



쇼핑몰 홈페이지

팀 네잎클로버

김상희 김준혁 이진우 고경복



목차



01

프로젝트 개요

02

프로젝트 팀 구성 및 역할

03

프로젝트 수행 절차 및 방법

04

프로젝트 수행 경과

05

자체 평가 의견



구성 및 역할



김상희

프로젝트 기획 및 구조 설계
전체 개발 방향 조정
코드 통합 및 전반적인 코드 구현



김준혁

자유 게시판 및 QnA
전체적인 페이지 간
회원 및 데이터 정보 전달



이진우

로그인
아이디 비밀번호 찾기
회원 정보 수정

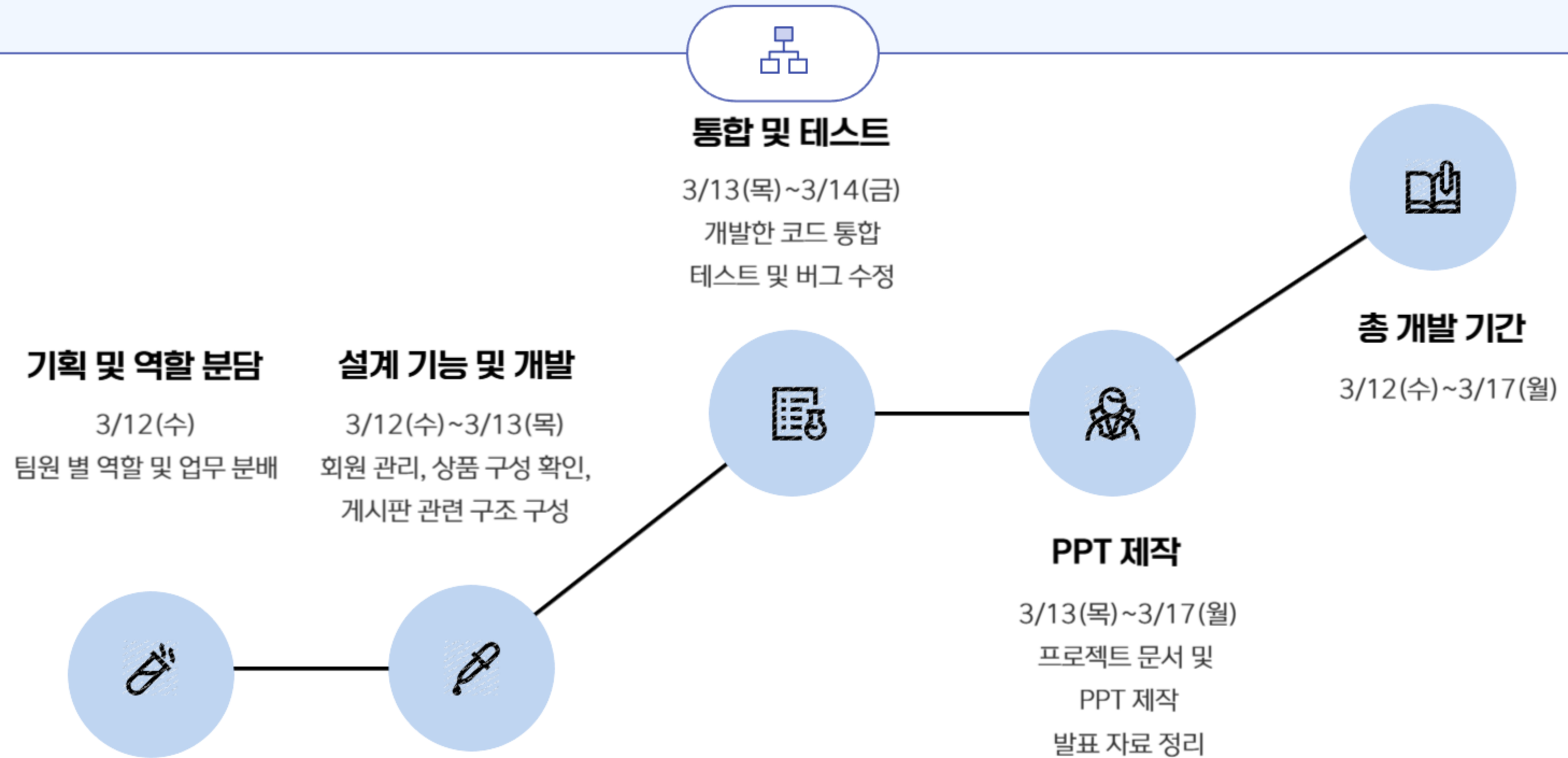


고경복

장바구니
상품 상세페이지
PPT 제작 및 발표



프로젝트 수행 절차



프로젝트 개요



주제 선정

사용자에게 직관적이고 편리한 쇼핑 환경을 제공할 수 있는 웹 페이지를 구현

내용

React, CSS를 활용하여
쇼핑몰 홈페이지를 구현

활용 장비

Visual studio Code, React, Node.js

구조

DOM Structure

활용 방안

직관적인 디자인으로 보다 쉽게 대중들이
이용 가능한 사이트이며, 향후 수요에 따라
쉽게 기능을 추가할 수도 있음



프로젝트 시각화

전체 구조

EXPLORER

PROJECT

node_modules

public

home

images

board.json

favicon.ico

index.html

logo192.png

logo512.png

manifest.json

products.json

qna.json

robots.txt

src

Mall

Board.js

Cart.js

CartContext.js

EditProfile.js

LogIn.js

Main.js

Product.js

ProductDetail.js

QnA.js

SearchId.js

SearchPassword.js

SignUp.js

App.css

App.js

JS App.js

src > JS App.js > ...

```
1 import { BrowserRouter, Routes, Route } from "react-router-dom"
2 import "./App.css"
3 import Main from "../Mall/Main"
4 import SignUp from "../Mall/SignUp"
5 import Cart from "../Mall/Cart"
6 import LogIn from "../Mall/LogIn"
7 import Board from "../Mall/Board"
8 import QnA from "../Mall/QnA"
9 import Product from "../Mall/Product"
10 import { CartProvider } from "../Mall/CartContext"
11 import ProductDetail from "../Mall/ProductDetail"
12 import SearchId from "../Mall/SearchId"
13 import SearchPassword from "../Mall/SearchPassword"
14 import EditProfile from "../Mall/EditProfile"
15
16 function App() {
17   return (
18     <CartProvider>
19       <BrowserRouter>
20         <Routes>
21           <Route path="/" element={<Main />} />
22           <Route path="SignUp" element={<SignUp />} />
23           <Route path="Cart" element={<Cart />} />
24           <Route path="Board" element={<Board />} />
25           <Route path="QnA" element={<QnA />} />
26           <Route path="Product" element={<Product />} />
27           <Route path="ProductDetail" element={<ProductDetail />} />
28           <Route path="LogIn" element={<LogIn />} />
29           <Route path="SearchId" element={<SearchId />} />
30           <Route path="SearchPassword" element={<SearchPassword />} />
31           <Route path="EditProfile" element={<EditProfile />} />
32         </Routes>
33       </BrowserRouter>
34     </CartProvider>
35   )
36 }
```



회원 관리



회원 가입

사용자의 정보를 입력하여 회원가입
회원가입이 완료된 후, 사용자의 정보는
localStorage(Server) 에 저장



로그인

로그인 시 사용자가 입력한 아이디와 비밀번호가
localStorage에 저장된 사용자 정보와 일치하면
로그인에 성공

아이디 비밀번호 찾기

아이디 : 사용자의 전화번호 입력 시 아이디 출력
비밀번호 : 사용자의 아이디와 전화번호 입력 시
SMS로 발송

회원 정보 수정

localStorage에 저장된 사용자 정보를 출력
변경된 정보로 localStorage 업데이트

회원관리



상품 관리



전체 상품

Json 파일에 저장된 전체 상품을 로드하여 출력

사용자는 각 상품을 클릭 가능

사용자가 상품을 클릭하면 상세 페이지로 이동



상품 상세 페이지

상품 목록에서 클릭한 상품의 product Id를 Json파일 내에서 비교 후 Id가 일치하는 상품 정보를 불러옴

선택한 상품 정보를 사용자 화면에 display

Productdetail.js 파일 내의 상품 옵션을 확인 및 선택 가능

장바구니 추가 및 상품 목록으로 돌아갈 수 있는 기능 포함



장바구니 관리



장바구니

비회원과 회원 모두 상품 추가 가능

상품 추가시 옵션 선택 필수

Add, Delete 함수를 이용하여 item Id와 product Id를 비교
서로 일치하는 상품과 옵션 데이터를 CartContext에 저장



장바구니 상태 관리

CartContext에서 전달 받은 데이터를 기반으로
각 상품 및 수량을 표시

장바구니 내에서 수량 변경 및 비우기 가능

상품 갯수를 추가할 경우 Add 함수 재실행
개별 상품 수량을 줄이거나, 전체 상품을 비우면
Delete 함수 재실행

장바구니



게시판 관리



QnA 및 자유 게시판

다른 사용자나 관리자와 소통할 수 있는 게시판 기능을 구현

로그아웃 상태에서 새 글 작성을 누르면 로그인 페이지로 이동 → 로그인 후 원래페이지로 복귀
useLocation을 사용하여 이전 화면 위치를 State_from으로 기억 navigation으로 돌아감

게시물 목록 출력 및 개별 열람, 조회수 기능을 제공

게시물 등록 후에는 localStorage에 저장되어 새로고침 시에도 유지



시연 영상

Main

회원 가입

장바구니

로그인

자유 게시판

Q&A

전체 상품

장바구니

장바구니가 비어 있습니다.



자체 평가



프로젝트 수행 후 평가 및 개선 방향

목표했던 기능은 모두 구현하는 데 성공했지만,
일부 추가 보완해야 할 점들이 남아 있어 9점으로 평가함.

만족한 점

프로젝트 초반에 빠른 작업 속도로 기본적인 틀을 완성
회원 관리, 로그인, 장바구니 등의 핵심 기능을 정상적으로 구현
React Router 및 Context API 활용을 통해 효과적인 상태 관리 및
페이지 이동을 구현

아쉬운 점

빠르게 개발을 진행하다 보니 일부 기능의 디테일이 부족하여 추가적인 수정이 필요
대규모 쇼핑몰 수준의 디자인과 기능을 갖추기에는 다소 부족
반응형 웹 디자인이 미흡하여 화면 크기에 따라 유동적으로 조정될 필요가 있음

추후 개선 및 보완할 점

현재 결과물은 기본적인 쇼핑몰 기능을 포함하고 있지만,
대형 쇼핑몰 수준으로 발전하기 위해서는 추가적인 개선이 필요

▶ UI 개선

반응형 웹 디자인을 적용하여 유동적인 Display 를 제공할 UI 최적화 작업

▶ 추가 기능 보완

상품 리뷰 및 별점 기능 추가
주문 및 결제 기능 구현
관리자 페이지 개발 (상품 등록 및 관리)

▶ 데이터 관리 및 보안 강화

현재 localStorage를 활용하고 있지만, DB와 연동하여 보안성을 강화



자체 평가



프로젝트를 수행하면서 느낀 점 및 경험한 성과

▶ React 및 상태 관리에 대한 이해 심화

Context API, useState, useEffect 등을 활용하여 전역 상태를 관리하는 경험을 쌓음
localStorage를 활용하여 로그인 및 회원 관리 기능을 구현

▶ 실전 경험을 통한 성장

단순히 기능 구현에 그치지 않고, 실제 서비스에서 필요한 요소들(UX, 보안, 데이터베이스 등)을 고려하는 시야가 넓어짐
프로젝트를 완료하면서 더 높은 완성도를 위해 어떤 점들을 추가적으로 고려해야 하는지 명확히 알게 됨

▶ 향후 목표

UI/UX 디자인 업데이트 및 반응형 웹 적용
상품 리뷰, 추천 시스템, 관리자 페이지 추가 등의 기능 확장
데이터베이스 및 보안 강화를 위한 서버 연동 계획

이 프로젝트를 기반으로 앞으로 더 완성도 높은 웹 애플리케이션을 개발할 수 있도록 발전시켜 나갈 예정입니다.

