| Front-end 개발자  장 동 현 | 생년월일  전화번호  이메일 주소  주소 | 1996년 7월 (만 26세)  010-5011-8007  [wshmin1234@gmail.com](mailto:wshmin1234@gmail.com)  경기도 부천시 원미구 중4동 |
| --- | --- | --- |



| 링크 | Github : <https://github.com/ww8007>  포트폴리오 : [장동현 포트폴리오](https://docs.google.com/presentation/d/1teJ0Ty4DJ4zQCHhkZMAfuaJ3AyvcZDcBxYkejVIGlXo/edit?usp=sharing)  기술 블로그 : <https://velog.io/@ww8007> | |
| --- | --- | --- |

| 소개 | * 열린 자세로 사용자와 적극적으로 소통하여 고객이 불편한 점을 빠르게 고쳐나감으로써 인정받는 개발자가 되고 싶습니다. * 현재는 함수형 프로그래밍과 견고한 테스트가 뒷받침 되는 개발에 관심을 가지고 있습니다. | |
| --- | --- | --- |

| Skill Set |  | JavaScript, TypeScript, React, React Native  Node.JS, Jest, Nest, Next.js Docker, Figma, Dart, AWS, C, C++, Python | | |
| --- | --- | --- | --- | --- |

| 교육 및 대외활동 | * 활동명 : Silicon Valley Online Bootcamp * 활동기간 : 2021.07 ~ 2021.08 * 내용 : 실리콘밸리 엔지니어가 직접 운영하는 실리콘밸리 기술 스택 기반, 실무 개발 프로젝트 부트캠프   페어프로그래밍 기반으로 5명을 팀을 이뤄  Next.JS 기반 TTS 기술을 이용한 웹 페이지를 제작 (프론트 2 백엔드 3)     * 활동명 : AI 프리인턴쉽, POSCO * 교육기관 : 성남시 청소년 재단 * 활동기간 : 2021.06 ~ 2021.07 * 내용 : 성남시 주최 실리콘밸리 기술 스택 기반 AI 접목 프로젝트 부트캠프   페어프로그래밍 기반으로 6명이 팀을 이뤄  React 기반 팀 일정관리 웹 페이지를 제작 (프론트 2 백엔드 4) | |
| --- | --- | --- |

| 학력 사항 | 2015. 03 - 2022. 02 | | 한국산업기술대학교 | 메카트로닉스공학과 졸업(학사) |
| --- | --- | --- | --- | --- |

수행 프로젝트

|  | **React Native 기반 학교 외박신청 어플리케이션(2021.05 ~2021.09)**  매일매일 외박신청을 수동으로 해야하는 불편한 로직을 한 번 클릭으로 해결하는 앱 서비스  현재 학교 학생 400명 이상 이용중 (프론트 1, 백엔드 1)  **GitHub 링크**   | PoC | <https://github.com/ww8007/KPU_AUTO_RN> | | --- | --- | | Product | <https://github.com/AUTO-Overnight/Auto_Overnight> |   **Store 링크**   | 플레이스토어 | [한국공학대학교 외박 신청 - Apps on Google Play](https://play.google.com/store/apps/details?id=com.ww8007.AutoOvernight) | | --- | --- |   **데모 ID**   | ID | 2018150030 | | --- | --- | | PW | toddlf0829\_ |   **기술 선정**   * React Native + Expo : 실질적인 스토어 배포와 IOS 환경의 경우 까다로운 배포 환경을 고려하여 선정 * TypeScript : 컴포넌트 Props 가시적인 전달 및 Redux Action type에서의 직관적인 추론에 용이 * Redux + Redux-Saga : Redux Persist 기능을 이용한 로그인 정보 및 데이터 저장 및 Devtools를 이용한 효율적인 디버깅 * serverless : DB가 필요 없는 환경으로 Aws Lambda 서비스를 이용한 API 연동   **기능 개발**   * 학생 요청 사항의 자동 로그인 기능을 위한 Redux Persist 도입   + login 상태를 저장하여 어플리케이션 종료 후에도 저장된 로그인 정보 이용하여 자동 로그인 요청 * 요청 사항의 외박신청을 달력 형식으로 하는 것과 성공/실패 여부 모두 표현을 위해 달력으로 변경   + 1일, 1주, 2주, 4주 형식의 신청, 성공/실패, 터치 토글 기능 구현 * 화이트 / 다크 테마 지원 * 기존 PoC 버전 API call 30초 이상을 1초 이내로 해결   + 학교 외박신청 API 패킷 분석하여 API 직접 이용 * 지속적인 유지 보수 중   + 날씨, 셔틀 버스 시간, 최종 설정일 기준 알람 설정 기능들 추가 및 리팩토링 진행 | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **스크롤 애니메이션 기반 NPM 모듈 배포(2022.04 ~ 2022.04)**  웹 페이지에 필수적으로 들어가는 스크롤 애니메이션을 모듈화여 배포(프론트 7)  **Github 링크**   | Module | <https://github.com/nice-people-meeting/scroll-animations-js-web> | | --- | --- | | Example | <https://github.com/nice-people-meeting/scroll-animations-js> |   **사용 링크**   | Example | [Scroll Animations JS](https://scroll-animations-js-web.vercel.app/) | | --- | --- | | Npm | <https://www.npmjs.com/package/scroll-animations-js> |   **기술선정**   * React : 예제 페이지 구성 및 배포 사용 * TypeScript : 모듈 배포 과정에서의 타입 추론 목적 및 협업 과정에서의 필요성   **기능 개발**   * 애니메이션 스크롤 이벤트 감지를 위한 Intersection Observer API 사용   + 모듈 배포 관점의 렌더링 성능 이슈를 고려하여 선정 * 콜백을 통한 애니메이션 클래스 추가 및 제거 용이   + Example 페이지 기준 Lighthouse 성능 99점 * css 성능 최적화를 위한 transition 사용   + composition 단계의 transition만 일어나도록 css 선정 * 사용 예제를 위한 예제 페이지 구성   + 옵션으로 설정 가능한 delay, duration 및 fade, scale에 대한 예제들 제공 | |
|  | **React Native 기반 일정 관리 어플리케이션 (2021.06 ~2022.02)**  팀원 간 모임 일정 관리 애플리케이션 서비스 (프론트1, 백엔드 1)  **Github 링크**   | PoC | <https://github.com/SeongNam-AI-PreInternship-Team-E/TeamEet> | | --- | --- | | Product | <https://github.com/Join2Gather/WeMeet> |   **Store 링크**   | 앱스토어 | <https://apps.apple.com/kr/app/we-meet/id1610966012> | | --- | --- | | 플레이스토어 | <https://play.google.com/store/apps/details?id=com.ww8007.weMeet> |   **기술 선정**   * React Native : JS 기반으로 IOS, Android 배포를 위해 사용 * Expo : Pure React Native 대비 좀 더 쉬운 빌드 환경을 위해 사용 * Redux : Redux-toolkit + Devtools를 이용해 좀 더 적은 보일러플레이트 코드 및 전역 상태 관리 * Redux-Saga : Iterator 기반의 비동기 요청에 대한 순서 보장을 위해 사용 및 mocking test 용이 * Jest : Redux action 관련 테스트 진행 및 코드 리팩토링 * Firebase : FCM을 이용한 Notification 처리 및 Analytics을 이용한 로그 분석   **기능 개발**   * 간편 로그인 기능을 위한 소셜 로그인 구현   + 카카오(웹뷰) + 애플(모듈)을 이용한 좀 더 쉬운 로그인 서비스 구현 * 추후 유지 보수를 고려한 직관적인 코드를 위한 함수형 프로그래밍 도입(리팩토링)   + week 데이터 생성의 지연평가 개념을 이용한 파이프라인 구조로 변경 * 사용자의 취향을 고려한 테마 색상 설정 및 편의성 목적의 Native 알람 기능 제공   + redux, 모듈을 이용한 테마 색상 및 Native 켈린더 기반으로 알림 설정 기능 제공 * 리팩토링 과정의 개발 과정의 재사용성 증대를 위한 스타일 컴포넌트 분리   + 버튼, 모달, 헤더와 같은 테마 컴포넌트 분리하여 사용 * 어플리케이션의 핵심 기능을 담당하는 API 및 redux action 테스트 코드 작성   + Saga 기반 Mocking test 진행 및 action 로직 테스트 | |
|  | **React Native 기반 주차 신청 어플리케이션(2021.01 ~ 2021.02)**  수기나 전화를 통해 신청하던 주차 신청 로직을 앱을 이용해 간소화 (개인 프로젝트)  **GitHub 링크**   | Github | <https://github.com/ww8007/ParkingApp_Firebase> | | --- | --- |   **기술 선정**   * React Native + Expo : 사용 대상 한정과 IOS, Android 동시 배포를 위한 Expo 환경 선택 * Redux : 전역 상태 관리 및 Redux-Persist를 이용한 * Firebase : FCM을 이용한 Notification과 Authentication 관리   **기능 개발**   * no-sql 특화의 Firebase를 이용한 유저 데이터 관리   + FCM을 이용한 주차 신청 성공/실패 및 관리자 알람 기능 구현   + Authentication 서비스를 이용한 가입 및 접근 권한 관리 * 일반 사용자 / 관리자 모드 분리하여 구현   + 최초 로그인 기점으로 관리자 판단하여 서로 다른 화면 및 기능 사용 가능 하도록 설정 * 주차 신청 API 이용   + 실제 주차 관리 서비스 API 패킷 분석 하여 Request/Response에 사용 * 주차 신청 요청 신청 성공/실패 여부 실시간성 데이터 보장   + Realtime DB를 이용하여 동적 신청 여부 알람 처리 및 화면 업데이트(아이콘, 색) | |
|  | **Next.js 기반 TTS 웹 페이지 (2021.07 ~ 2021.08)**  태연의 목소리로 다른 가수의 노래를 재생시켜주고, 텍스트 입력을 받아서 KSS와 태연의 목소리로 변환하여서 재생을 시켜주는 사이트를 제작 (프론트 2, 백엔드 3)  **GitHub 링크**   | Github | <https://github.com/SiliconWildCat/SiliconWildCat> | | --- | --- |   **배포 주소 (페이지만 동작 중)**   | Example | <https://distracted-turing-cc464a.netlify.app/> | | --- | --- |   **기술 선정**   * Next.JS : SSG 기반으로 정적 HTML 제공을 위한 빠른 페이지 제공 * TypeScript : 협업 과정의 타입 제공과 Redux action 타입 보장 * Redux + Redux-Saga : Devtools를 이용한 디버깅   **기능 개발**   * 화면 전환 애니메이션 구현   + Sass를 이용한 화면 분할을 이용한 페이지 전환 구현 * SPA의 단점인 처음 빈화면 렌더링 해결을 위한 SSG 도입   + 초기 HTML 파일 제공으로 사용자 입장 최적화 제공 | |