

Introduction

브라우저 성능 최적화와 UX 개선을 통해 복잡한 화면의 로딩 속도를 개선하며, 사용자에게 직관적이고 빠른 경험을 제공합니다. 처음 도전한 Flutter 앱을 7개월 만에 성공적으로 출시하며 비즈니스 우선순위에 맞춘 기술 적용 역량을 증명했습니다. 더 나아가 JavaScript 환경에서 컴포넌트 설계와 상태 관리를 통해 대규모 UI와 동적인 사용자 경험을 확장성 있게 구현하고자 합니다. 기술적 완성도와 사용자 경험의 조화를 목표로 가치 있는 결과물을 만듭니다.

Career

설로인

22.04.04 - 23.07.31 (1년 4개월)

한우 1차 가공업자와 2차 가공업자 정육점, 음식점 등을 이어주는 B2B 플랫폼 “본대로”

프론트엔드 엔지니어

Flutter, Dart

■ 핵심 성과

처음부터 B2B 플랫폼 런칭까지, Zero to One

- Flutter 학습을 업무와 병행하며 **Android, iOS, Web** 버전을 단 7개월 만에 성공적으로 출시
- Flutter 아키텍처와 네이티브 코드까지 깊숙히 학습해보며 플랫폼 간 클릭, 드래그 및 핀치 이벤트의 차이를 조율
- MVVM 패턴, Service Locator 패턴을 활용한 의존성 주입, 레이어드 아키텍처 등으로 **유지 보수성과 확장성을 고려한 설계**
- 제한된 리소스 환경에서 **사용자 행동 기반(BDD) 단위 테스트 코드 작성**으로 앱 안정성을 높이고 자체 품질 보증 및 디버깅 시간 단축
- 초기 세팅부터 소셜 로그인, 회원가입, 홈 화면, 상품 조회, 필터 및 정렬, 장바구니, 주문 생성 및 조회, 내 정보 기능 개발 등

앱 심사 시간을 줄이고 홈 화면에 즉시 반영하는 Server Driven UI

- JSON으로 UI를 정의하고 **Flutter에서 렌더링 로직을 구현**해 앱 업데이트 없이 사용자 데이터 수집 및 AB 테스트 가능하도록 구현
- 초기 렌더링 최적화를 위해 무거운 로직을 build 메소드에서 분리하고, 위젯을 작은 단위의 불변 객체로 선언하여 불필요한 렌더링 방지
- 이미지 리사이징과 Custom Painter를 적극 활용해 무거운 **asset 사용**을 지양하고 홈 화면 진입 속도 개선
- RepaintBoundary를 활용해 재렌더링 최적화 및 **Flutter inspector**로 병목 현상 분석 및 제거
- 사용자의 행동패턴**에 근거하여 하루 한 번 데이터를 갱신하는 캐싱 전략을 결정하고 도입

API 요청 감소 및 UI 성능 향상을 위한 반응형 상품 필터링 시스템 구현 (RxDart)

- Stream과 BehaviorSubject**를 활용하여 필터 변경 시 전체 API 재요청을 방지하고 필요한 데이터만 서버에서 가져오도록 최적화
- 비동기 데이터 흐름을 기반으로 필터링, 로딩, 오류 상태를 분리하여 유지보수성을 향상시키고 UI와 데이터 상태를 효과적으로 관리
- 이전 필터와 새로운 필터를 비교해 동일한 조건일 경우 **상태 변경**을 방지, 불필요한 데이터 로딩을 줄이고 UI 성능 향상

Annotation Processor와 build_runner를 활용해 DateTime, Decimal 등의 직렬화/역직렬화 자동화

- JsonConverter로 JSON 변환 로직을 공통화하여 코드 응집도 향상, UI 안정성 및 데이터 무결성 보장

■ 그 외

- 22년 12월 첫 출시 이후, 3주차 총 매출 **4,500만 원** 달성
- 50대 고객층 맞춤 UX/UI 개선으로 가독성 강화, **UI 관련 CS 요청 건수 주 10건 → 0건 감소** ([참고링크](#))
- Build Flavor 기반 엔지니어링 모드 설계로 **앱 버전 정보, User Agent 관리, 버그 재현 시스템** 구축
- 멀티 레포지토리 환경에서 공통 Custom Exception을 정의 및 통합하여 **예외 처리 일관성 확보**
- 국제화 (i18n) 지원을 위한 다국어 시스템 구축

Project 01

Gravidot

[Source Code](#) [Website](#) [Blog](#)

24.12.02 - ing

멀티터치로 아이디어를 연결하고 실시간 협업이 가능한 브레인스토밍 툴

1명

TypeScript, Next15, Zustand, Supabase, Turbopack, Jest, Playwright, Github Actions

고해상도 대응 Shape 클래스 설계

- HTML5 Canvas API 기반으로 도형의 위치, 크기, 색상, 회전 등 속성을 유연하게 조작
- DPR(Device Pixel Ratio) 적용으로 확대/축소 및 다양한 디바이스 환경에서도 선명한 렌더링 품질 유지

멀티터치 기반 실시간 도형 편집 인터랙션

- 직접 멀티터치 입력을 처리하는 커스텀 훅을 구현하여 여러 손가락으로 도형의 크기, 색상, 회전, 모양을 직관적으로 조정
- TouchEvent를 활용해 손가락 간 거리와 각도를 계산하고 도형 속성을 실시간 업데이트하며 최적화된 사용자 인터랙션 제공

FSD(Feature-Sliced Design) 기반의 확장 가능한 아키텍처

- entities, features, shared, views 레이어로 도형 데이터, 이벤트 핸들링, 상태 관리를 분리하여 유지보수성과 확장성 강화
- 각 레이어의 구성 요소는 동일 또는 상위 레이어에 의존하지 않아, 사이드 이펙트 없이 독립적인 수정 및 유지보수가 가능

Project 02

Commit Guardians

[Source Code](#) [Website](#) [Blog](#)

24.10.30 - 24.11.22 (기획 1주 + 개발 3주)

깃헙 레포지토리의 커밋 퀄리티를 계산하고, 배지를 마크업 문서에 올릴 수 있는 프로젝트

프론트엔드 3명

JavaScript, React, Zustand, idb-keyval, Firebase Storage, Vite, Vitest, Github API

빈번한 API 호출을 병렬 요청과 배치 처리 방식으로 최적화

- GitHub API를 활용해 레포지토리 커밋 데이터를 페이지 단위로 병렬 요청하며 Promise 기반 비동기 처리로 성능 최적화
- 특정 브라우저의 동시 요청 한도를 고려해서, 배치 처리와 요청 간 딜레이를 추가하여 효율적으로 데이터를 가져오는 로직 구현
- Zustand의 persist 미들웨어를 통해 IndexedDB와 상태 관리를 통합하고 GB단위의 대량 데이터를 안정적으로 저장

많은 컴포넌트들을 화면에 렌더링 하기 위한 커스텀 Virtual Scroll 도입 (참고코드)

- 1,000개 이상의 컴포넌트 렌더링으로 인한 성능 저하를 해결하기 위해 Virtual Scroll을 직접 구현하여 INP 지수를 안정적으로 유지
- window 객체와 boundingClientRect() 를 활용해 뷰포트 내 컴포넌트 높이 등을 감지

Project 03

Scribubble

[Source Code](#) [Youtube](#) [Blog](#)

22.12.04 - 22.03.10

3D 웹 환경에서 함께 낙서할 수 있는 메타버스 그림판

프론트엔드 3명 | 백엔드 1명

JavaScript, Preact, Three.js, Socket.io

3D 드로잉 라인 생성 및 3D 객체 삽입

- LineGeometry, LineMaterial로 다양한 라인 스타일 지원, 부드러운 선 구현
- createLineGeometry, createLine, addPosition, removeLastLine 함수로 라인 추가/업데이트/삭제
- OrbitControls, TransformControls로 3D 객체 삽입 & 자유로운 조작
- Raycaster로 마우스 위치 감지, 3D 객체 충돌 체크 및 선택 기능 추가

Socket.io 기반 협업과 3D 공간에서 실시간 동기화

- Socket.io 이벤트 구조 (draw start, drawing, draw stop)로 양방향 동기화
- initSocketListener로 객체 생성, 이동, 회전, 크기 조정까지 실시간 반영
- 독립적인 드로우 상태 관리를 통해 동시성 문제를 해결하여 다수 사용자 협업 시 충돌 최소화

Activities

글또 (글쓰는 개발자 모임) 9기, 10기 - 기술 블로그 운영 및 글 작성

23.12.01 - ing

- [WEB & APP, 어떻게 사용자를 설득하는가?](#)
- [\[Flutter\] 코드가 어떻게 화면이 되나요\(1\) — AOT vs JIT compiler](#)
- [\[React\] Fiber Architecture - 수 천 개의 UI 요소들을 합리적인 성능으로 업데이트하기](#)

Flutter Seoul Organizer - 플러터 서울 커뮤니티에서 여러 행사를 주최

24.01.01 - 24.12.31

- [플러터 컨퍼런스 행사, Future<Flutter>](#)
- [튜링의 사과 월간 발표 모임](#)
- [Flutter Web 텍스트 렌더링 문제를 주제로 발표](#)

모듈과 타이포그래피 워크샵 p5.js 보조강사

24.06.24 - 24.06.26

서울시립대 워크샵에서 코딩을 어려워하는 시각디자인 전공 학생들에게 canvas 기반 코드 구현 방식 제공

OSSCA 오픈소스 컨트리뷰션

23.07.01 - 23.10.25

OpenGL ES 3.0 책을 한국어로 번역하면서 오픈소스에 기여

Education

한양사이버대학교

24.03.01 - ing

컴퓨터공학 컴퓨터시스템의 이해, 컴퓨터그래픽스와 VR, 시스템분석 및 설계, 자료구조론, 대학수학

바닐라코딩

24.08.01 - 24.12.21

프론트엔드 부트캠프 JavaScript, Asynchronous, Closure, Event Loop, React, Virtual DOM, Redux

연세대학교 미래캠퍼스

16.03.01 - 21.08.01

시각디자인 전공 / 디지털 아트 부전공 Design Thinking, Business Driven, Infographic, UXUI, Motion Graphic