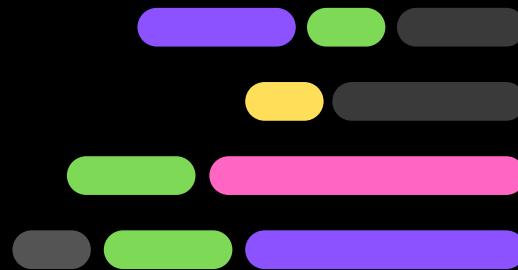




SUPERFAN

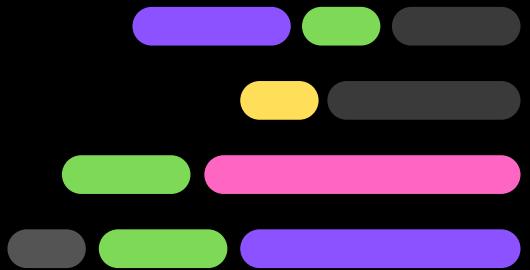
**INTERACTIVE
EXPERIENCE**





BRAINSTORM

come up with three different ideas of themes + interactive features



PHASE 0

Theme



Guilty Gear Strive choose your character

Interactive JS Feature #1

Click on a character and the screen will change to display their name + info.

Interactive JS Feature #2

And when you click on the title the opening song plays + sequence

Interactive JS Feature #3

Also larger image of character when clicked plays a voice line of theirs and changes to gif



Theme

Link Click - time travel

Click on an image to go back in time

Interactive JS Feature #1

Click on a picture a reverse sound activates + gif or image of the character who took the picture is shown + summary



Interactive JS Feature #2

Click on the title and the mcs show up + op song plays

Interactive JS Feature #3

Click on a character and name displays

Theme

Kirby - switches characters



Interactive JS Feature #1

Click on a character sound of kirby eating +a kirby image of that character appears

Interactive JS Feature #2

Click on regular kirby and a gif of kirby appears flying with a star + audio

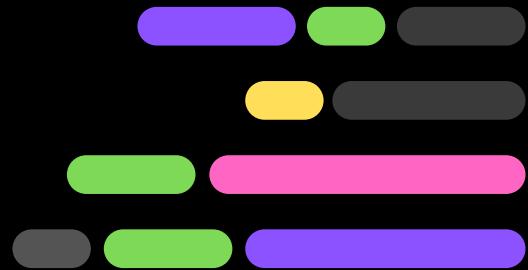
Interactive JS Feature #3

Click on kirby as character and voice line starts



EVALUATE

think about the good and bad for each idea then choose one



PHASE 1

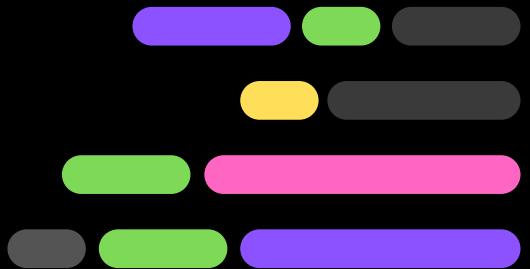
Pros/Cons of Each Idea:

	Pros	Cons
Idea 1:	<ul style="list-style-type: none">• Its cool + interactive• Gives people info. on the characters• I have more ideas of features to add• I get to code my favorite characters	<ul style="list-style-type: none">• have to limit the amount of characters• audio could be annoying to find• Might be a lot of work to find the gifs
Idea 2:	<ul style="list-style-type: none">• Interesting and connects to the plot of the show• add the op song (a favorite of mine)	<ul style="list-style-type: none">• Pictures could be annoying to find• layout might be weird
Idea 3:	<ul style="list-style-type: none">• Kirby gets turns into other cool characters	<ul style="list-style-type: none">• Gif might be hard to find (?) + low quality• Maybe too simple



CONTENT

gather and edit essential content for your site



PHASE 2

Characters + logo



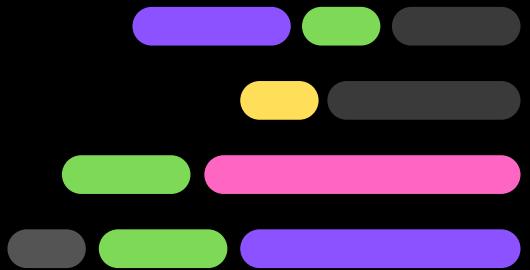
Gifs of Characters



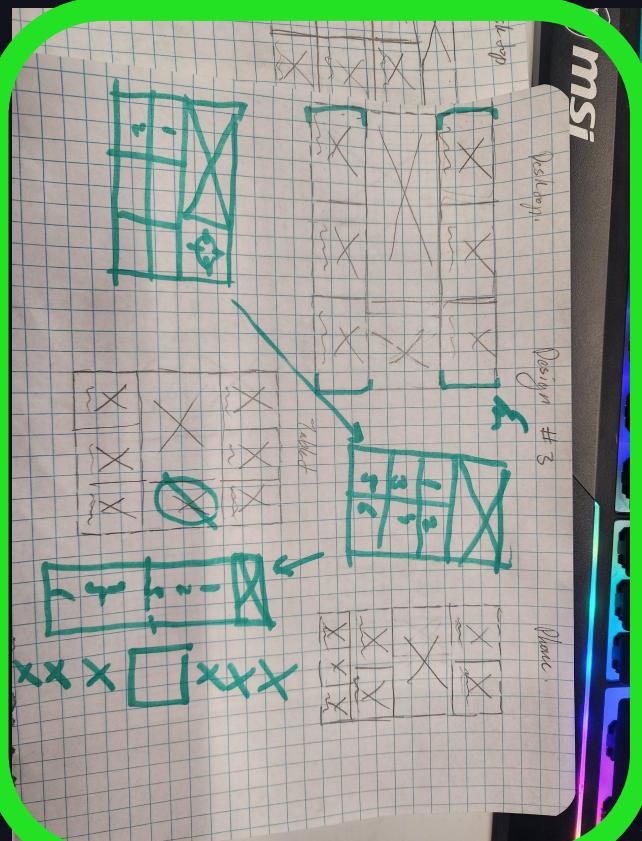
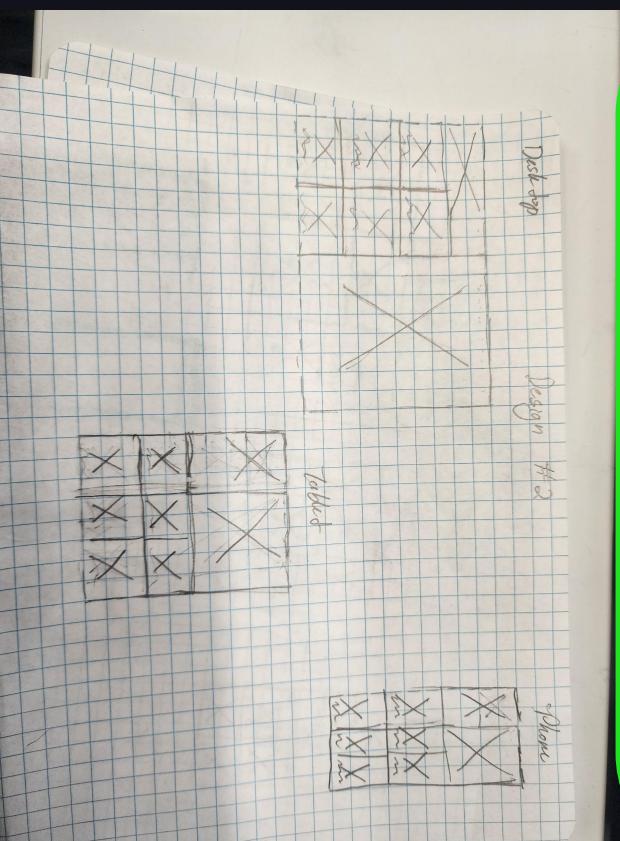
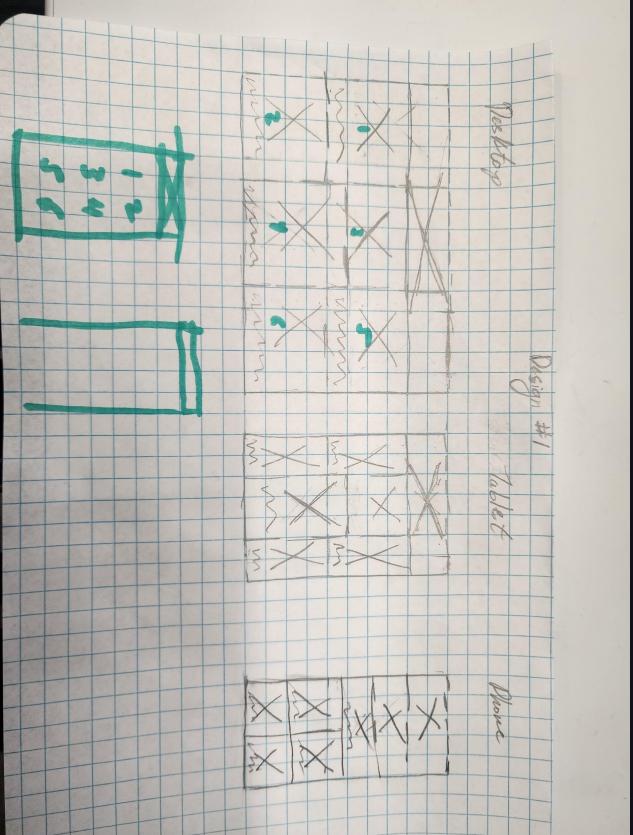


DESIGN

create three possible responsive designs for your chosen idea



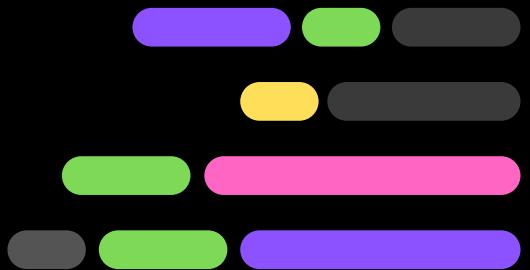
PHASE 3





MOCKUP

choose your final layout and create a digital mockup



PHASE 4

GUILTY GEAR

— STRIVE —



Sol



Asuka RII



Bridget



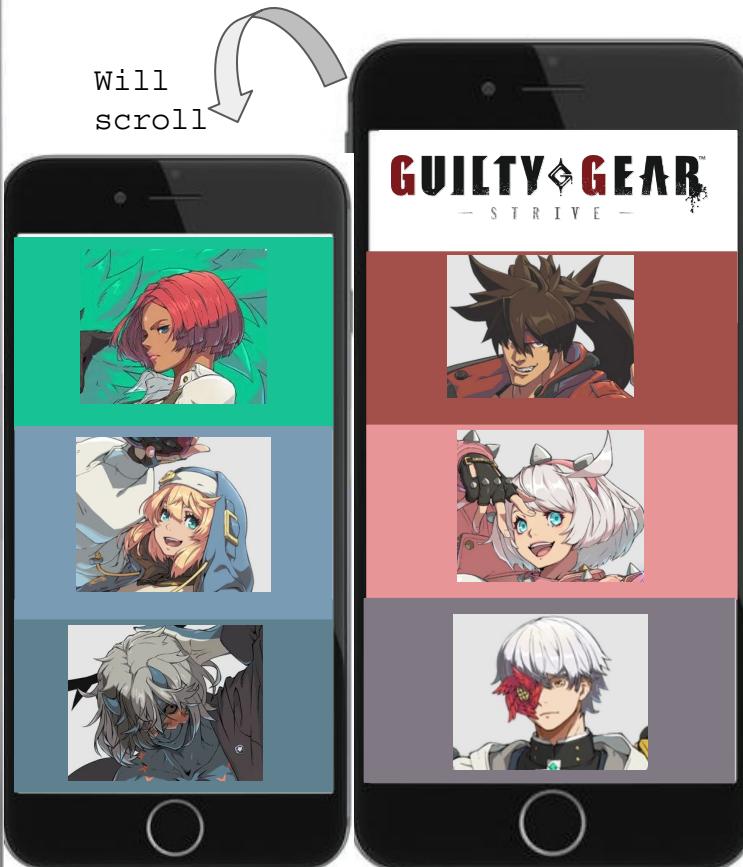
Elphelt



Giovanna



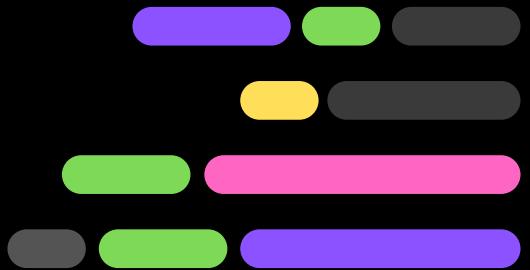
Happy Chaos





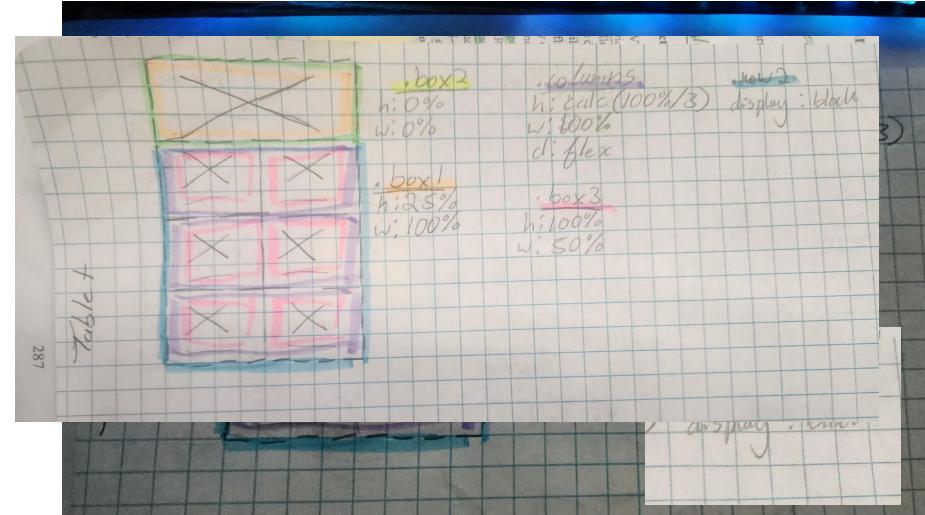
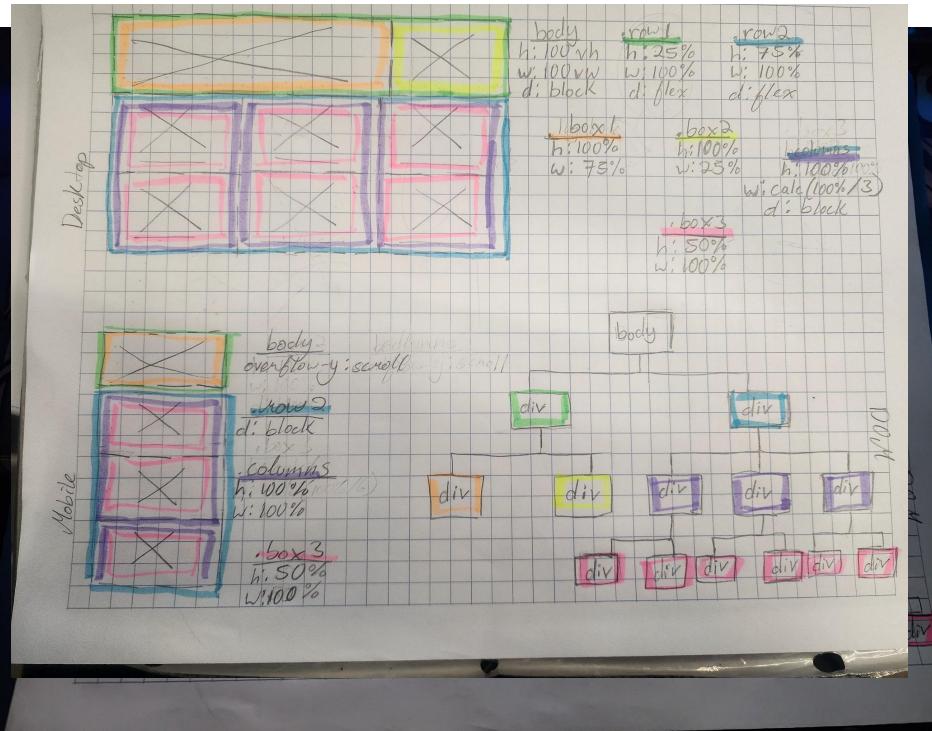
ANNOTATE

add explanations, colors, percentages, and display properties + draw your dom



PHASE 5

Note to self- retake pics



HTML



CSS

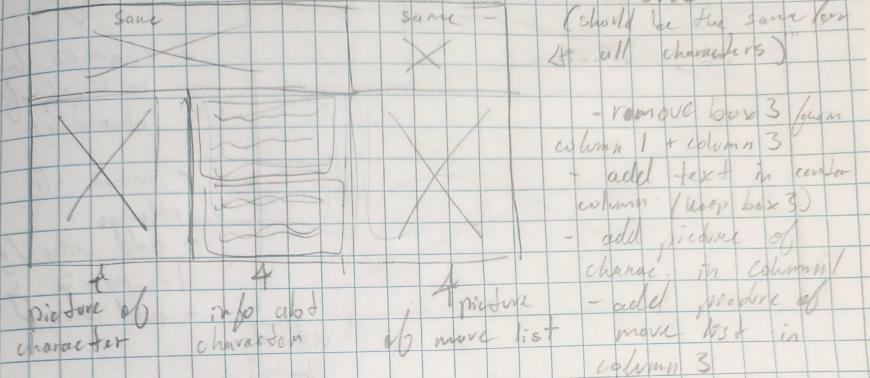


Java Script

*1 Click on character

Once

(should be the same for
all characters)



*2 Click on character

twice.

- = char. mouse plays.
- = voice line plays.

= remove columns 2 + 3

At change now
background

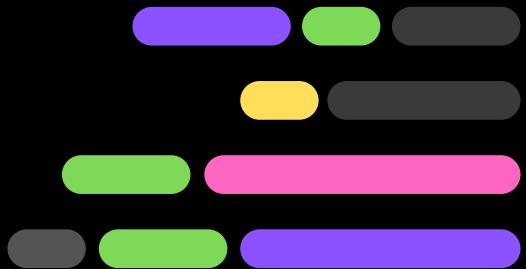
A displays of b #1



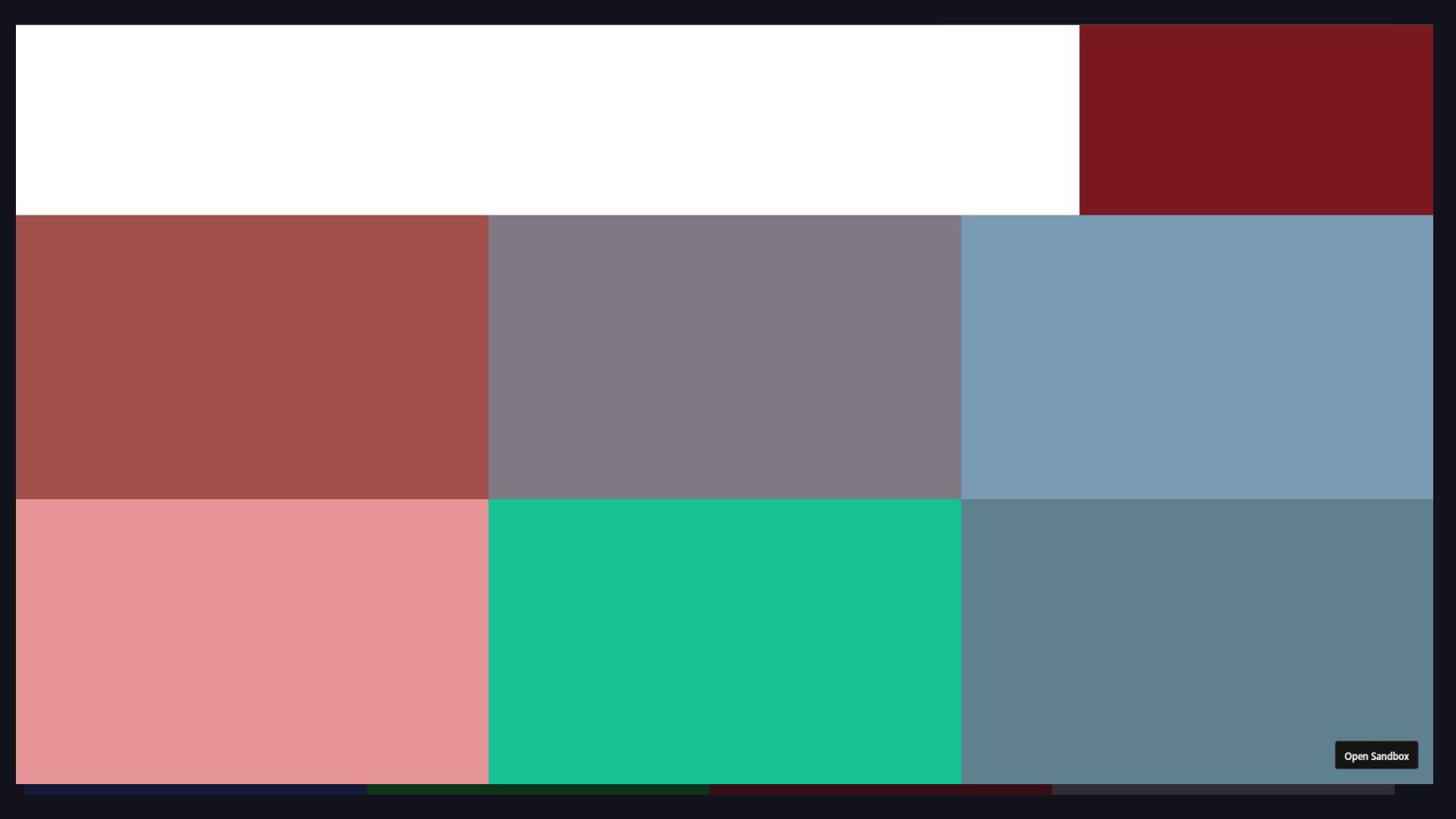


BUILD

create your desktop divs using html and css then make it responsive



PHASE 5



An 800x800 pixel image featuring a 4x4 grid of colored squares. The colors are: top-left (white), top-right (dark red), middle-left (brownish-orange), middle-center (grey), middle-right (light blue), bottom-left (pink), bottom-center (bright green), and bottom-right (teal). A thin black border surrounds the entire grid.

Open Sandbox

A screenshot of the Chrome DevTools Elements tab showing the DOM structure and a detailed view of an element.

The DOM tree on the left shows:

```
<!DOCTYPE html>
<html lang="en">
  <head> ...
  <body>
    <div class="row1 grey"> ...
    <div class="row2 black">
      <div class="columns"> ...
        <div class="columns"> ...
          ...
            <div class="box3 color3"> ...
              <div class="box3 color4"> ...
                ...
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
    <div id="sb__open-sandbox65" style=" ...
      position: fixed;
      margin: 0;
      padding: 0;
      bottom: 16px;
      right: 16px;
      border: none;
      width: 118px;
      height: 36px;
      z-index: 9999999999999999;
    "> ...
  </body>
</html>
```

The element `div#sb__open-sandbox65` is highlighted in blue. A callout box below it points to the element in the DOM tree with the text "iframe#sb__open-sandbox65 118x36".

The right panel displays the Styles tab with the following CSS rules:

```
element.style {
}
@media only screen and (max-width: 800px) {
  .box3 {
    height: 100%;
    width: 50%;
  }
}
.color3 {
  background-color: #807983ff;
}
.box3 {
  height: 100%;
  width: 100%;
}
div {
  display: block;
  unicode-bidi: isolate;
}
```

A detailed element inspector on the far right shows the bounding box of the `div#sb__open-sandbox65` element, which is 360.500x227.750 pixels in size. The element has a light orange background and a dark orange border.

Screenshot of the Chrome DevTools Elements tab showing the DOM structure and a detailed view of an iframe.

DOM Tree:

```
<!DOCTYPE html>
<html lang="en"> scroll
  <head> </head>
  <body>
    <div class="row1 grey"> flex
      <div class="box1 white"></div> == $0
      <div class="box2 red"></div>
    </div>
    <div class="row2 black"> </div>
    <iframe id="sb_open-sandbox62" style="<br/>
      position: fixed;<br/>
      margin: 0;<br/>
      padding: 0;<br/>
      bottom: 16px;<br/>
      right: 16px;<br/>
      border: none;<br/>
      width: 118px;<br/>
      height: 36px;<br/>
      z-index: 999999999999;<br/>
    "> </div>
  </body>
</html>
```

Element Inspector:

The element `iframe#sb_open-sandbox62` is selected. The status bar shows its dimensions as `118x36`. A tooltip labeled "Open Sandbox" is visible below the element.

Styles Panel:

The Styles panel displays the following CSS rules:

```
element.style {
}
@media only screen and (max-width: 800px) {
  .box1 {
    height: 25%;
    width: 100%;
  }
}
.white {
  background-color: #ffffff;
}
.box1 {
  height: 100%;
  width: 75%;
}
div {
  display: block;
  unicode-bidi: isolate;
}
```

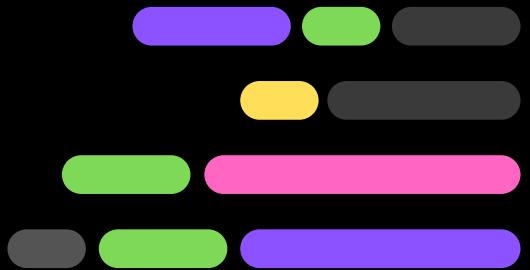
Computed:

The Computed tab shows the final computed styles for the selected element. A visual representation of the element's bounding box is shown with a dashed border and a size of `472x56.938`.



CONTENT

insert all images and text into your site



PHASE 7

GUILTY GEAR™

— STRIVE —



Open Sandbox

GUILTY GEAR™

— S T R I V E —



Elements Console Sources Network Performance Memory Application Security Lighthouse Recorder Performance insights ▾

Styles Computed Layout Event Listeners >

Filter .cls +

element.style { }

@media only screen and (max-width: 800px) {

.box3 { height: 100%; width: 50%; }

.color3 { background-color: #807983ff; }

.center-content { justify-content: center; align-items: center; display: flex; }

.box3 { height: 50%; width: 100%; }

div { user agent stylesheet }

display: block; unicode-bidi: isolate;

margin border padding - 373.500x227.750 -

```
<!DOCTYPE html>
<html lang="en">
  <head> </head>
  <body>
    <div class="row1 grey"></div> (flex)
    <div class="row2 black">
      <div class="columns"></div> (flex)
      <div class="columns"> (flex)
        <div class="box3 color3 center-content"></div> flex == $0
        <div class="box3 color4 center-content"></div> flex
      </div>
      <div class="columns"></div> (flex)
      <script crossorigin type="text/javascript" src="https://codesandbox.io/static/js/watermark-button.eeb14a97b.js"></script>
      <script type="text/javascript" src="https://kt7n3z.csb.app/83e09c4a/loader.js?fn=inject.js"></script>
    </div>
    <div id="sb__open-sandbox26" style="<br/> position: fixed; margin: 0; padding: 0; bottom: 16px; right: 16px; border: none; width: 118px; height: 36px; z-index: 999999999999;"></div>
  </body>
</html>
```

html body div.row2.black div.columns div.box3.color3.center-content

GUILTY GEAR™ — STRIVE —



Elements Console Sources Network Performance Memory Application Security Lighthouse Recorder Performance insights ▾

```
<!DOCTYPE html>
<html lang="en"> scroll
  > <head> ... </head>
  > <body>
    > <div class="row1 grey"> ... </div> flex
    > <div class="row2 black">
      > <div class="columns">
        > <div class="box3 color1 center-content"> flex
          ...
            > <img class="sol"> == $0
          </div>
        > <div class="box3 color2 center-content"> ... </div> flex
        </div>
      > <div class="columns">
        > <div class="box3 color3 center-content"> ... </div> flex
        > <div class="box3 color4 center-content"> ... </div> flex
      </div>
    > <div class="columns"> ... </div>
    > <script crossorigin="use-strict" type="text/javascript" src="https://codesandbox.io/static/js/watermark-button.eeb14a97b.js"></script>
    > <script type="text/javascript" src="https://kt7n3z.csb.app/83e09c4.../loader.js?fn=inject.js"></script>
  </div>
  > <iframe id="sb_open-sandbox26" style="<br/>
    position: fixed;<br/>
    margin: 0;<br/>
    padding: 0;<br/>
    bottom: 16px;<br/>
    right: 16px;<br/>
    border: none;<br/>
    width: 118px;<br/>
    height: 36px;<br/>
    z-index: 9999999999999999;<br/>
  "> ... </iframe>
</body>
</html>
```

Styles Computed Layout Event Listeners >

Filter element.style { }

```
.sol {
  content: url(images/Sol.PNG);
  object-fit: cover;
  height: 75%;
  width: 75%;
}

img {
  user agent stylesheet
  overflow-clip-margin: content-box;
  overflow: clip;
}
```



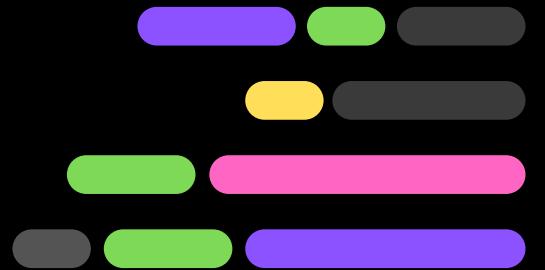
html body div.row2.black div.columns div.box3.color1.center-content img.sol

html body div.info div#description.description.fuchsia div#text1.text.fuchsia.center-content



JAVASCRIPT

add behavior to your page by inserting two pieces of javascript



PHASE B

GUILTY GEAR

— STRIVE —

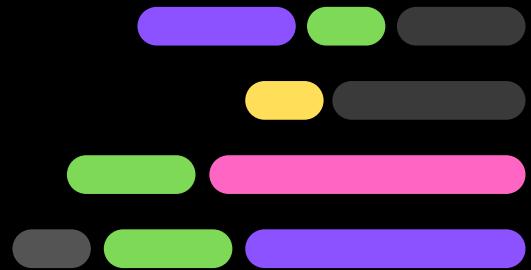


Open Sandbox



ENHANCE UX

add in effects when the user hovers and clicks to make the page intuitive and fun



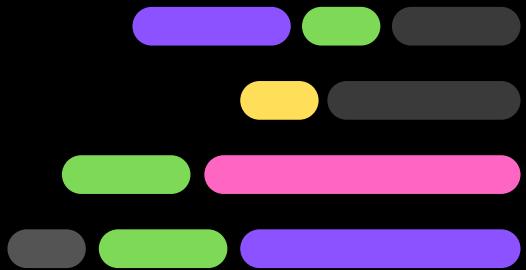
PHASE 9





TEST + ITERATE

get feedback from users then make revisions to your user interface



PHASE 10

GUILTY GEAR

- STRIVE -

