, but no other actions should be involved.

Think about the sequential relationship leading up to and after the click, how do they influence one another? How do you imply what the user should expect (and how do you mask or alter that expectation)? What is the relationship between the first click and the 50th? Are you clicking the same thing over and over or clicking a multitude of things one time? How might a single click activate multiple senses simultaneously or what factors dictate which sense it activates? How does scale, location, and proximity of events to one another affect the resulting experience? What happens when a single click affects multiple elements, and vice versa?

As always, the project needs no knowledge-based goal. It can be a purely sensory playground dictated by non-linear exploration, a complex game-style environment encouraging users to do particular tasks, or something entirely different. Your goal is to activate and captivate your audience's participation.

Objectives

- Increase your familiarity, comfort, and knowledge of mouse events
- String together a complex, compelling, and engaging experience from a simple action
- Challenge and analyse the status-quo of how mouse events are typically implemented and how they might alternatively be used
- Break down sensory barriers to create multi-sensory and spatial experiences
- Explore activating unexpected elements on screen

Examples

jellotime.com/
pleasetouchme.com/
misternicehands.com/
onandoff.org/
beefchickenpork.com/
papertoilet.com/
swissarmyman.com/
wheresthepixel.com/
learningprocessing.com/
linkedbyair.net/Shrub
linkedbyair.net/Bug
collectie.hetnieuweinstituut.nl/en
studiorotterdam.hetnieuwe
instituut.nl/en
pointerpointer.com/
display.xxx/
convoy.me/#!/landing

Charrette

Develop 100 alternatives for a single click event using HTML, CSS, and/or javascript/jquery (below).

Submit a single page website with all 100 experiments.

HTML:

```
<button type="button">Click Me!</button>
```

javascript/jquery:

```
$(Element').click(function(){  
YOUR CODE GOES HERE  
});
```

css:

```
Element.active {  
YOUR CODE GOES HERE  
}
```

Project resources

<http://api.jquery.com/>

<https://developer.mozilla.org/en-US/>

Google ;)

Project 2 – Scroll

With the increasing use of smaller devices, our platforms and experiences are “less about the searching and more about the getting.” Apps with pre-described and packaged experiences thrive in our everyday life. One for Facebook, one for Twitter, one for weather, one for sports, one for news. Each app allots a prepackaged experience. More often than not, the act of scrolling is the most important navigational element on mobile devices.

For this project you’ll develop another single-serve website — this time specifically for a mobile device—only using scrolling to dictate experience. The whole class will use a scroll-based animation plug-in called Skrollr (<https://github.com/Prinzhornskrollr>). As in your previous explorations with mouse events, your goal will be to develop an immersive and sensory experience over the course of a single scroll. You’ll link and manipulate animation elements to encourage participation, exploration, and investment in the experience. As before, no content will be provided or used.

Think about what it means to scroll. What a single scroll versus the entire journey entails. Particular time will be important in this project. How do you reward your users for continued participation? Do you need to finish the entire length of the site to get the intended experience or can shorter durations be appropriate? Be sure to consider the opportunity that moving beyond the viewport allots. Does your user feel as if they’re moving or they’re moving artifacts (or a combination of both)?

While your project can function on the desktop or other devices you’re designing a mobile-first product. Be sure to consider and contemplate the difference between scrolling on a device versus on a trackpad or mouse.

Objectives

- Increase your familiarity, comfort, and knowledge of scroll events, particularly data-attributes and the skrollr.js library
 - String together a complex, compelling, and engaging experience from a simple, repetitive action
 - Experiment with linking usually disconnected elements and effects
 - Challenge and analyze the status quo of how scroll events are typically implemented and how they might alternatively be used
 - Break down sensory barriers to create multi-sensory and spatial experiences
 - Explore activating unexpected elements on a mobile device
 - Developing x,y, and z-oriented spatial experiences beyond the frame of the browser
-

Charrette

Develop 100 alternatives for a single scroll event using HTML, CSS, and skrollr.js. Submit a single page website with all 100 experiments.

Project resources

github.com/Prinzhorn/skrollr

w3schools.com/css/css3_intro.asp

developer.mozilla.org/en-US/docs/Web/CSS

[developer.mozilla.org/en-US/docs/Web/ Guide/HTML/HTML5](http://developer.mozilla.org/en-US/docs/Web/Guide/HTML/HTML5)

Project 3 – Re-Size

In the mobile first landscape of contemporary interaction design, the ability for content and experiences to adapt based on the media platform from which they are viewed is often the most important aspect of a project. We design websites whose content reorganizes, adjusts, and optimizes, to make the viewing experience seamless and fluid no matter the platform. To do this, we develop a system of layouts which appear, based on pixel ratios present in whatever device is used. The relationship between these layouts is of the utmost importance and also what we'll explore in this project. In a client-based web project, we typically develop between 3-6 layouts (desktop, laptop, iPad, iPhone, etc.). What would happen if we developed 100, or 200 layouts?

In your first project, you explored the potential of the simple act of mouse events to create an unexpected yet immersive multisensory desktop experience. The second project shifted, and explored the linear and repetitive action of scrolling to unfold a mobile experience. For project 3, you'll explore the sequential and disruptive potential of shifting between different screen sizes as a tool for creating yet another multisensory and immersive experience. While the act of shifting from size to size is not regularly instigated by a user — we access a site on a device and then move on — in this project you'll be pushed to redefine what it means to be responsive, and how a responsive layout can be active as opposed to passive. Consider how the linear sequence from large to small — and vice versa — affect what is present on the screen. You'll also need to define how each screen resolution exists in isolation. Are there 'keyframe' moments which associate themselves with popular resolutions,

along with other steps between keyframes which adapt to their context? What happens on the screen is completely up to you. You have creative freedom to have multiple action and event types on screen simultaneously, and may take inspiration or use some of the experiments you've developed in past charrettes, but the action of experiential importance should be the resizing of the window. Explore how elements manipulate, what things only appear or disappear on particular sizes, how you might make extremely fine-grain shifts (vs. larger device-based jumps), etc. As always, you're encouraged to evaluate and reconsider what it means to be 'responsive' and how the relationship between each resolution can defy expectations.

Objectives

- Hone skills of designing responsive web pages that push the limits of responsive web best practices
 - Analyze the narrative potential of shifts in proportion and browser sizes
 - Design and build both isolated and connected web experiences in different dimensions
 - Investigate ideas of activating senses other than visual, based on size and device
 - Explore fine-grain shifts in sizes, both horizontally and vertically
 - Develop a hierarchical relationship between screen sizes and proportions, that has a narrative effect on an experience
-

Charrette

Use the provided web template to develop 3 different events which shift the provided div (.box) at each media query (40 total).

Project resources

developer.mozilla.org/en-US/docs/Web/CSS/Media_Queries/Using_media_queries

