

Table of Contents

Overview of Scrollytelling Project	2
Steps to take	2
Lottie Animations	2
Bodymovin	2
JavaScript Libraries	3
Useful Links for Beginners to Web Development	4
HTML&CSS	5
JavaScript In-Depth	6
Reusable JavaScript Parameters	8

Overview of Scrollytelling Project

Steps to take

1. Animate in Adobe After Effects
2. Install the bodymovin add on
3. Convert animation to json file
4. Drag and drop it in lottie website to view the animation
5. Use code to animate it on scroll.

Lottie Animations

- 01_Cover - <https://lottiefiles.com/share/rdgseik8>
- 01_Cover without Logo - <https://lottiefiles.com/share/jgitxsov>
- 02_Gov - <https://lottiefiles.com/share/eqravj6k>
- 03_Gov - <https://lottiefiles.com/share/zr2ommfg>
- 04_Gov - <https://lottiefiles.com/share/zqoxvx9n>
- 05_Key Loved Ones - <https://lottiefiles.com/share/dftwvmpm>
- 06_Key Loved Ones - <https://lottiefiles.com/share/4pf2baej>
- 07_Cover - <https://lottiefiles.com/share/vhejrh15>
- 08_Gov and Loved Ones - <https://lottiefiles.com/share/tht2czrc>
- 09_Cover - <https://lottiefiles.com/share/zi08qyw5>

Bodymovin



1. Bodymovin is an Adobe After Effects extension. It is used to export Adobe After Effects animations as .json files. The download link is in here <https://aescripts.com/bodymovin/>.
2. Not all Adobe After Effects features are supported. The supported features are as listed in the link above.

JavaScript Libraries



1. Lottie-web is used to load the .json files onto the webpage.
2. ScrollTrigger by GSAP is used to sync the .json files (the animations) to the user's scroll, as well as to set different parameters to tweak the scrollytelling experience.
3. The links to the respective libraries are as follows.
 - a. Lottie-Web: <https://github.com/airbnb/lottie-web/wiki/loadAnimation-options>
 - b. ScrollTrigger: <https://greensock.com/docs/v3/Plugins/ScrollTrigger>
4. The code that I used for scrollytelling is a combination of both lottie-web and scrolltrigger. Most of the code that I have used came from here. <https://github.com/chrisgannon/ScrollLottie>. I will explain the codes more in depth later on in this guide.

Useful Links for Beginners to Web Development

1. Knowing basic HTML, CSS and JavaScript will help greatly.
2. These are the two crash courses I feel is good for beginners to web development.
 - a. HMTL&CSS:
https://www.youtube.com/playlist?list=PL4cUxeGkcC9ivBf_eKCPIAYXWzLIPAm6G
 - b. JavaScript:
https://www.youtube.com/playlist?list=PL4cUxeGkcC9haFPT7J25Q9GRB_ZkFrQAc
 - c. You do not need to watch all the videos, just watch to the point you feel that you understand enough to continue on with this project.

HTML&CSS

I used divisions (div) to hold different animations.

```
<div class="loader">  
    
</div>
```

The loader class is used to house the loading placeholder. This gif I used will need to be replaced by cpf's current placeholder.

```
<div class="key government-2"></div>  
<div class="pie-chart government-3"></div>  
  
<div class="cover loved-ones-cover-4"></div>  
<div class="key loved-ones-5"></div>  
<div class="pie-chart loved-ones-6"></div>
```

The rest of the divisions holds the different animations. Classes like government-2 and loved-ones-5 need to be created to be referred to later. Classes like key and pie-chart are being reused.

JavaScript In-Depth

```
11 //loading screen with a 0.25 seconds delay after for first animation to not look laggy
12 window.addEventListener("load", () => {
13     loader.style.display = "none";
14     html.style.overflowY = "visible";
15     setTimeout(() => {
16         const titlePage = lottie.loadAnimation({
17             container: document.querySelector(".government-cover-1"), // the dom element that
18             // will contain the animation
19             renderer: 'svg',
20             loop: 0,
21             autoplay: true, //it is true here as the designers want the first animation to be
22             // auto played
23             path: "./animations/01_Cover.json", // the path to the animation json
24             rendererSettings: {
25                 preserveAspectRatio: 'xMidYMid slice'
26             }
27         });
28         titlePage.setSpeed(0.8); //1 is the current speed;
29     }, 250);
30 });
```

This portion houses the code for the loading placeholder, as well as for the first animation to be played. What the code does is once the JavaScript is loaded, I will set the placeholder class display to be none, and start the first animation.

```

31  /////LOTTIE ANIMATION FUNC
32  const ScrollLottie = (obj) => { //https://github.com/chrisgannon/ScrollLottie
33
34      let anim = lottie.loadAnimation({ //https://github.com/airbnb/lottie-web/wiki/
loadAnimation-options
35          container: document.querySelector(obj.target), //div to place the animation in
36          renderer: 'svg', //render the json file as an svg
37          loop: false,
38          autoplay: false,
39          path: obj.path, // the json file path
40          rendererSettings: { //https://github.com/airbnb/lottie-web/wiki/Renderer-Settings
preserveAspectRatio: obj.aspectRatio} //keep aspect ratio of svg
41      });
42
43
44      let timeObj = {currentFrame: 0}
45      ScrollTrigger.create({ //https://greensock.com/docs/v3/Plugins/ScrollTrigger
46          trigger: obj.target,
47          scrub: true,
48          pin: obj.pin,
49          start: obj.start,
50          end: obj.end,
51          onUpdate: self => { //whenever the user scrolls, the code calculates which lottie
frame to show
52              if(obj.duration) {
53                  gsap.to(timeObj, {
54                      duration: obj.duration,
55                      currentFrame: (Math.floor(self.progress * obj.totalFrames) + obj.startingFrame),
56                      onUpdate: () => {
57                          anim.goToAndStop(timeObj.currentFrame, true)
58                      },
59                      ease: 'power1.out'
60                  })
61              } else {
62                  anim.goToAndStop(self.progress * obj.totalFrames + obj.startingFrame, true)
63              }
64          }
65      });
66

```

Lottie.loadAnimation

ScrollTrigger.create

This is the code for the loading of lottie animations onto the webpage. It contains two functions from lottie-web and ScrollTrigger. lottie.loadAnimation (from lottie-web) is used to load lottie animations (json files) onto the webpage. and ScrollTrigger.create (from ScrollTrigger) allows us to set up different parameters for the scrollytelling experience.

The in-depth explanation of these codes can be found here.

- a. Lottie-Web: <https://github.com/airbnb/lottie-web/wiki/loadAnimation-options>
- b. ScrollTrigger: <https://greensock.com/docs/v3/Plugins/ScrollTrigger>

Reusable JavaScript Parameters

```
//02_Gov
ScrollLottie({
  target: ".government-2", //the division the lottie animation will be in
  path: "./animations/02_Gov.json", //downloaded json file
  duration: 1, //adds smooth scrolling, can set any number
  end: "+=1200%", //how many percent of the viewheight a user has to scroll to finish
  the animation
  totalFrames: 708, //get this from the website. essentially the total frames for the
  whole animation
  start: "top 99%", //where trigger starts
  pin: false,
  startingFrame: 15, //usually 0, unless we want animation to start from a later frame
  in this case
  aspectRatio : 'xMidYMin meet' //toggle svg aspect ratio setting
});

//trigger for div to be pinned, has to be seperated from above ScrollLottie function as I
want the animation to play at a different instance compared to when i want to pin it...
ScrollTrigger.create({
  trigger: ".government-2",
  start: "top top",
  end: "1000%",
  pin: true,
});
```

This is the reusable portion of the code. Whenever I add in a new animation, I will have to do so in this format. I will call the ScrollLottie function, and then call the ScrollTrigger.create function.


```

ScrollLottie({
  target: ".government-2", //the division the lottie animation will be in
  path: "./animations/02_Gov.json", //downloaded json file
  duration: 1, //adds smooth scrolling, can set any number
  end: "+=1200%", //how many percent of the viewheight a user has to scroll to finish
  the animation
  totalFrames: 708, //get this from the website. essentially the total frames for the
  whole animation
  start: "top 99%", //where trigger starts
  pin: false,
  startingFrame: 15, //usually 0, unless we want animation to start from a later frame
  in this case
  aspectRatio : 'xMidYMin meet' //toggle svg aspect ratio setting
});

//trigger for div to be pinned, has to be seperated from above ScrollLottie function as I
want the animation to play at a different instance compared to when i want to pin it...
ScrollTrigger.create({
  trigger: ".government-2",
  start: "top top",
  end: "1000%",
  pin: true,
});

```

This is the ScrollLottie function. For each new animations, I just need to call ScrollLottie, which is a function already created above.

1. target: the division the lottie animation will be in
2. path: the location of the json file
3. duration: adds smooth scrolling, can be set to any number
4. end: how many percent of the viewheight the user must scroll to finish an animation
5. totalFrames: the total frames of the lottie animation
6. start: where the scrolltrigger starts
(<https://greensock.com/docs/v3/Plugins/ScrollTrigger/start>)
7. pin: determine if we want to pin the div when scrolltrigger starts
8. aspectRatio: toggle the svg aspect ratio (<https://developer.mozilla.org/en-US/docs/Web/SVG/Attribute/preserveAspectRatio>)

```

//trigger for div to be pinned, has to be seperated from above ScrollLottie function as I
want the animation to play at a different instance compared to when i want to pin it...
ScrollTrigger.create({
  trigger: ".government-2",
  start: "top top",
  end: "1000%",
  pin: true,
});

```

This is the trigger for pinning the division. I have this as I want the animation to be played early, but I only want the division to be pinned only when it resides in 100% of my viewport.