

# 통합 테스트 작성하기

## - NavigationBar

실무에 바로 적용하는 프론트엔드 테스트

이재성

# 사용 브랜치



shopping-mall-integration-test

```
$ git checkout shopping-mall-integration-test
```

# NavigationBar의 비즈니스 로직

✓ “Wish Mart” 텍스트 로고를 클릭할 경우 메인 페이지로 이동

로그인이 안된 경우

✓ 로그인 버튼이 노출되며, 클릭시 로그인 페이지로 이동

로그인이 된 경우

✓ 장바구니 카트 버튼, 장바구니에 담긴 상품수가 나타남

✓ 회원 이름으로 된 로그아웃 버튼이 노출 됨

✓ 장바구니 카트 버튼을 누르면 장바구니 페이지로 이동

✓ 로그아웃 버튼을 클릭할 경우 “로그아웃 하시겠습니까?” 문구가 담긴 모달 노출

✓ 모달의 확인 버튼 클릭시 로그아웃, 모달 닫힘. 취소 클릭시 모달 닫힘

# 정리

## NavigationBar

- 테스트를 위한 사전 준비
  - 로그인, 로그아웃 여부를 판단하기 위한 user 스토어 모킹
  - 장바구니 상품 수량을 가져오기 위한 cart 스토어 모킹
  - 로그인한 사용자 정보를 가져오기 위한 /user API를 msw로 모킹
    - 비로그인 상태로 모킹
- 테스트 작성하기
  - /user API에 대해 상황에 따라 다른 모킹을 하기 위해 server.use() API 사용
  - Teardown(afterEach)에서 server.resetHandlers() 호출로 동적인 API 모킹 초기화
  - 일관된 모킹 데이터로 안정성 있는 테스트 작성