

OBI-HWAN STORY 소개

개인 과제 - 블로그 만들기

st06 김환희

목차

1. 기획

- 1.1 주제 - 스타워즈 p1
- 1.2 자료 수집 p2
- 1.3 블로그 제목 p4
- 1.4 메인 색상과 글꼴 p4

2. 스케치 및 드로잉

- 2.1 스케치 p5
- 2.2 드로잉 p6

3. 구현

- 3-1 메뉴바 p8
- 3-2 타이틀 줌아웃 p9
- 3-3 스토리 자동 스크롤 p10
- 3-4 행성 슬라이드 p11

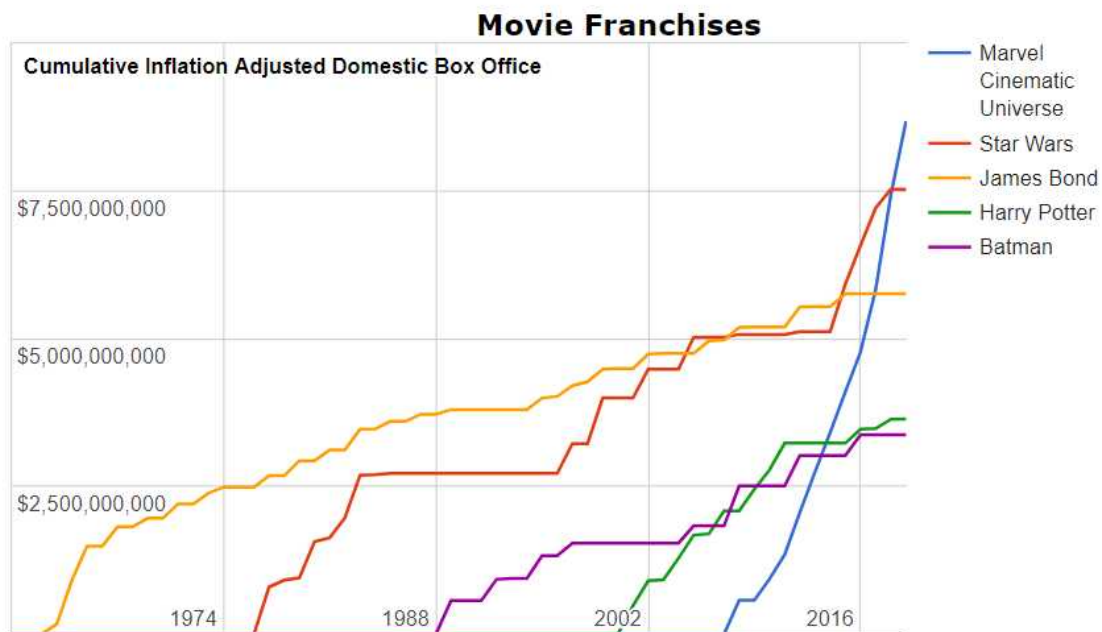
4. 구현 결과

- 4-1 구현 결과 p13

1. 기획

1.1 주제 - 스타워즈

스타워즈는 1970년대 대중문화의 아이콘이자 미국을 상징하는 할리우드 영화의 대표 주자였으며, 생산지인 미국을 넘어 세계적인 문화코드로 자리매김하여 아직까지도 전 세계에서 사랑받는 엔터테인먼트 상품의 상징 그 자체입니다. 50년이 지난 현재까지도(2019년) 전 세계 프랜차이즈 영화 박스오피스를 놓고 보면 93억 달러에 이르는 전 세계 2위의 흥행 수익을 기록하고 있습니다.¹⁾



전 세계 박스오피스 흥행 수익

스타워즈 시리즈가 여타 다른 블록버스터 프랜차이즈보다 좀 더 특별하게 취급되는 이유는 시리즈의 기나긴 역사뿐만 아니라 스타워즈라는 프랜차이즈의 뿌리 때문이기도 합니다. 가령 위에서 보시는 바와 같이 프랜차이즈 역대 흥행성적으로 순위권에 있는 해리포터 시리즈, 007 시리즈, 반지의 제왕 실사영화 시리즈 등은 그 기원이 전 부 소설에서 비롯되었으며, 21세기의 스타워즈라 불리고 있는 마블 시네마틱 유니버스를 포함한 슈퍼히어로 영화들은 그 기원이 코믹스에서 비롯되었습니다. 즉, 최고의 흥행을 거둔 영화 프랜차이즈들 중에서 그 시작이 영화라는 매체에서부터 출발한 건 스타워즈 시리즈 하나뿐이라는 것입니다. 따라서 스타워즈는 단순히 가장 유명한 영화 시리즈임을 넘어서 영화라는 상상력의 산물, 그 자체를 대변하는 브랜드라고 봐야 할 것입니다.

이를 바탕으로 저는 개인 블로그 컨셉을 스타워즈를 오마주하여 제작하기로 하였습니다

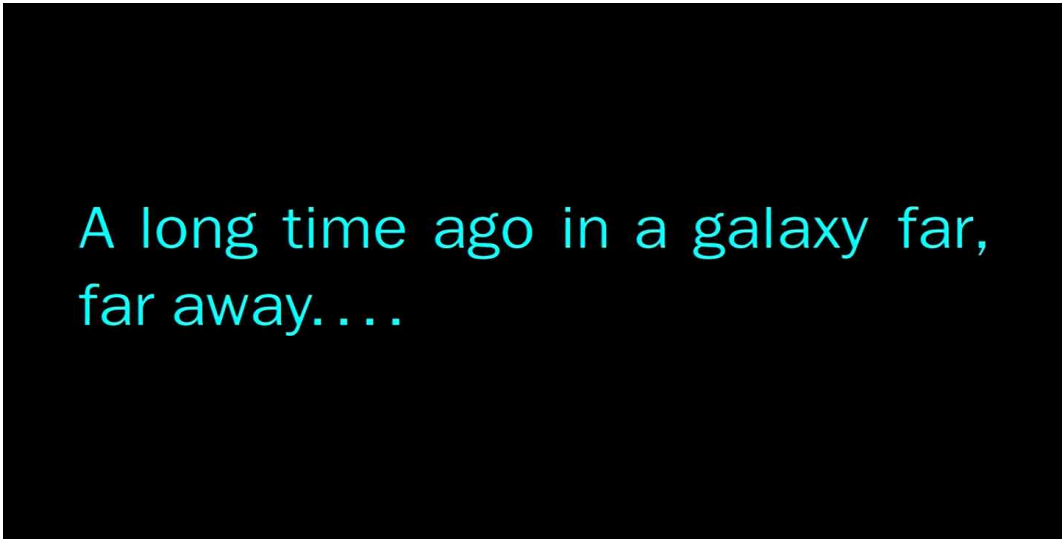
1) 출처: [THE NUMBERS](#)

니다.

1.2 자료 수집

스타워즈의 전통²⁾은 아래와 같습니다.

1) 음악으로는 존 윌리엄스가 작곡한 스타워즈의 메인 테마가 흐른다. 어떤 시리즈든지 처음에 나오는 음악은 바뀌지 않는다.



A long time ago in a galaxy far,
far away....

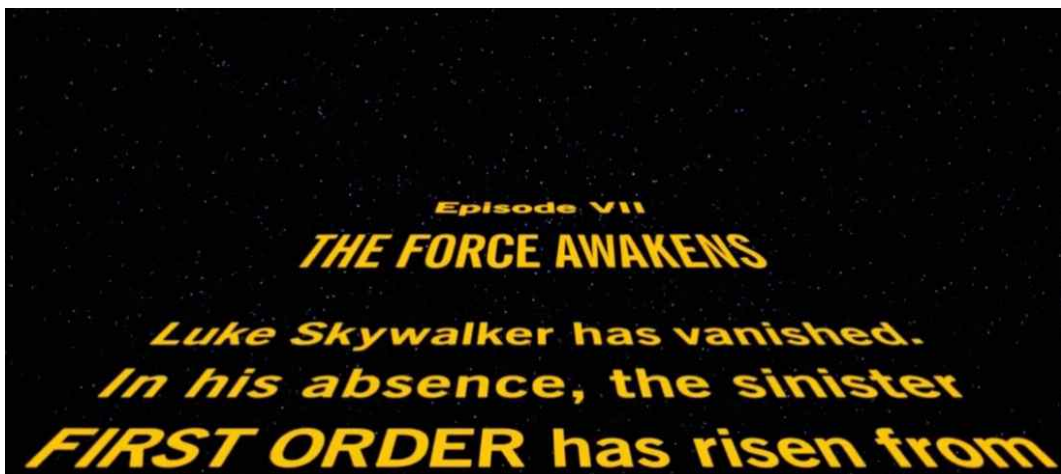
2) 파란색으로 "오래 전 멀고 먼 은하계에...(A long time ago in a galaxy far, far away...)"이라는 내레이션이 뜬다.

지켜야 할 것은 점은 4개(말줄임표+마침표로 이해 가능), 글자 색은 하늘색이어야 한다. 또한 galaxy far, 부분에서 줄이 넘어가야 한다.

2) 출처: 나무위키 [스타워즈/전통](#)



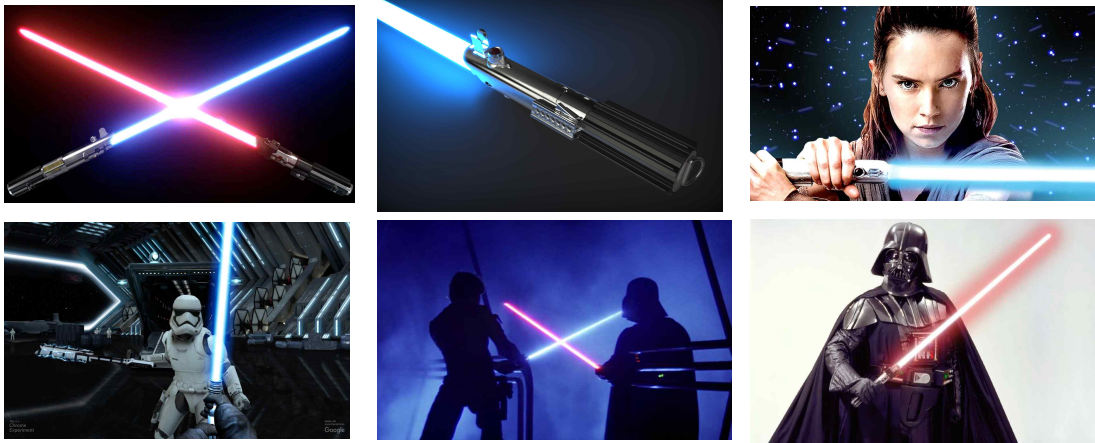
3) 우주를 배경으로 STAR WARS 로고가 뜨며 음악이 나온다. 글자 색은 노란색이며, 외곽선만 그려져 있어야 한다. 또한 로고가 줌아웃으로 나와야 한다.



4) 로고가 사라지면서 아래에서부터 인트로 자막이 올라온다. 노란색 글자가 사다리꼴 모양으로 올라와야 한다. 아래는 넓고 위로 올라갈수록 서서히 사라지는 식의 구조다. 사실 사다리꼴이라기보다 자막이 우주 공간에 입체적으로 펼쳐지는 것을 2D 화면에 보여주는 방식이라고 할 수 있다.

5) 자막이 다 올라오면 카메라가 아래로 내려가면서 행성과 우주선이 보인다.

또한, 스타워즈에서 빠질 수 없는 것이 바로 광선 검입니다.



아래 사진은 작중 캐릭터인 오비완의 광선 검으로 파란 빛의 광선 검을 사용합니다.



위와 같이 조사한 자료를 바탕으로 다음과 같이 디자인을 하였습니다.

1.3 블로그 제목

작중 캐릭터인 오비완(obi-wan)의 완(wan)과 제 이름의 환(hwan)에서 공통적인 부분을 차용하여 개인 블로그 이름을 OBI-HWAN STORY 라고 지었습니다.

1.4 메인 색상과 글꼴

#ffb401

rgb(255, 180, 1)

스타워즈의 메인 타이틀과 글꼴 색상은 노란색입니다. 영화에서는 채도가 높고 밝은 톤의 노란색을 사용했지만 저는 차분한 느낌을 원해서 약간 툇다운 되고 진한 노란색을 선택했습니다.

“안녕하세요.” 느낌표!
물음표? {한글서체}기
[검은고딕] <제스타입>
(타입디자인) 더하기+
빼기- 세번째:폰트 *
#해시태그, @글뱅이.

2. 스케치 및 드로잉

HOME PROFILE PORTFOLIO

OBI-HWAN
MAY THE FORCE BE WITH YOU
STORY

Episode I
LET ME INTRODUCE MYSELF

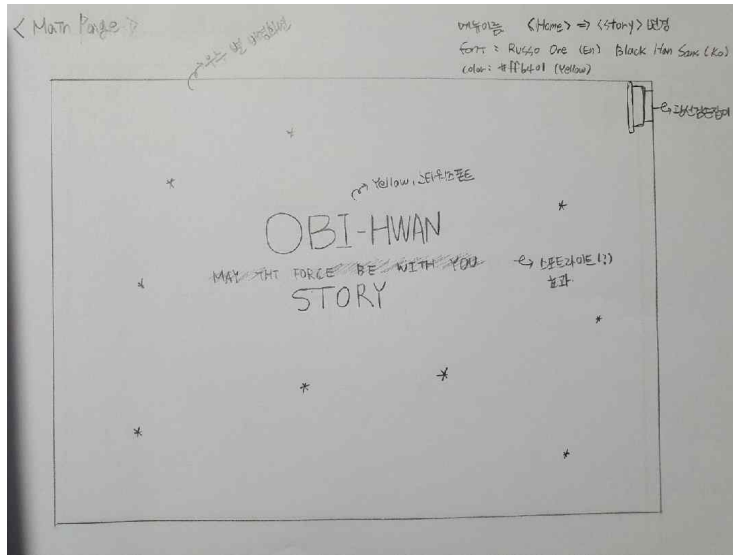
KIM HWAN-HEE
1989. 10. 03.

HTML
CSS
JavaScript
PHP/PHP

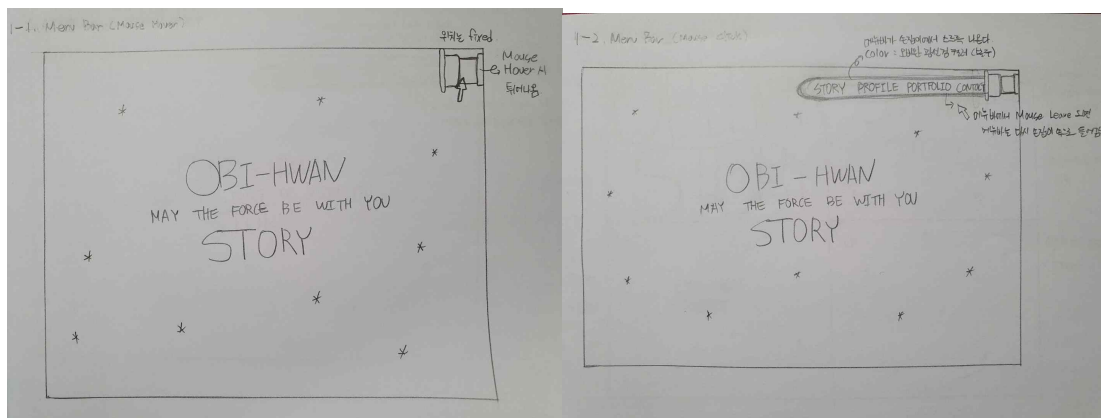
Portfolio

main color: Black / Green
font color: Yellow / White

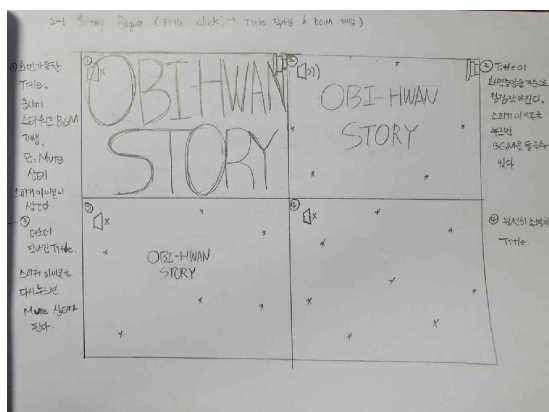
2.2 드로잉



< 메인 페이지 >

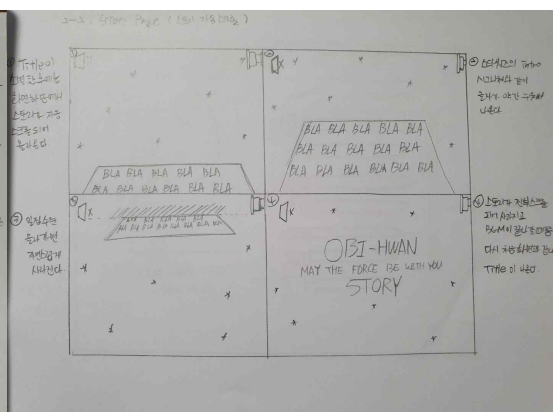


< 메뉴바 - 호버 >

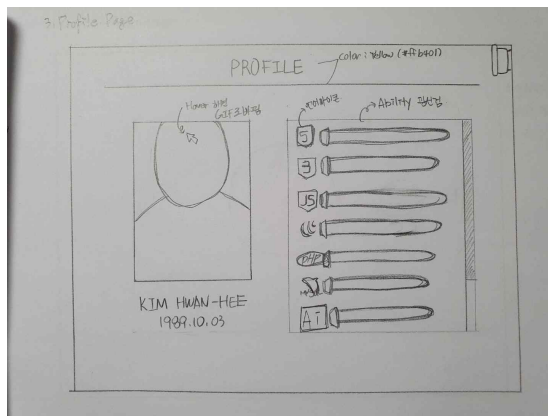


<스토리 메뉴 - 타이틀 클릭>

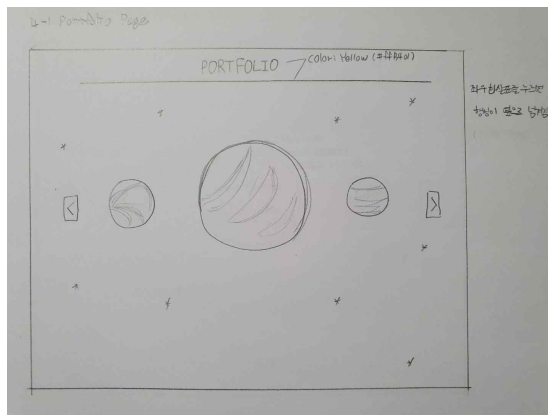
< 메뉴바 - 클릭 >



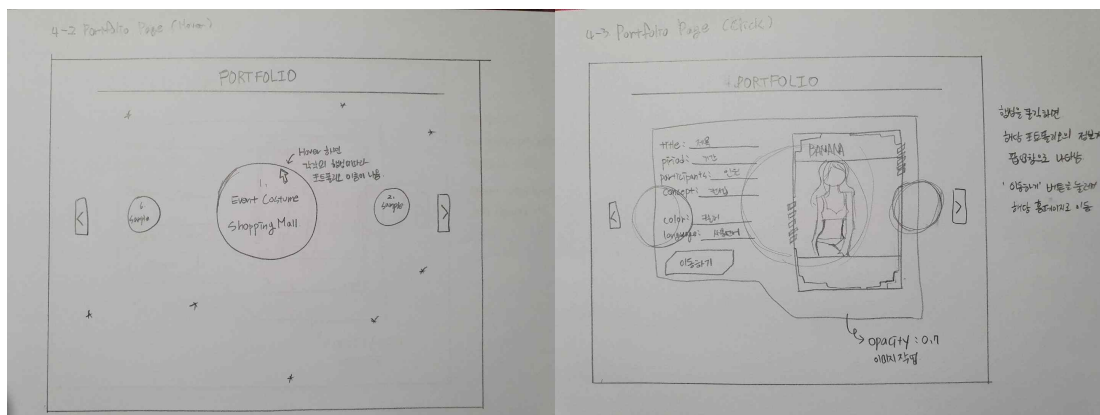
<스토리 메뉴 - 스토리 자동 스크롤>



<프로필 메뉴>



<포트폴리오 메뉴>



<포트폴리오 메뉴 - 호버>

<포트폴리오 메뉴 - 클릭>

3. 구현

3-1 메뉴바

```
1 function menuMotion(number){
2     switch (number){
3         case 0: if(menuBar.style.right!="0px"){rightValue+=5}; setTimeout("menuMotion(0)",0); break;
4         case 1: if(menuBar.style.right!="-610px"){rightValue-=5}; setTimeout("menuMotion(1)",0);break;
5     }
6     menuBar.style.right= rightValue+"px";
7     if(menuBar.style.right=="0px")
8     {
9         clearTimeout(longgerMotion);
10    }
11    if(menuBar.style.right=="-610px")
12    {
13        clearTimeout(longgerMotion);
14    }
15 }
16 menuBox.addEventListener("click",function(){
17     longgerMotion= setTimeout("menuMotion(0)",0);
18 });
19 menuBar.addEventListener("mouseleave",function(){
20     longgerMotion= setTimeout("menuMotion(1)",0);
21 });
22 }
```

광선검의 손잡이 부분을 click하면 메뉴가 나오고, 메뉴바에서 mouseleave 되면 메뉴가 들어가는 것을 구현하기 위해, 광선검의 right 값을 증가시키는 경우와, 감소시키는 경우의 함수를 만들어서 이벤트 발생 시, 해당 경우의 값을 매개변수로 값을 넘겨 원하는 효과의 함수를 실행시켜 구현하였습니다.



< click >



< mouseleave >

3-2 타이틀 줌아웃

```
1  function introMotion(){
2      titleText.style.display="none";
3      shineText.style.display="none";
4      subTitleText.style.display="block";
5      starwarsBGM.play();
6      musicMute.style.display="block";
7      smallMotion=setInterval("titleMotion()",80);
8      timeStemp= new Date().getTime();//재생시작 타임
9      stopTime = timeStemp+60000;
10     //console.log(timeStemp);
11     //console.log(stopTime);
12 }
13 function titleMotion(){
14     widthSize= widthSize-16;
15     heightSize=heightSize-9;
16     subTitleText.style.backgroundColor = widthSize+"px "+ heightSize+"px";
17     if(widthSize==0)
18     {
19         clearInterval(smallMotion);
20         widthSize = 1920;
21         heightSize = 1080;
22         subTitleText.style.display="none";
23         autoscrollStory();
24     }
25 }
```

Colored by Color Scriptor

화면 전체 꽉 차는 사이즈인 width값 1920px, height값 1080px에서 해당 비율인 16:9 만큼 감소하는 함수를 만들어서 setInterval 함수로 실행 시켜 타이틀이 계속 감소하는 효과를 구현하였습니다.

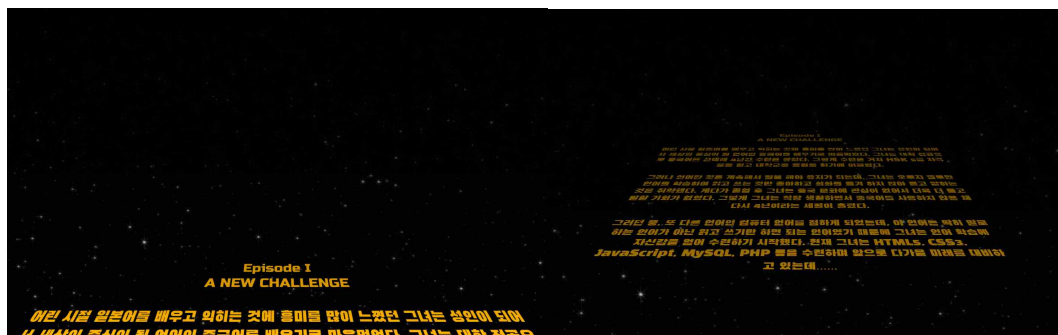


3-3 스토리 자동 스크롤

```
1 .descStoryAni{
2     position:absolute;
3     width: 1500px;
4     top: 1020px;
5     left: 50%;
6     right: 0px;
7     bottom: 0px;
8     margin-left: -800px;
9     padding: 50px;
10    overflow: hidden;
11    font-family: 'Black Han Sans', sans-serif;
12    font-size: 50px;
13    color:#ffb401;
14    text-align: justify;
15    /*border: 1px solid white;*/
16    transform-origin: 50% 100%;
17    transform: perspective(300px) rotateX(15deg);
18    animation: autoscroll 50s linear;
19 }
20 @keyframes autoscroll{
21     to{
22         margin-top:-3100px;
23     }
24 }
25
```

```
function autoscrollStory(){
1     descStory.setAttribute("class","descStoryAni");
2     runningTime=setInterval("endTime()",100);//러닝타임
3
4 }
```

top값을 많이 주어 화면 밑으로 안보이게 만든 후, 애니메이션을 통해 점점 위로 올라오도록 합니다. 다만, 처음부터 애니메이션이 있으면 홈페이지 접속하자마자 애니메이션이 실행되므로, JavaScript를 통해 원하는 타이밍에 애니메이션이 기재된 class를 추가하여 애니메이션이 실행되도록 구현했습니다.



3-4 행성 슬라이드

```
1  this.onclick=function(e){
2      if(e.target.id=="leftArrow" || e.target.id=="rightArrow"){
3          if(e.target.id=="leftArrow"){ arrowClick--;}
4          if(e.target.id=="rightArrow"){ arrowClick++;}
5          if(arrowClick==0){ arrowClick=arrowClick+6} //6 개의 샘플
6          if(arrowClick>6){ arrowClick=(arrowClick%6)}
7          console.log(arrowClick);
8
9          switch(arrowClick){
10             case 1:
11                 switch(e.target.id){
12                     case "leftArrow":
13                         planet1.classList.remove("planet1Ani");
14                         planet1.classList.remove("planet3AniReverse");
15                         planet1.classList.add("planet2AniReverse");
16
17                         planet1.style.animationFillMode="both";
18                         planet2.classList.remove("planet2Ani");
19                         planet2.classList.remove("planet1AniReverse");
20                         planet2.classList.add("planet3AniReverse");
21                         planet2.style.backgroundImage=pathArray[0];
22
23                         planet2.style.animationFillMode="both";
24                         planet3.classList.remove("planet3Ani");
25                         planet3.classList.remove("planet2AniReverse");
26                         planet3.classList.add("planet1AniReverse");
27
28                         planet3.style.animationFillMode="both";
29                         Portfolio6.displayOnPlanet();
30                         Portfolio6.addAnimation("S");
31                         Portfolio1.addAnimation("N");
32                         Portfolio2.addAnimation("S");
33                     break;
34                     case "rightArrow":
35                         planet1.classList.remove("planet1AniReverse");
36                         planet1.classList.remove("planet2Ani");
37                         planet1.classList.add("planet3Ani");
38
39                         planet1.style.animationFillMode="both";
40                         planet2.classList.remove("planet2AniReverse");
41                         planet2.classList.remove("planet3Ani");
42                         planet2.classList.add("planet1Ani");
43
44                         planet2.style.animationFillMode="both";
45                         planet3.classList.remove("planet3AniReverse");
46                         planet3.classList.remove("planet1Ani");
47                         planet3.classList.add("planet2Ani");
48                         planet3.style.backgroundImage=pathArray[2];
49                         planet3.style.animationFillMode="both";
50                         Portfolio2.displayOnPlanet();
51                         Portfolio2.addAnimation("S");
52                         Portfolio6.addAnimation("S");
53                         Portfolio1.addAnimation("N");
54                     break;
55             } //6 가지의 case 중 1 개만 발췌
```

샘플로 6개의 행성으로 슬라이드를 만들었습니다. 화살표를 클릭했을 때마다 변수 arrowClick 값을 증감시켜 6개의 경우로 나누어 버튼 방향에 따라 행성의 위치를 변경하는 애니메이션 class를 부여하고, 행성 hover시 나타나는 포트폴리오 제목의 폰트 사이즈 변경을 적용하여 슬라이드를 구현하였습니다.

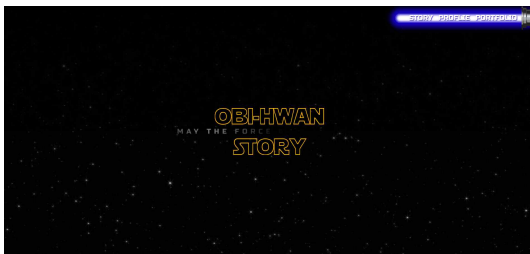


4. 구현 결과

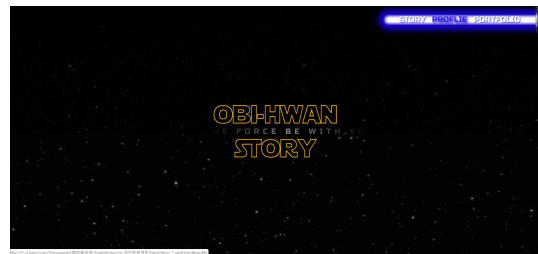
4.1 구현 결과



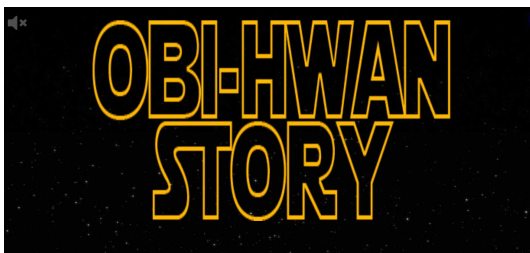
< 메인 - 스토리 메뉴 >



< 메뉴바 >



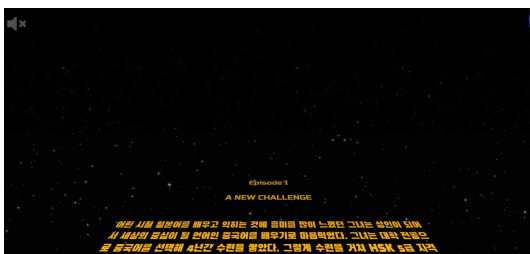
< 메뉴바 - 메뉴 hover >



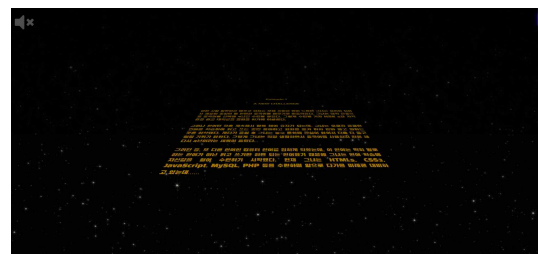
< 스토리 메뉴 - 1 >



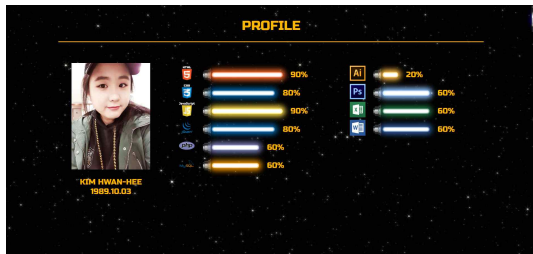
< 스토리 메뉴 - 2 >



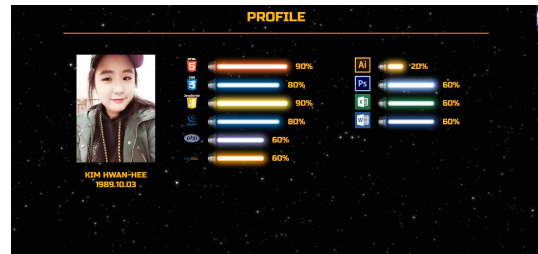
< 스토리 메뉴 - 3 >



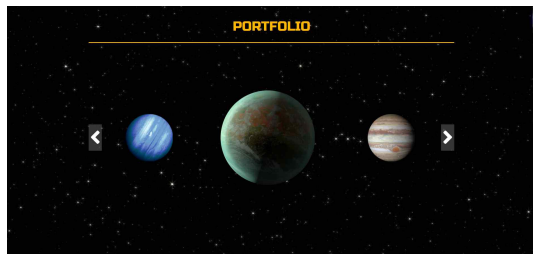
< 스토리 메뉴 - 4 >



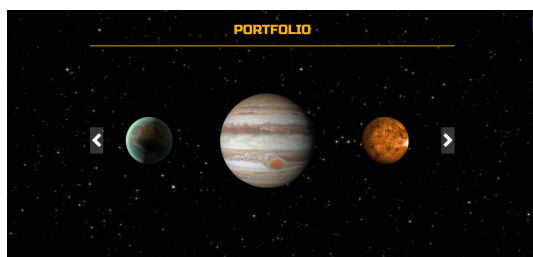
< 프로필 메뉴 >



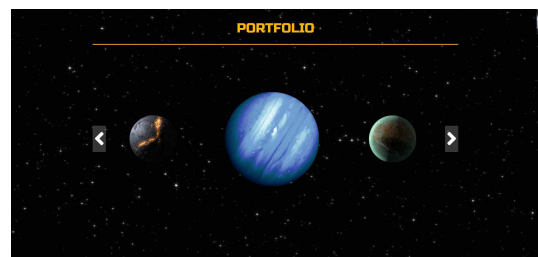
< 프로필 메뉴 - 사진 hover : GIF >



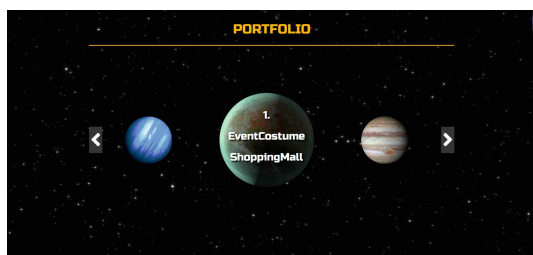
< 포트폴리오 메뉴 >



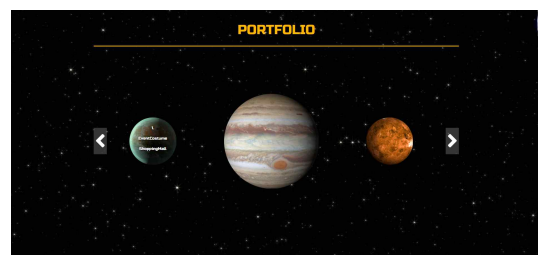
< 포트폴리오 메뉴 - 오른쪽 화살표 Click >



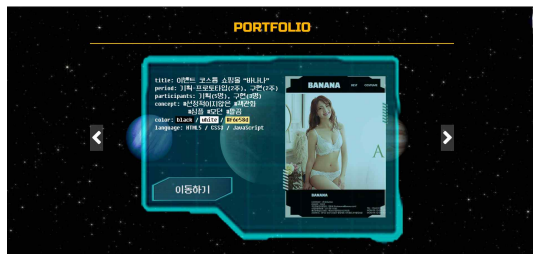
< 포트폴리오 메뉴 - 왼쪽 화살표 Click >



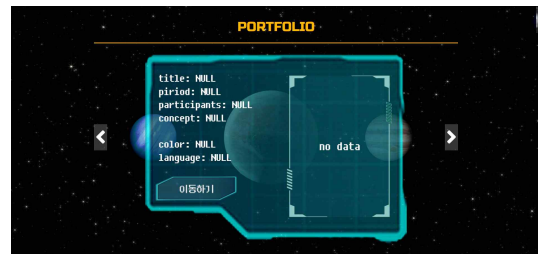
< 포트폴리오 메뉴 - hover 1 >



< 포트폴리오 메뉴 - hover 2 >



< 포트폴리오 메뉴 - Click 1 >



< 포트폴리오 메뉴 - Click 2 >