

E2E 테스트의 한계

실무에 바로 적용하는 프론트엔드 테스트

E2E 테스트를 통해

- ✓ 단위·통합 테스트에서 검증하지 못했던 API 호출에 대해 검증
 - ✓ 모든 모듈이 조합 되었을 때 기능 연동에 대해 검증
 - ✓ 페이지 이동과 같은 후처리 검증
- ✓ 직접 API를 사용하기 어렵다면, API 스텀빙을 통해 워크플로우 검증

E2E 테스트의 한계

1. 시간이 오래 걸린다

- ✓ 렌더링, API 응답 등 각 작업 워크플로우를 직접 시뮬레이션

E2E 테스트의 한계

1. 시간이 오래 걸린다

- ✓ 렌더링, API 응답 등 각 작업 워크플로우를 직접 시뮬레이션
→ 테스트 실행시간이 길다는 것은 테스트 피드백을 받아
수정하는 시간도 증가한다는 것을 의미

E2E 테스트의 한계

1. 시간이 오래 걸린다

- ✓ 렌더링, API 응답 등 각 작업 워크플로우를 직접 시뮬레이션
 - 테스트 실행시간이 길다는 것은 테스트 피드백을 받아 수정하는 시간도 증가한다는 것을 의미
 - 개발 생산성이 떨어질 수 있음

E2E 테스트의 한계

1. 시간이 오래 걸린다

- ✓ 렌더링, API 응답 등 각 작업 워크플로우를 직접 시뮬레이션
 - 테스트 실행시간이 길다는 것은 테스트 피드백을 받아 수정하는 시간도 증가한다는 것을 의미
 - 개발 생산성이 떨어질 수 있음

병렬 실행을 위해서는, Cypress 유료 플랜을 사용하거나
Sorry Cypress를 활용해 개인 서버를 구축해야 함

E2E 테스트의 한계

2. 외부 환경 요소의 영향이 크다

✓ 백엔드의 직접적인 영향을 받아 테스트가 실패한다

🤔 간헐적으로 실패하는 네트워크 문제였다면?

🤔 테스트 데이터베이스의 데이터를 수정하거나 초기화한다면?

🤔 번거로운 테스트 환경을 위해 매번 백엔드의 협조를 바라야한다면..?

E2E 테스트의 한계

3. 디버깅이 오래 걸린다

- ✓ 전체 앱을 구동하여 검증하다보니, 테스트에 영향을 미칠 수 있는 범위가 다른 테스트에 비해 훨씬 많다
- ✓ 많은 것을 검증할 수 있다는 장점에 역설적으로 많은 부분에 걸쳐 실패할 수 있어 문제를 찾기 힘들 수 있다

E2E 테스트의 한계

4. 그 외 기술적 한계..

- Chrome
- Chrome Beta
- Chrome Canary
- Chromium
- Edge
- Edge Beta
- Edge Canary
- Edge Dev

- Electron
- Firefox
- Firefox Developer Edition
- Firefox Nightly
- WebKit (Experimental)

Chromium

Google Chrome & Microsoft Edge

Firefox

WebKit

Cypress 지원 환경

Playwright 지원 환경

✓ 아직은 기술적 한계로 많은 모바일 환경과 브라우저를 지원하지 않음

정리

E2E 테스트

- 단위·통합 테스트에서 검증하지 못했던 API 호출이나 모든 모듈이 조합 되었을때의 기능을 검증할 수 있음

E2E 테스트의 한계

- 단위·통합 테스트에 비해 느린 속도
 - 실제 웹앱을 구동해 테스트를 진행하기 때문에 렌더링이나 API 응답에 많은 시간 소요
 - 테스트 실행 시간 증가 → 개발 생산성 저하
- 외부 환경 요소로 인해 테스트가 깨질 수 있음
 - 온전한 테스트 실행을 위한 관리 비용이 많이 들어감
- 디버깅 시간이 오래 걸림
 - 전체 앱을 구동해 검증 → 테스트에 영향을 미칠 수 있는 모듈의 범위가 매우 큼
 - 테스트가 실패 했을 때 원인을 찾아 수정하는데 많은 시간이 소요될 수 있음