

교육 과정 소개서.

사이드 프로젝트 : 10개 기술스택으로 구현하는 풀스택
서버리스 프로젝트 with React



강의정보

강의장	온라인 강의 데스크탑, 노트북, 모바일 등
수강 기간	평생 소장
상세페이지	https://fastcampus.co.kr/dev_online_spdutch
강의시간	36시간 40분
문의	고객센터

강의특징

나만의 속도로	낮이나 새벽이나 내가 원하는 시간대 에 나의 스케줄대로 수강
------------	---

원하는 곳 어디서나	시간을 쪼개 먼 거리를 오가며 오프라인 강의장을 찾을 필요 없이 어디서든 수강
---------------	---

무제한 복습	무엇이든 반복적으로 학습해야 내것이 되기에 이해가 안가는 구간 몇번이고 재생
-----------	--



강의목표

- 풀스택 서버리스 환경에서 추상적인 아이디어를 서비스로 구체화하는 방법을 학습합니다.
- 동일한 프로젝트 결과물을 얻기 보다, 실무 개발 프로세스를 체화하는 데 집중합니다.
- 개발 과정에서 문제에 직면했을 때, 적절한 해결 방법을 찾아내고 이를 실행하는 능력을 향상시킵니다.
- 배운 내용을 복습 및 응용하고, 추가 기능을 구현하는 파트 별 과제와 심화 실습을 통해 스스로 문제를 해결할 수 있는 역량을 길러 봅니다. (* 퀴즈, 실습형 과제 해설 영상 제공)
- 질의응답 게시판에서 서로 모르는 것을 묻고 답하고, 서로의 결과물을 공유하고, 함께 프로젝트를 진행할 팀원을 구해보세요.

강의요약

- 하나의 프로젝트, 그 시작부터 끝까지 전체과정 (기획-설계-구현-배포-기능 추가)을 경험합니다.
- 기술을 언제, 어떻게, 왜 적용해야 하는지 이해해 트렌디한 기술을 온전히 자신의 것으로 만듭니다.
- 선배 개발자의 업무 루틴, 사고 방식을 익혀 문제해결 능력을 갖춘 실무 개발자로 거듭납니다.
- 강사님 & 수강생에게 언제든지 물어볼 수 있는 질의응답 게시판이 운영됩니다. (게시판은 2024년 8월까지 운영됩니다.)



강사

오아리

과목

- 10개 기술스택으로 구현하는 풀스택 서버리스 프로젝트 with React

약력

- 현) 캐나다 의료 스타트업 시니어 풀스택 엔지니어
- 전) 아마존 소프트웨어 엔지니어
- 전) 카카오 모빌리티 소프트웨어 엔지니어
-
- [사이드프로젝트]
- App 'Better' 프로젝트 관리 및 개발
- App 'Lipp' 개발 (APEC App challenge 2019 입상)
- '교육 웹사이트'를 개발해 아마존 사내 해커톤 입상
- '개발자 업무 공유 챗봇' 제작해 카카오 사내 배포



CURRICULUM

01.

**프로젝트
시작하기**

파트별 수강시간 02:57:46

Ch01. 사이드프로젝트 첫 걸음
01. 강사 및 강의 소개
02. 사이드 프로젝트, 왜 해야 하나요?
03. 사이드 프로젝트 주제 정하기, 아무거나 하지 맙시다, 쏘!
04. 1인 개발 VS. 팀 개발, 어떤 것이 좋나요
Ch02. 프로젝트 관리
01. 프로젝트 관리, 개발자가 왜 알아야 하죠
02. 프로젝트를 잘 관리하기 위한 방법 (1)
03. 프로젝트를 잘 관리하기 위한 방법 (2)
04. Notion으로 프로젝트 관리 템플릿 생성하기
05. 소프트웨어 개발 주기
06. 사이드 프로젝트에서 적용할 것들
Ch03. 마무리
01. 파트 마무리 및 Recap
02. [과제 소개] 나의 사이드 프로젝트 찾기



CURRICULUM

02.

**프로젝트
기획하기**

파트별 수강시간 03:04:18

Ch01. 서비스 요구사항
01. 같이 만들어 볼 프로젝트, 더치페이 서비스 소개
02. 서비스 요구사항 작성법
03. 더치페이 서비스 요구사항 작성하기
Ch02. 디자인 기획
01. 디자인 Mock-up 툴 소개
02. 더치페이 디자인 Mock-up 같이 보기
Ch03. 프로젝트 플래닝
01. Task breakdown
01. [과제 해설] Task 내용 작성하기
02. 우선 순위 정하기
03. 릴리즈 날짜 계산하기
Ch04. 마무리
01. 파트 마무리 및 Recap
02. [과제 소개] 더치페이에 내가 원하는 기능도 넣어봅시다



CURRICULUM

03.

**프로젝트
설계하기**

파트별 수강시간 07:58:16

Ch01. 시스템 설계
01. 시스템 설계가 왜 필요한가요 ?
02. [이론] 시스템 설계 (1) - Architecture diagram
03. [이론] 시스템 설계 (2) - Sequence diagram
04. [이론] 시스템 설계 (3) - Class diagram
05. [실전] 시스템 설계 (1) - Architecture diagram
06. [실전] 시스템 설계 (2) - Sequence diagram
07. [실전] 시스템 설계 (3) - Class diagram for 프론트 엔드
Ch02. 기술셋 선정하기
01. 기술셋을 선정하는 기준
02. 기술셋 선정하기 (1) - 프론트엔드
03. 기술셋 선정하기 (2) - DevOps
04. AWS Amplify 소개
05. React 소개
06. React Router 소개
07. Recoil 소개
08. Styled 소개
09. React Testing Library 소개
10. Jest 소개
Ch03. TDD - 테스트 주도 개발
01. TDD(Test-Driven Development)를 활용하기
02. 테스트 작성의 정석 - 의미 있는 테스트 작성하기
Ch04. 마무리
01. 파트 마무리 및 Recap
02. [과제 소개] 중간 점검! 퀴즈 타임
03. [과제 해설] 중간 점검! 퀴즈 타임



CURRICULUM

04.

**비용 정산 사이트
구현하기**

파트별 수강시간 10:49:49

Ch 01. 개발 환경 구축
01. AWS Amplify를 이용한 개발 환경 셋업
02. Amplify와 Github 연동
Ch 02. 레이아웃 구현
01. 디자인을 토대로 레이아웃 구현하기
02. Bootstrap layout 소개
03. CSS flex 소개
Ch 03. 그룹 생성 컴포넌트 구현
01. 그룹 생성 컴포넌트 구현 (1) - UI 테스트 코드 및 테스트 환경 셋업
02. 그룹 생성 컴포넌트 구현 (2) - React component
03. 비용 입력 컴포넌트 구현 (3) - Styling
Ch 04. 멤버 추가 컴포넌트 구현
01. 멤버 추가 컴포넌트 구현 (1) - UI 테스트 코드
02. 멤버 추가 컴포넌트 구현 (2) - React component
03. 멤버 추가 컴포넌트 구현 (3) - Styling 및 리팩토링
Ch 05. 정산 내역 입력 컴포넌트 구현
01. 비용 입력 컴포넌트 구현 (1) - UI 테스트 코드
02. 비용 입력 컴포넌트 구현 (2) - React component
03. 비용 입력 컴포넌트 구현 (3) - Styling
Ch 06. 정산 리스트 컴포넌트 구현
01. 정산 리스트 컴포넌트 구현 (1) - UI 테스트 코드
02. 정산 리스트 컴포넌트 구현 (2) - React
03. 정산 리스트 컴포넌트 구현 (3) - Styling
Ch 07. 정산 결과 조회 컴포넌트 구현
01. 정산 결과 조회 컴포넌트 구현 (1) - UI 테스트 코드
02. 정산 결과 조회 컴포넌트 구현 (2) - React component
03. 정산 결과 조회 컴포넌트 구현 정산 결과 조회 컴포넌트 구현 (3) - Styling
Ch 08. 컴포넌트 조합하기
01. 페이지 링크 생성 - React Router
Ch 09. 마무리
01. 파트 마무리 및 Recap
02. [과제 소개] 정산 결과 이미지 파일로 export 하기
03. [과제 해설] 정산 결과 이미지 파일로 export 하기



CURRICULUM

05.

서비스 배포 하기

파트별 수강시간 03:53:12

Ch 01. 배포 준비
01. 배포 프로세스 수립하기
Ch 02. QA 테스트
01. [이론] Quality Assurance 테스트 (1) - cross-browser testing
02. [이론] Quality Assurance 테스트 (2) - functional testing
03. [실전] 테스트 환경 구축
04. [실전] Quality Assurance 테스트 (1) - cross-browser testing
05. [실전] Quality Assurance 테스트 (2) - functional testing
Ch 03. 서비스 배포
01. AWS Amplify를 통해 서비스 배포하기
02. 서비스 배포 상태 확인하고 모니터링하기
03. 서비스 도메인은 어디에?
Ch 04. 마무리
01. 파트 마무리 및 Recap
02. [과제 소개] AWS Amplify를 이용해서 백엔드 환경 설정하기



CURRICULUM

06.

[Advanced]
서비스 디벨롭
하기

파트별 수강시간 07:56:59

Ch 01. 디벨롭 준비
01. 첫 번째 릴리즈가 끝나면 해야 할 것들
02. 요구사항 소개 - 공유용 URL 생성 기능 추가하기
Ch 02. 시스템 설계 업데이트
01. 시스템 설계 업데이트 (1) - Architecture diagram
02. 시스템 설계 업데이트 (2) - Sequence diagram
Ch 03. 백엔드 기능추가
01. [백엔드] 기술셋 선정하기
02. [백엔드] 데이터베이스 (1) - 기술셋 선정
03. 데이터베이스(2) - 스키마 설계
04. [백엔드] AWS Amplify를 이용한 환경 구축
05. [백엔드] 프로젝트 구조 설명
06. [백엔드] 데이터베이스 Dynamo DB 연동
07. [백엔드] API 구현 (1) - 그룹 생성
08. [백엔드] API 구현 (2) - 멤버 추가
09. [백엔드] API 구현 (3) 정산 내역 업데이트
10. [백엔드] API 구현 (4) - 그룹 읽기
Ch 04. 프론트엔드 기능추가
01. [프론트엔드] API 연동 (1) - 그룹 생성
02. [프론트엔드] API 연동 (2) - 멤버 추가
03. [프론트엔드] API 연동 (3) - 정산 내역 업데이트
04. [프론트엔드] API 연동 (4) - 그룹 읽기
05. [프론트엔드] 공유 버튼 컴포넌트 구현 (1) - UI testing
06. [프론트엔드] 공유 버튼 컴포넌트 구현 (2) - React component
07. [프론트엔드] 공유 버튼 컴포넌트 구현 (3) - Styling
Ch 05. 테스트와 배포
01. 서비스 테스트 및 배포
Ch 06. 마무리
01. 파트 마무리 및 Recap
02. 강의 마무리 멘트



주의 사항

- 상황에 따라 사전 공지 없이 할인이 조기 마감되거나 연장될 수 있습니다.
- 패스트캠퍼스의 모든 온라인 강의는 아이디 공유를 금지하고 있으며 1개의 아이디로 여러 명이 수강하실 수 없습니다.
- 별도의 주의사항은 각 강의 상세페이지에서 확인하실 수 있습니다.

수강 방법

- 패스트캠퍼스는 크롬 브라우저에 최적화 되어있습니다.
- 사전 예약 판매 중인 강의의 경우 1차 공개일정에 맞춰 '온라인 강의 시청하기'가 활성화됩니다.



환불 규정

- 온라인 강의는 각 과정 별 '정상 수강기간(유료수강기간)'과 정상 수강기간 이후의 '복습 수강기간(무료수강기간)'으로 구성됩니다.
- 환불금액은 실제 결제금액을 기준으로 계산됩니다.

수강 시작 후 7일 이내	100% 환불 가능 (단, 수강하셨다면 수강 분량만큼 차감)
수강 시작 후 7일 경과	정상(유료) 수강기간 대비 잔여일에 대해 환불규정에 따라 환불 가능

※ 강의별 환불규정이 상이할 수 있으므로 각 강의 상세페이지를 확인해 주세요.