



Frontend Developer

정다운 Jeong Daun

☎ 010-9386-6984

✉ daunje0@gmail.com

🏠 <https://devlog-dusky.vercel.app/>

🌐 <https://daunje0.tistory.com/>

## About Me

안녕하세요. Vue.js, JavaScript 기반의 프로젝트를 진행해 온 3년차 프론트엔드 개발자 정다운 입니다.

SI 프로젝트 현장 경험을 통해 협업 능력과 이슈 처리 능력을 키웠습니다.

기존 SI 프로젝트에서 접하기 어려웠던 React, Next.js, TypeScript 등의 기술 스택을 학습하고 적용하기 위해 개인 프로젝트를 진행하고 있습니다. 또한, Medium, ByteByteGo, daily.dev 등 기술 블로그 번역을 통해 꾸준히 새로운 개발 트렌드를 학습하고 있습니다.

자영업 경험을 통해 얻은 고객 중심 사고방식과 거시적인 문제 해결 능력을 바탕으로 사용자 경험 개선에 기여하고자 합니다.

## Work Experience

2022.07 ~ 2024.09(2년3개월)

주식회사 코드몬스터

2022.12 ~ 2023.12

Vue 기반 프론트엔드 개발자

하나 캐피탈 리빌딩 프로젝트 (1년 1개월)

### 1. 주요성과

- 레이지 로딩 적용으로 초기 로딩 시간 최소화, 사용자 경험 향상
  - 메인 페이지 화면의 초기 렌더링 속도 91% 개선 ( 7 초 -> 0.6 초)
- 하나캐피탈 코어 서비스인 차량 렌트/리스/핫딜 서비스 화면의 Vue3 마이그레이션 및 신규 기능 개발 담당 ( 프론트 기여도 80 % )

### 2. 주요업무

- JSP에서 Vue3로 대고객 서비스 화면 마이그레이션 및 신규 요건 개발
- 다이렉트 리스/렌트 업무의 차량 상품 관련 공통 프로세스 구현 (차량 제조사 선택, 차종 및 색상, 세부 옵션 선택, 견적 산출)
- REST API 연동 및 상태 관리 구현
- 별도 퍼블리싱 팀과 협업하여 UI 구현

### 3. 사용 기술 및 툴 : Vue3, JavaScript, Pinia, Git/Github

### 4. 얻은 인사이트

- SPA 의 렌더링 방식에 대해 다시 한번 고민할 수 있었습니다.

레이어 구성은 모든 화면이 연결되어 있기 때문에 초기 모든 리소스를 불러오는 데 많은 시간이 들 수 밖에 없으며, **dynamic import**를 이용한 레이저 로딩으로 코드 스플리팅 효과는 필수라 판단 되었습니다. 이 결정으로 가장 중요한 이슈였던 초기 렌더링 속도를 91% 향상시킬 수 있었습니다.

- 리소스의 최적화의 필요성을 느꼈습니다.

당시 프로젝트 구조는 차량 이미지를 외부 리소스를 받아오는 형식이었습니다.

국내외 모든 차종의 이미지 파일도 webp가 아닌 png 그대로 받아 화면에 렌더링을 해야 했고, 이로 인해 많은 리소스가 낭비되는 상황이었습니다.

프론트엔드에서 할 수 있는 최적화 방안을 모색했지만, 자체 이미지를 사용하지 않는

구조적인 제약으로 인해 적용하지 못했습니다. 차기 프로젝트에서는 **이미지 포맷 변경**,

필요하다면 **CDN적용**을 통해 리소스 최적화를 적극적으로 추진하고 싶습니다.

- 협업 툴 (JIRA, Confluence) 활용을 통한 프로젝트 관리 효율성 향상 필요성 인식.

당시 PM분께서 일정 및 담당 업무 진행 상황을 엑셀로 하시면서 항상 고생하시는 모습을 봤습니다. 협업에 있어 프로젝트 / 일정 / 이슈 관리를 위한 툴의 사용은 생산성 향상을 위해 필수라고 느꼈습니다.

#### 5. 해당 링크 (개발자 화면에서 모바일로 설정해야 볼 수 있습니다)

차량 리스 : <https://m.hanacapital.co.kr/untc/atfi/diat/AtfiDiat01P>

차량 렌트 : <https://m.hanacapital.co.kr/untc/atfi/dirt/AtfiDirt01P>

핫딜 : <https://m.hanacapital.co.kr/atfi/deal>

2024.02 ~ 2024.05

Web Frontend Developer

### 지에스 리테일 수퍼 통합 프로젝트(4개월)

#### 1. 주요성과 : 공통 컴포넌트 지원 작업 병행으로 **Atomic 컴포넌트 제작 참여**

#### 2. 주요업무 : 행사, 발주 기여도 25 / 25 %

- GS THE FRESH 전사 수주 관련 어플리케이션 화면 개발
- MiPlatform에서 Vue3로 마이그레이션
- 행사, 발주 파트의 AG-Grid를 활용한 데이터 바인딩 및 기능 구현
- Atomic 디자인 방법론 기반의 컴포넌트 시스템 활용
- 별도 퍼블리싱 팀과 협업하여 UI 구현

#### 2. 사용 기술 및 툴 : Vue3, JavaScript, Pinia, AG-Grid, JIRA, Confluence, Bitbucket, StoryBook

#### 3. 얻은 인사이트

- 구조화된 협업 개발 워크플로우의 중요성을 배웠습니다.
  - 체계화된 협업 워크플로우로 이슈 및 버그에 관한 빠른 대응을 할 수 있었습니다.
  - 담당자를 가시적으로 알 수 있어 개발자들 상호 간의 커뮤니케이션, 이슈 관련 화면 첨부로 테스터들과의 커뮤니케이션 비용도 절감 되었습니다.
- Atomic 디자인의 장단점을 알 수 있었으며, 팀 전체의 충분한 이해, 필요한 부분만 선 적용 하며 점진적으로 구체화가 필요하고, 유연한 컴포넌트를 만들고 storybook 같은 적절한 도구가 필요하다는 점을 알았습니다.

2024.06 ~ 2024.09

Web FullStack Developer

## Contents Bridge CRM 신규 개발 (4개월)

1. 주요성과 : 공통 컴포넌트 및 공통 함수 모듈화로 팀 생산성 향상 기여
  - Ag-grid 메서드 **공통 함수로 모듈화** (그리드 필터링, 엑셀다운 로드 기능 등)
  - 버튼, 팝업, 모달창 등 **공통 컴포넌트 구현**
  - 출퇴근 개발 담당자가 사용할 MySql 프로시저 및 이벤트 스케줄러 작성
2. 주요업무
  - 사내 전자 결재 프로세스 관련 화면 개발
  - 해당 도메인 FE부터 BE, DB 설계 풀스택 개발 수행
  - 공통 컴포넌트 및 함수 모듈화 작업 수행
3. 사용 기술 및 툴 : Vue3, TypeScript, Tailwind CSS, AG-Grid, Java, Spring Boot, MyBatis, MySQL, Git/GitLab
4. 얻은 인사이트
  - 업무 프로세스 이해의 중요성을 깨달았습니다.
  - 온전치 않은 디자인 시안과 요구사항 분석이 누락된 예상치 못한 상황에서 개발을 시작하게 되었습니다. 기안문, 협조문, 신청서의 결재 흐름 차이, '결재 요청 후 문서 수정 방법'과 같은 특이 사항에 대한 설명이 부족하여, 결재 프로세스를 구현하는 데 어려움을 겪었습니다. 이러한 어려움을 극복하기 위해 **타사의 CRM을 분석**하여 결재 흐름을 파악하고, 누락된 요구사항을 적극적으로 찾아 보완하여 개발을 완료할 수 있었습니다. 이러한 경험을 통해 업무 프로세스에 대한 깊이 있는 이해가 성공적인 개발에 필수적임을 깨달았습니다. 향후 고객사와 처음부터 업무를 수행할 기회가 주어진다면, 개발 전 요구사항 정의 단계에서 사용자 인터뷰, 데이터 분석 등을 통해 요구사항을 명확히 하고, 디자인 시안 검토 시 개발 가능성 및 기술적 제약 사항을 적극적으로 협의하여 업무 프로세스 개선에 기여하고자 합니다

## Portfolio

2024.02 ~ 2024.03

Web Frontend Developer

## 개인 소개 페이지 제작

1. 주요성과 :
  - 하나 캐피탈 프로젝트에서 얻은 경험을 바탕으로 이미지 포맷을 png에서 webp로 전환하여 이미지 리소스 크기를 71.45% (1.66MB -> 474KB) 압축했습니다. 초기 로딩 속도 개선, 사용자 경험 향상, 트래픽 비용 절감 효과를 기대했습니다.
  - 핵심 로직 및 컴포넌트에 대한 단위 테스트를 통해 코드 커버리지 90.76%를 달성했습니다.
  - 다양한 해상도 및 화면 크기에서 테스트를 진행하여 반응형 디자인의 일관성을 확보했습니다. (테스크탑 6종, 테블릿 2종, 모바일 3종)
  - 테일윈드 4버전 도입하여 빌드 파일 크기 감소, 빠른 빌드 속도, 트리 웨이킹 등 성능 최적화를 통해 개발 효율성을 높였습니다.

2. 사용 기술 : NextJs, TypeScript, TailwindCSS, Framer Motion, Vitest

### 3. 얻은 인사이트

- 학습 내용 활용에 중점을 두어, TDD를 제대로 적용시키지 못하였습니다.  
앞으로 개발 초기 단계부터 테스트 코드를 먼저 작성하는 TDD방식을 적용하여 코드 품질을 향상시키고 개발 효율성을 높이하고자 합니다. Red-Green-Refactor 주기를 철저히 준수하여 테스트 가능한 코드를 작성하고 지속적인 리팩토링을 통해 코드 유지보수성을 확보할 계획입니다.
- 단위 테스트만으로는 사용자 시나리오 기반의 통합적인 테스트가 어렵다는 것을 인지하고, E2E 테스트의 필요성을 느꼈습니다. 특히, 차후 실제 api를 이용하게 될 경우, 여러 컴포넌트 간의 상호작용을 검증하고 예상치 못한 예외 상황을 발견하기 위해 E2E 테스트 도입도 함께 해야한다는 것을 느꼈습니다.
- 현재 프로젝트에서는 시간 제약으로 인해 크로스 브라우저 테스트를 진행하지 못했지만, 관련 학습을 통해 해당 기술을 습득하고 적용할 계획입니다
- light house 의 LCP 에서 1.8S / CLS 는 0.375로 에 대한 개선이 필요하다는 점을 확인했습니다. 렌더링 방식 개선 등 다양한 성능 개선 기법을 적용하여 LCP를 1.5초 이하, CLS를 0.1 이하로 개선하는 것을 차후 목표로 하고 있습니다.

## Education

2021.09 ~ 2022.03

한국 소프트웨어 인재 개발원 JAVA 과정 수료

2011.04

CCNA 취득

2009.02

정보처리 산업기사 취득

2000.04 ~ 2004.03

리츠메이칸 아시아 태평양 대학 | 학사

## Skills

### Frontend

HTML, CSS, Tailwind, Javascript, TypeScript, Framer-motion  
React.js, Next.js, Zustand, Vue.js, Pinia, Vite,  
Vitest, React Testing Library, AG-Grid

### Backend

Express.js, java, SpringBoot

### Others

Notion, Git/Github, Jira, Confluence, Bitbucket

### Design

Photoshop, Figma, StoryBook

### Data-Base

MySQL, Postgresql

### Dev-Ops

Docker, Docker Compose