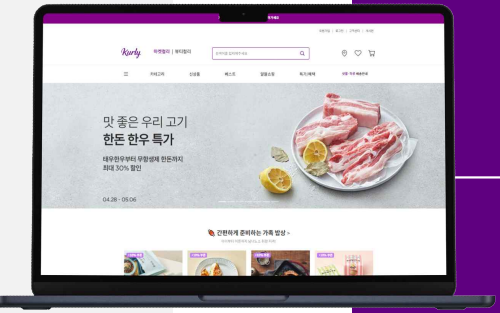


REACT PROJECT

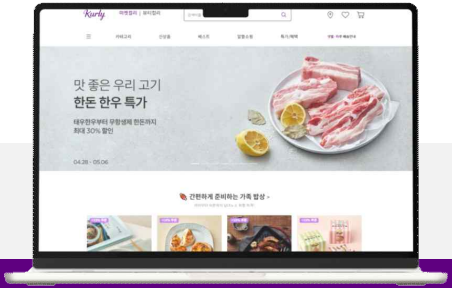
MARKET KURLY

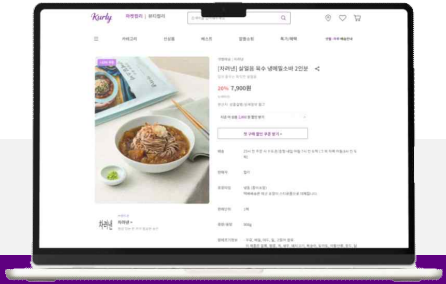
윤단비

2025.05

MARKET Kurly

마켓컬리 선정 이유





직관적이고 깔끔한 UI/UX 구성으로 사용자 흐름 파악에 용이

검색, 장바구니, 상세페이지 등 실무 기능을 전부 구현 가능

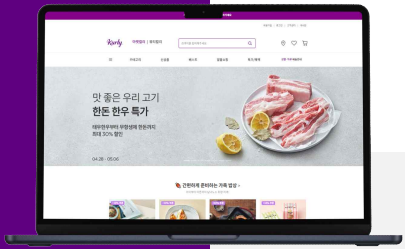
실무에서 자주 쓰이는 쇼핑몰 구조 익히기직관적인 UI/UX 구성 연습

MARKET Kurly

개발도구

JS

CSS



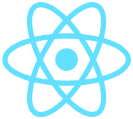
Redux

Bootstrap

React Router

MARKET Kurly

개발도구



사용자 인터페이스(UI)를 구축하기 위한
자바스크립트 라이브러리

JavaScript는 클라이언트 단에서
웹 페이지가 동작하는 것을 담당

웹 페이지의 스타일과 레이아웃을
디자인하는 데 사용

개발도구

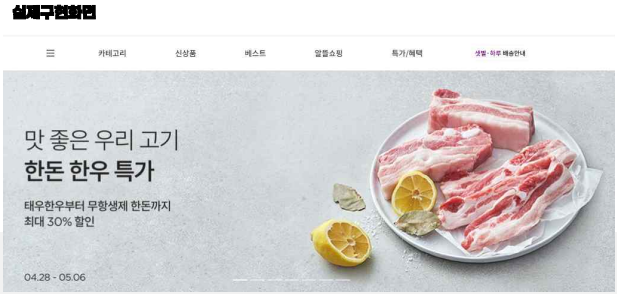
JavaScript용 상태관리 라이브러리

오픈소스 프론트엔드 프레임워크

React Router 라이브러리를 통해
React 애플리케이션에서 여러 페이지를 쉽게 관리하고
내비게이션을 구현가능

프로젝트 제작

menubar
Slider
Bootstap 사용



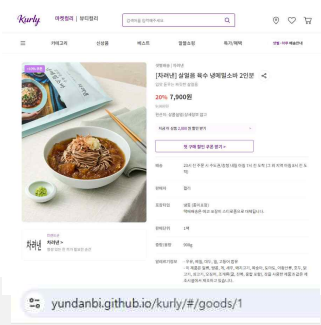
Bootstrap

Bootstrap 사용해 메뉴바,슬라이더를
React Bootstrap을 사용하여 빠르게 반응형 UI를 구축하였습니다.

프로젝트 제작

Router Parms 사용

실제구현화면



React Router

사용자가 상품을 클릭하면 React Router의 useParams()를 사용하여
상품의 ID를 URL에서 추출하고 해당 ID에 맞는 상품의 상세 정보를 렌더링합니다.
페이지마다 고유한 ID를 통해 각각의 상품 페이지로 이동합니다.
상품 상세 페이지에서는 주문하기와 상품 확인하기 버튼을 제공하며,
주문 시 장바구니로 이동됩니다.

MARKET Kurly

프로젝트 제작

검색구현

```
const [input, setInput] = useState("");
const filteredData = rawData.filter(
  (item) => item.name && item.name.toLowerCase().includes(input.toLowerCase())
);
```

```
<section className="search-result-section easy-meal-section">
  <div className="section-title"></div>
  <div className="product-list">
    {filteredData.length > 0 ? (
```

```
) : (
  <p style={{ padding: "20px" }}>검색 결과가 없습니다.</p>
)}
```

검색어에 따라 데이터 필터링하고
조건부 렌더링을 사용해 검색어가 있을 때만 검색 결과 보여줍니다

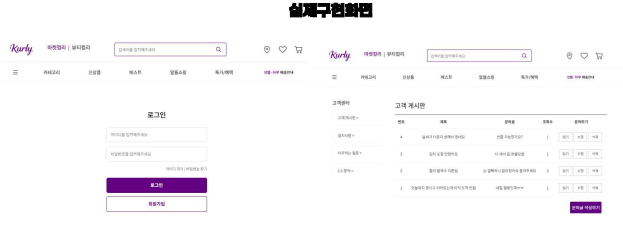
MARKET Kurly

프로젝트 제작

```jsx
function App() {
  return (
    <Routes>
      <Route path="/" element={<Home />} />
      <Route path="/home" element={<Home />} />
      <Route path="/goods/:id" element={<ProductDetail />} />
      <Route path="/cart" element={<Cart />} />
      <Route path="/login" element={<Login />} />
      <Route path="/board" element={<Board />} />
      <Route path="/login" element={<Login />} />
    </Routes>
  );
}

export default App;
```

실제구현화면



Router 사용 이유

React Router

React Router의 동적 이동 기능을 활용해,

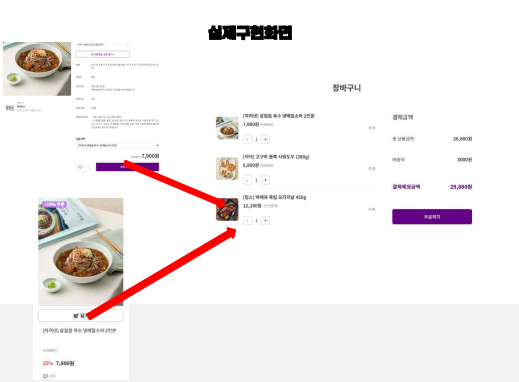
사용자의 액션에 따라 자연스럽게 화면 전환이 가능하도록 구성했습니다

MARKET Kurly

프로젝트 제작

```
const cartSlice = createSlice({
  name: "cart",
  initialState,
  reducers: {
    addToCart: (state, action) => {
      const exists = state.find((item) => item.id === action.payload.id);
      if (exists) {
        exists.quantity += 1;
      } else {
        state.push({ ...action.payload, quantity: 1 });
      }
    },
    removeFromCart: (state, action) => {
      return state.filter((item) => item.id !== action.payload);
    },
    changeQuantity: (state, action) => {
      const item = state.find((item) => item.id === action.payload.id);
      if (item) {
        item.quantity = action.payload.quantity;
      }
    },
  },
});

export const { addToCart, removeFromCart, changeQuantity } = cartSlice.actions;
export default cartSlice.reducer;
```



Redux 사용해 장바구니 구현

Redux

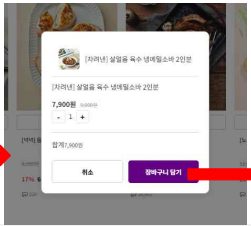
사용자가 상품을 선택하면 Redux를 통해 상태가 관리되며,
장바구니 페이지에서는 현재 담긴 상품들을 확인할 수 있습니다.
장바구니에 상품을 추가하고 삭제하는 기능을
Redux의 action과 reducer를 통해 처리합니다

MARKET Kurly

프로젝트 제작

Modal 사용해 장바구니 가기전 체크 구현

실제구현화면




```
import React, { useState } from "react";
import "./CartModal.css";

function CartModal({ product, onClose, onAdd }) {
  const [qty, setQty] = useState(1);

  const handleAdd = () => {
    onAdd(product, qty);
  };

  return (
    <div className="modal-overlay">
      <div className="modal-content">
        <div className="modal-header">
          <h2>
            <image
            src={process.env.PUBLIC_URL + `/${product.thumbnail}`}
            alt={product.name}
            width="50"
            height="50"
            />
            <p className="modal-title">{product.name}</p>
          </h2>

          <p className="product-subtitle">{product.name}</p>

          <div className="price-qty-row">
            <div>
              <strong className="price-discount">
                {product.product_choice.discount_price}
              </strong>
              <del class="price-original">{product.price.original}</del>
            </div>
            <div> <div class="qty-control">
              <button onClick={() => setQty((q) => Math.max(1, q - 1))}>-</button>
              <span>{qty}</span>
              <button onClick={() => setQty((q) => q + 1)}>+</button>
```

Main 홈화면에서 담기버튼을 클릭하면
Modal 사용해 장바구니에 바로 들어가기 전에
취소할 수 있는 창을 만들었습니다.

MARKET Kurly

프로젝트 제작

Login 페이지 구현

실제구현화면


```
const handleLogin = () => {
  alert("로그인 시도: " + id);
};

return (
  <>
    <Header />
    <div className="login-container">
      <h2 className="login-title">로그인</h2>
      <input
        type="text"
        placeholder="아이디를 입력해주세요"
        value={id}
        onChange={(e) => setId(e.target.value)}
        className="login-input"
      />
      <input
        type="password"
        placeholder="비밀번호를 입력해주세요"
        value={password}
        onChange={(e) => setPassword(e.target.value)}
        className="login-input"
      />
      <div className="login-links">
        <span>아이디 찾기</span> | <span>비밀번호 찾기</span>
      </div>
      <button className="login-btn" onClick={handleLogin}>
        로그인
      </button>
      <button className="signup-btn">회원가입</button>
    </div>
    <Footer />
```

로그인을 시도하면 id와함께
알람창이 나오는 간단한 로그인 페이지 구현하였습니다.

MARKET Kurly

프로젝트 제작

게시판 페이지 구현

```
{boardList
  .slice()
  .reverse()
  .map((board) => {
    <tr key={board.no}>
      <td>{board.no}</td>
      <td onClick={() => boardRead(board.no)}>
        {board.title}
      </td>
      <td onClick={() => boardRead(board.no)}>
        {board.description}
      </td>
      <td>{board.viewCount}</td>
      <td>
        <button
          className="btn-action"
          onClick={() => boardRead(board.no)}
        >
          읽기
        </button>
        <button
          className="btn-action"
          onClick={() => boardEdit(board.no)}
        >
          수정
        </button>
        <button
          className="btn-action"
          onClick={() => boardDelete(board.no)}
        >
          삭제
        </button>
      </td>
    </tr>
  })}
</tbody>
```

실제구현화면

고객 게시판

번호	제목	문의글	조회수	문의하기
4	날씨가 더운지 상해서 왔네요	반품 가능한가요?	1	읽기 수정 삭제
3	김치 포장 안댔어요	다 새서 옴 환불요망	1	읽기 수정 삭제
2	컬러 합국수 지존임	논 금백하니 없어장어요 돌려주세요	2	읽기 수정 삭제
1	오늘까지 온다고 되어있는데 아직 도착 안함	대입 캠핑인데ㅠㅠ	1	읽기 수정 삭제

문의글 작성하기

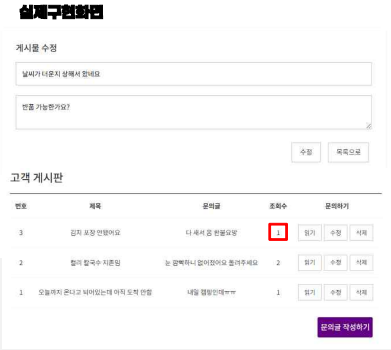
문의글 목록 출력
(게시판 리스트)

프로젝트 제작

게시판 페이지 구현

```
const boardDelete = (no) => {
  const updatedList = boardList.filter((b) => b.no !== no);
  setBoardList(updatedList);
  boardListView();
};

const boardEdit = (no) => {
  const boardToEdit = boardList.find((b) => b.no === no);
  setEditNo(boardToEdit.no);
  setEditTitle(boardToEdit.title);
  setEditDescription(boardToEdit.description);
  setListOk(false);
  setReadOk(false);
  setWriteOk(false);
  setEditOk(true);
};
```



글 수정 및 삭제
조회수 카운트 기능

React의 useState를 활용하여 글 작성, 조회, 수정, 삭제 기능을 모두 구현하고,
조회수 카운트 기능과 조건부 렌더링으로 페이지 전환 없이 동작하도록 구성했습니다.

MARKET Kurly

감사합니다

MARKET KURLY

운단비