



Bose PJ

정윤철



목차

1. 브랜드 소개
2. 원본 홈페이지 탐구
3. 컴포넌트 구성
4. 디자인 포인트
5. 주요 기술(코드)
6. 향후 계획



브랜드 소개

1

글로벌 기업

Bose는 세계적으로 인정받는 **음향기기** 전문 기업입니다.

2

혁신적 기술

최첨단 음향 기술을 개발하고 적용합니다.

3

품질 중시

고품질 제품으로 소비자들의 신뢰를 얻고 있습니다.



원본 홈페이지 탐구

심플한 디자인

간결하고 깔끔한 홈페이지 구성으로 사용자 경험을 개선합니다.

유동적 정보

다양한 미디어 정보를 통해 제품에 대한 이해를 높입니다.

단순한 메뉴

주 카테고리를 메뉴로 구성하여 사용자가 정보를 찾기 편리합니다.

색상 변경 기능

상세 구매페이지 외에서도 제품 색상 변경이 가능합니다.

[1] 메인페이지

1 상단영역

2 메인 컴포넌트

1) 동영상 배너

2) 추천제품

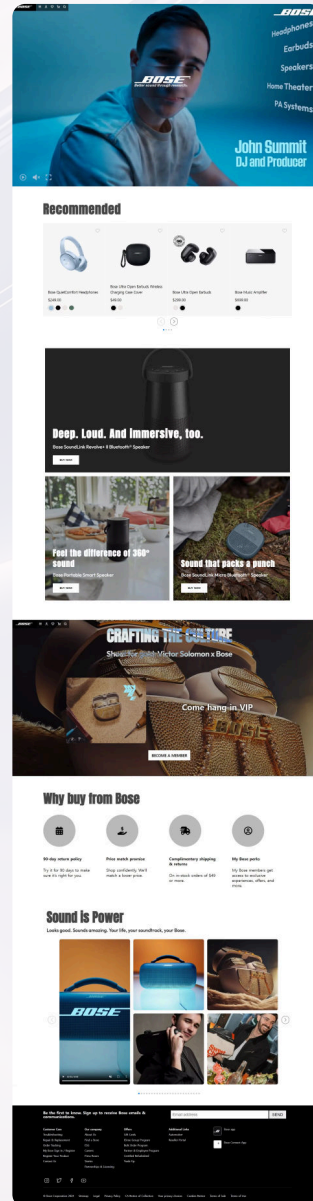
3) 서브배너

4) vip 커스텀

5) 구매이유

6) sns미디어

3 하단 영역



[2] 서브페이지

1 상단영역

2 메인 컴포넌트

1) 이미지 배너

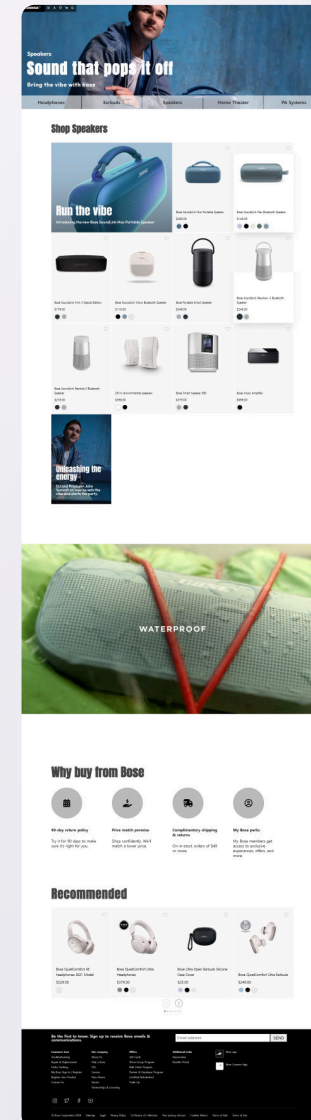
2) 카테고리별 제품들

3) 동영상

4) 구매이유

5) 추천제품

3 하단 영역



디자인

심플한 카테고리 *

복잡하지 않은 카테고리 구조로 사용자 편의성을 높입니다.

간결한 배치

심플한 배치로 전체적인 간결함을 유지합니다.

메인 배너

현재 페이지를 쉽게 알 수 있는 배너를 사용합니다.

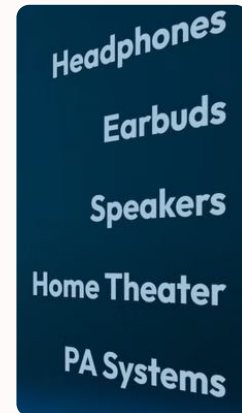
컨텐츠 크기

각 컨텐츠를 한눈에 볼 수 있도록 크기를 조정합니다.

기술 : 1. 상단 동적표현 구현

1) 로고배경 동영상 겹치기

- 작은 동영상박스 **width:20%**
- 안에있는 동영상 **width:500%**
- 이동효과 **position : sticky**



2) 서브메뉴 음향효과를 시각화표현

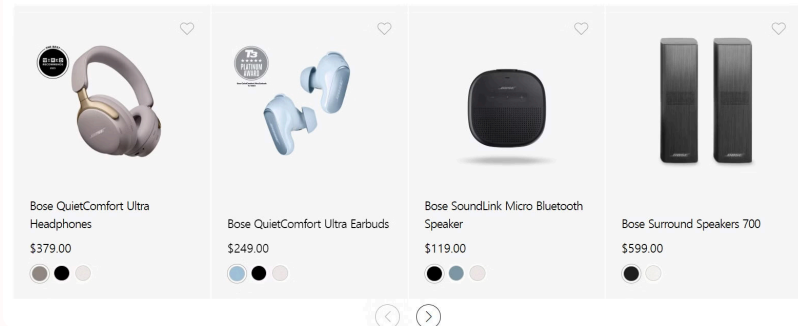
- 페이지 랜더링 시 측면에서 등장
- 원근감+ **Y축회전 -20%**
- 마우스 오버시

애니 효과(**scale & translate**) + 컬러반전 (**filter:invert(1)**)

```
@for $i from 1 to 6 {  
  // .tit-box 아래 자식들 .tit1 의 애니효과 지연  
  &:nth-child(#{6-$i}) .tit1 {  
    animation-delay: #{1 + 0.15 * $i}s;  
    //animation-delay: 1.15 ,1.3 ,1.45 ,1.6 .....  
    //  
  }  
}  
  
.tit1 {  
  transition: 0.3s;  
  animation: ani2 2s cubic-bezier(0.25, 1, 0.5, 1) forwards;  
  transform: rotateY(-90deg);  
  transform-origin: right;  
}
```

기술 : 2.랜덤함수 사용

Recommended



```
// 각 카테고리별로 1~2개의 제품 랜덤 선택
products.flatMap((v, i) => {
  // 1.각 카테고리별로 1~2개 제품 랜덤선택
  const rdProduct = getRdItems(v);
  return rdProduct.map((prod, j) => {
    // 2.선택된 제품에서의 가지고있는 색 랜덤선택
    const rdColor = getRdItem(prod.color);

    return (
      <SwipeSlide key={`$${i}-${j}`} className=
```

```
<img
  idx={prod.idx}
  src={`${process.env.PUBLIC_URL + prod.isrc}${
    prod.idx
  }/${rdColor}/0.webp`}
  alt={prod.name}
/>
```

```
{prod.color.map((clr, idx) => (
  <div key={idx}
    className={`color-circle-wrap ${
      clr == rdColor ? "on" : ""
    }`}
    onClick={(e) => colorFn(e, clr)}>
    <div
      className="color-circle"
      style={{ backgroundColor: colorList[clr] }}
      title={clr}/>
  </div>
```

랜더링시 제품이 바뀐

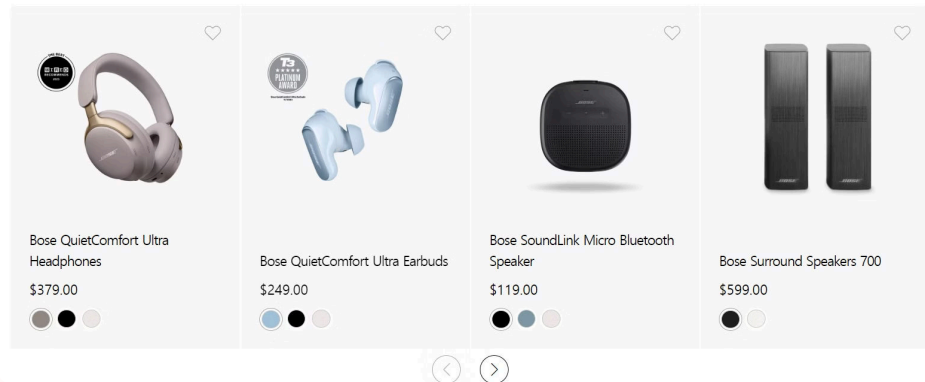
- 제품 랜덤출력
- 품목별 개수1~2 랜덤출력
- 품목내의 제품 랜덤
- 제품내에서 색상 랜덤

```
// 랜덤1 함수 //////////////////////////////////////
const getRdItem = (arr) => {
  const rdIdx = Math.floor(Math.random() * arr.length);
  return arr[rdIdx];
}; ////////////////////////////////// 랜덤1 함수 getRdItem //////////////////////////////////

// 랜덤2 함수 //////////////////////////////////////
const getRdItems = (arr) => {
  // 1~2개 랜덤넘버
  const rNum = Math.ceil(Math.random() * 2); // 1 또는 2
  // 선택된 제품을 저장: 1~2 를 선택해 배열로 담음
  // 원본 배열 복사 : 카테고리의 품목 배열로 복사!
  const catCopyArr = [...arr];
  // 선택된 아이템들을 배열로 담을 준비
  const items = [];
  for (let i = 0; i < rNum; i++) {
    // 1) catCopyArr 복사배열에서 무작위 아이템 선택
    const item = getRdItem(catCopyArr);
    // 2) 선택된 아이템을 items 빈배열에 추가
    items.push(item);
    // 3) 중복 방지를 위해 선택된 아이템 제거
    catCopyArr.splice(catCopyArr.indexOf(item), 1);
  }
  return items;
}; ////////////////////////////////// 랜덤2 함수 //////////////////////////////////
```


기술 : 2-1. 제품 색상 클릭 변경

Recommended



색상버튼 클릭시 **colorFn**함수적용

버튼 자신의 조상(**parents**)까지 올라가서 그아래(**prev**) 이미지를 **find()** 으로 찾음

이미지 경로 **src** 를 불러와 **split("/")** 으로 분할

분할된 배열데이터중 색상 부분[9] 를 클릭한 버튼 자신의 색상으로 교체 후 **join("/")** 으로 합침

최종 경로를 이전경로에 교체시킴

```
// 제품 색상 변경 함수
const colorFn = (e, clr) => {
  e.preventDefault();
  // ./images/speakers/1/white_smoke/0.webp
  // 0. 원조대상
  let org = $(e.currentTarget);
  // 1. 대상이미지선택
  let tg = org.parents(".rd-p-info").prev().find("img");
  // 2. 대상이미지 src값 읽기
  let isrc = tg.attr("src").split("/");
  console.log("전isrc:", JSON.stringify(isrc));
  // 3. 대상이미지 src값 변경
  isrc[9] = clr;
  console.log("후isrc:", isrc);
  // 4. 대상이미지경로 복원
  isrc = isrc.join("/");
  console.log("최종isrc:", isrc);
  // 5. 대상이미지 src 실제로 변경
  tg.attr("src", isrc);

  // 6. 현재 자신에게 클래스 "on" 넣기
  org.addClass("on").siblings().removeClass("on");
}; ////////////// colorFn ///////////////////
```

기술 : 2.랜덤함수를 위한 데이터

```
const speakers = [
  {
    idx: "0",
    name: "Bose SoundLink Max Portable Speaker",
    price: "$399.00",
    MainType: "Speakers",
    SubType: "Portable",
    color: ["blue_dusk", "black", ],
    isrc: `/images/speakers/`,
  },
  {
    idx: "1",
    name: "Bose SoundLink Flex Bluetooth Speaker",
    price: "$149.00 ",
    MainType: "Speakers",
    SubType: "Portable",
    color: [
      "chilled_lilac",
      "black",
      "white_smoke",
      "cypress_green",
      "stone_blue",
    ],
    isrc: `/images/speakers/`,
  },
];
```

```
export const products = [
  headphones,
  earbuds,
  speakers,
  homeTheater,
  paSystems,
];
```

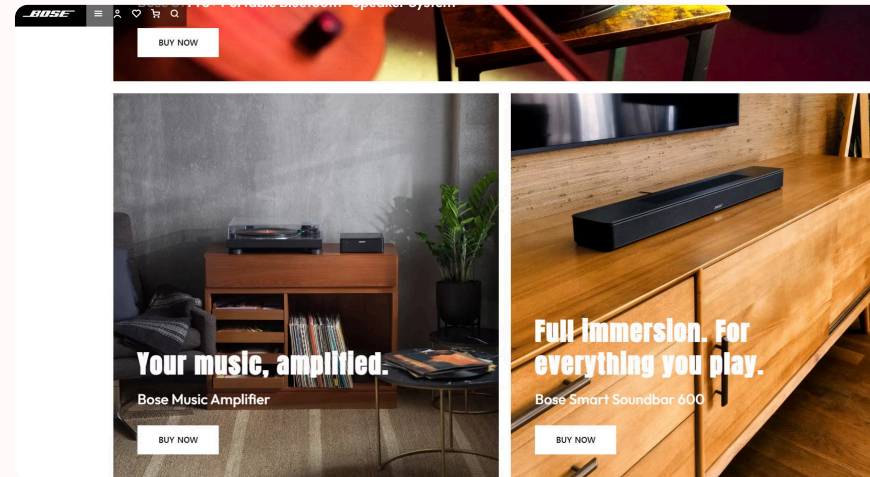
```
export const colorList={
  "black": "#000",
  "white_smoke": "#eee8e5",
  "sand_stone": "#928982",
  "lux_e_silver": "#adb0b1",
  "cypress_green": "#537262",
  "moonstone_blue": "#a2c2d9",
  "chilled_lilac": "#c5c6e1",
  "blue_dusk": "#486e8a",
  "stone_blue": "#7f98a5",
  "triple_black": "#283238",
  "arctic_white": "#f4f3ef",
  "bose_black": "#202020",
  "bgc1": "#181818",
  "bgc2": "#191919",
  "bgc3": "#0d0d0d",
  "bgc4": "#282828",
};
```



기타 랜덤함수 사용



상단 동영상 랜덤



서브 배너

기술 : 3.스크롤 이벤트

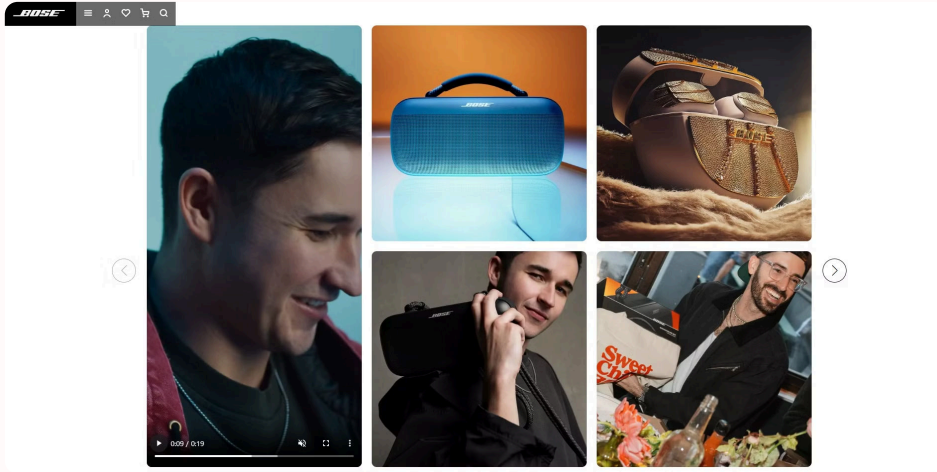


스크롤 이벤트 변수할당 및 구간설정

```
const mbBox = document.querySelector(".mb-box");
// 박스탑
const rectTop = mbBox.getBoundingClientRect().top;
const rectTopPer = Math.floor((rectTop / window.innerHeight) * 100);
// 제목 변수할당
const tit = mbBox.querySelector(".craft-tit1");
const ctit = mbBox.querySelector(".craft-tit2");
// 배너비디오 들 변수할당
const backImg = mbBox.querySelector(".back-img");
// 정보박스 들 변수할당
const infoBox = mbBox.querySelector(".cat-info-box");
// 이미지 수집
const img = mbBox.querySelectorAll(".box1-img");
// 글자 수집
const spanT = mbBox.querySelectorAll(".p-text");
```

```
// 스크롤 증감설정구역1 : 내용 나오기 설정
if (rectTopPer >= -200 && rectTopPer <= 100) {
  // console.log("달려~", rectTopPer);
  if (rectTopPer <= 50) {
    tit.classList.add("on1");
  } else {
    tit.classList.remove("on1");
  }
  if (rectTopPer <= 20) {
    ctit.classList.add("on1");
    infoBox.classList.add("on2");
  } else {
    ctit.classList.remove("on1");
    infoBox.classList.remove("on2");
  }
  if (rectTopPer <= -20) {
    spanT[0].classList.add("on1");
    img[0].classList.add("on1");
  } else {
    spanT[0].classList.remove("on1");
    img[0].classList.remove("on1");
  }
  if (rectTopPer <= -60) {
    spanT[0].classList.add("on2");
    spanT[1].classList.add("on1");
    img[0].classList.remove("on1");
    img[1].classList.add("on1");
  } else {
    spanT[0].classList.remove("on2");
    spanT[1].classList.remove("on1");
    img[1].classList.remove("on1");
  }
}
```

기술 : 4.그리드 출력



슬라이드 > 그리드

1. 그리드 배치 : 삼항 연산자
2. `map` (`li` 중 0번에 동영상태그, 나머지는 이미지태그)
3. 스크롤 이벤트 적용
 - 해당영역이 스크린의 일정부분 통과시 `autoplay`
 - 슬라이드 넘어가면 다음 동영상 `play`
 - 슬라이드 넘어가면 이전 동영상 `pause`

```
{snsData.map((item, index) => (  
  <SwiperSlide key={index}>  
    <ul className="sns-gallery">  
      {item.map((media, idx) => (  
        // ul>li 중 첫번째 li 만 그리드클래스 따로 적용하기  
        <li  
          key={idx}  
          className={idx === 0 ? "grid-item-2x2" : "grid-item-1x1"}  
        >  
          { /* 첫번째 li 에 비디오 출력 /  
            나머지는 이미지 출력 */ }  
          {idx === 0 ? (  
            <video  
              className="sns-video"  
              ref={(el) => (videoRefs.current[index] = el)}  
              src={process.env.PUBLIC_URL + media.src}  
              controls  
            ></video>  
          ) : (  
            <div className="sns-txt-box">  
              <p className="sns-txt">  
                <span>{media.txt}</span>  
              </p>  
              <img  
                className="sns-img"  
                src={process.env.PUBLIC_URL + media.src}  
                alt="gallery"  
              />  
            </div>  
          )}  
        </li>  
      )}    </ul>  
  </SwiperSlide>  
)
```


향후 계획

회원가입 / 로그인 모듈

장바구니 / 위시리스트

미디어쿼리

서브페이지 모듈 추가