

강의명(가안)	클라우드 네이티브 애플리케이션 부트캠프	강의난이도(택1)	초급
수강 대상	<p>강사님께서 예상하시는 수강 대상에 대해 작성 부탁드립니다.</p> <p>1 기획자/개발자: 고객 중심으로 클라우드 네이티브 애플리케이션의 기획/설계/개발 전 과정을 체험하고 싶은 기획자</p> <p>2 스타트업: 변화에 민첩하고 안정적이며 비용도 절감할 수 있는 서비스를 만들고 싶은 스타트업 대표나 개발자</p> <p>3 이직자/직무전환자: 클라우드 네이티브 애플리케이션 설계 및 개발 역량이 필요한 이직 및 직무전환 희망자</p> <p>4 * 권장 기술 역량: Vue.js, React, jQuery, Spring Boot, Node.js, Python 중 1개 이상 개발 가능자(초급 이상) * 위 개발 언어를 모르셔도 수강 가능함(원하시는 분들에게 별도 비용 없이 별도 시간에 코칭 제공함)</p>		
예상 수강 대상의 현재 문제점	<p>강사님의 경험을 바탕으로 현재 해당 강의가 필요한 수강생들의 문제점을 자유롭게 작성 부탁드립니다.</p> <p>1 개발은 하고 있는데, 애플리케이션이 어떻게 설계되고 어떤 기술들이 포함되어 있는지 모르겠다.</p> <p>2 성공확률이 높은 서비스를 어떻게 만들어야 할지 모르겠다.</p> <p>3 클라우드 네이티브 애플리케이션 공부를 어떻게 시작하고 무엇을 공부해야 할지 모르겠다.</p> <p>4 내용을 기입해주세요.</p>		
위의 문제점을 토대로 우리 강의가 꼭 필요한 이유	<p>※ 위의 문제점을 바탕으로 우리 강의에서 해결할 수 있는 부분 혹은 결과물에 대한 작성 부탁드립니다.</p> <p>1 클라우드 네이티브 애플리케이션의 내부와 외부 아키텍처를 설계하는 방법을 이해하고 실습할 수 있음</p> <p>2 서비스의 성공확률을 높이기 위해 고객 중심으로 서비스를 기획/설계/개발하고 검증하는 방법을 배울 수 있음</p> <p>3 서비스의 기획/설계/개발에 필요한 도구와 기법을 거시적으로 이해하고 중요 과정은 미시적으로 자세히 배우고 실</p> <p>4 내용을 기입해주세요.</p>		
	<p>※ 본 강의를 통해 수강생이 강화할 수 있는 역량, 강의의 특징 및 강의 진행 방식에 대해 작성 요청 드립니다.</p> <p>1 클라우드 네이티브 애플리케이션에 필요한 도구와 기법 중 중요한 과정은 깊게 덜 중요한 과정은 핵심개념만 기억</p>		

과정 특징	<p>2 고객의 문제를 정의하고 그 솔루션을 만들고 검증하여 고객이 원하는 서비스를 만드는 방법론과 기법을 이해하고 실습할 수 있음</p> <p>3 실제 서비스를 직접 완성해 가면서 클라우드 네이티브 애플리케이션의 기획/설계/개발을 현실감 있게 체험하여 이해도를 높일 수 있음</p> <p>4 내용을 기입해주세요.</p>
수강 효과	<p>※ 본 강의를 수강한 이후, 수강생분들이 가져갈 수 있는 스킬셋 혹은 지식 등을 설명해주세요.</p> <p>1 고객 문제 정의 및 검증 기법인 고객경험 인터뷰/유저저니맵/5WHY/문제검증 인터뷰/JTBD(Jobs To Be Done) 프레임워크, 솔루션 구상 및 검증 기법인 Design Thinking/린캔버스/솔루션 검증 인터뷰, 사용성 검증 기법인 제품검증 인터뷰의 핵심을 이해하고 가이드 및 예시를 제공 받을 수 있음</p> <p>2 클라우드 네이티브 애플리케이션의 설계/개발/배포에 필요한 DDD(Domain Driven Design), 유저스토리, 외부 아키텍처 설계(MSA, Service Mesh, Spring Cloud, Istio), 내부 아키텍처 설계(Hexagonal Architecture, Layered Architecture, UML), 마이크로서비스 개발, 빌드 및 배포(컨테이너와 쿠버네티스, CI/CD, Helm)에 대한 기술적 지식을 배우고 실습을 통해 이해도를 높일 수 있음</p> <p>3 애자일, 마이크로서비스, 데브옵스, 클라우드에 대한 개념적 이해와 스크럼/칸반으로 개발하는 방법을 배울 수 있음</p> <p>4 내용을 기입해주세요.</p>
우리 강의의 차별점	<p>※ 국내 유일, 오프라인 유일 혹은 타사의 강의와 비교했을 때, 우리 강의에서만 얻어갈 수 있는 특징점</p> <p>1 최근 5년 15회 이상의 서비스 기획/설계/개발을 리딩하고 누적 방문수 61만의 블로그 '온달의 해피클라우드'를 운영중인 전문 코치가 진행</p> <p>2 서비스의 기획/설계/개발의 전 과정을 End to End로 배울 수 있는 국내 유일의 교육</p> <p>3 기업 대상 코칭 경험을 바탕으로 실질적 도움이 있는 가이드와 풍부한 예제를 제공</p> <p>4 내용을 기입해주세요.</p>

커리큘럼			
Day	강의 목표	이론	실습
1(개강)	일하는 방식 변화를 위한 애자일, 마이크로서비스, 데브옵스, 클라우드에 대한 기본적 이해	<ul style="list-style-type: none"> ○ 애자일 사상 이해 <ul style="list-style-type: none"> - 애자일 핵심 사상 'M'과 애자일 마인드셋 - 일하는 방식 변화: 조직구조, 의사결정, 업무방식, 조직문화 - 애자일 방법론 기본: 린스타트업, 스크럼, 칸반 ○ 마이크로서비스와 MSA 기본 이해 ○ 데브옵스 이해: 데브옵스 조직과 CI/CD ○ 클라우드 이해: 클라우드 분류, 컨테이너, 쿠버네티스 기본 이해 <p>일하는 방식 변화 프레임워크</p>	내용을 기입해주세요.
2	1. 사용자 중심의 문제 정의 및 검증: 현상 문제에 대한 근본 문제 가설을 수립하고 고객과 검증하고 피보팅하는 방법 이해 2. 솔루션 구상 및 비즈니스 모델 기획: 고객과의 공감과 이해를 통해 근본 문제를 해결하는 아이디어를 도출하는 방법과 서비스의 핵심 내용을 9가지 영역으로 나누어 정리하는 린캔버스에 대한 이해	<ul style="list-style-type: none"> 1. 사용자 중심의 문제 정의 및 검증 ○ 근본문제 정의의 중요성과 문제 정의 단계 이해 ○ 문제가설 수립 방법 <ul style="list-style-type: none"> - 정보 수집 방법: 유저저니맵, 고객경험 인터뷰 - 문제가설 수립 방법: 5WHY - 과학적 니즈 도출 방법: JTBD(Jobs To Be Done) 프레임워크 ○ 문제검증 방법: 문제검증 인터뷰 수행, 수행 결과 분석, 문제가설 피보팅 <ul style="list-style-type: none"> 2. 솔루션 구상 및 비즈니스 모델 기획 ○ 고객과의 공감과 이해 (Understand & Explore) <ul style="list-style-type: none"> - 고객 유형 및 페르소나 정의 - Empathy Map(공감맵)과 User Journey Map(고객경험 여정 맵)으로 고객 이해 - 근본문제와 요구사항 정의 ○ 문제해결 아이디어 도출 (Prototype & Evaluate) <ul style="list-style-type: none"> - 문제 해결 Idea 도출 및 우선순위 평가 - To-Be User Journey Map과 Storyboard로 변화되는 사용자 경험 표현 ○ 비즈니스 모델 기획 <ul style="list-style-type: none"> - 린캔버스 9가지 영역에 대해 이해 - 해적지표(AARRR) 기반의 핵심 지표 도출법 - 다양한 린캔버스 예시를 통한 작성법 이해 	디자인씽킹의 문제해결 Idea 도출 및 우선순위 평가
3	To-Be 고객 경험 여정을 구체화하고 필요한 기능, 데이터, 연동시스템, 정책/규정 등을 식별하는 방법을 이해	<ul style="list-style-type: none"> ○ DDD(Domain Driven Design) 방법론에 대한 이해 ○ Event Storming 수행 절차 <ul style="list-style-type: none"> - 도메인 이벤트 식별과 고객 여정 흐름에 대한 공감과 이해 - 고객 여정에 필요한 외부시스템, 정책/규정, 데이터 식별 - Bounded Context (동일한 비즈니스 주제별 그룹) 정의 ○ 다양한 예시를 통한 Event Storming 이해 	<ul style="list-style-type: none"> ○ Event Storming 기법을 이용한 예제 서비스 기획 구체화
		사용자 중심 요구사항 정의, 솔루션 검증	

4	<p>1. 사용자 중심 요구사항 정의: 사용자 관점에서 서비스에 필요한 요구사항을 정의하는 유저스토리 작성법 이해</p> <p>2. 솔루션 검증: 낮은 수준의 MVP(Minimum Viable Product)를 이용하여 고객과 함께 솔루션 가설을 검증하는 방법 이해</p>	<p>1. 사용자 중심 요구사항 정의</p> <ul style="list-style-type: none"> o Bounded Context를 마이크로서비스로 정의하기: 데이터 영향도, 서비스 중요도, 부하집중도, 서비스크기를 고려한 정의 방법 o 유저스토리, 비즈니스 중요도, 테스트 시나리오 정의 <ul style="list-style-type: none"> - 유저스토리 작성 문법 Tip - 비즈니스 중요도 분류 방법 MosCow - 테스트 시나리오 작성 문법과 Tip o 유저스토리 스코어링 <ul style="list-style-type: none"> - 스코어링 목적과 수행법 - 삼각측량법을 이용한 속도 높이기 <p>2. 솔루션 검증</p> <ul style="list-style-type: none"> o 솔루션 검증 목적과 수행 단계 이해 o 솔루션 검증 인터뷰지 작성법 o 솔루션 검증 방법: 인터뷰 수행, 수행결과 분석, 솔루션 피보팅 	<ul style="list-style-type: none"> o 예제 서비스의 유저스토리 작성 o 유저스토리 스코어링
5	Service Mesh와 Event Driven Architecture를 이해하고 예제 서비스의 외부 아키텍처를 설계함	<p>외부 아키텍처 설계</p> <ul style="list-style-type: none"> o MSA와 Service Mesh 이해 <ul style="list-style-type: none"> - MSA 구성 컴포넌트: 인프라, 컨테이너관리, Runtime환경, Service Mesh, 공통 서비스(Backing Services), CI/CD, 애플리케이션 - N-Tier Layered Architecture o Service Mesh와 프레임워크인 Spring Cloud/Istio 이해 o EDA(Event Driven Architecture) 이해 	<ul style="list-style-type: none"> o Spring Cloud 실습: Gateway, Eureka, Config o 예제 서비스의 외부 아키텍처 설계
6	내부 아키텍처를 설계하는 방법을 이해하고 예제 서비스의 내부 아키텍처를 설계함	<p>내부 아키텍처 설계</p> <ul style="list-style-type: none"> o 레이어드(Layered)/헥사고널(Hexagonal)/클린(Clean) 아키텍처 이해 o Class Diagram, Sequence Diagram, Data 설계, REST API 설계 	마이크로서비스 개발1
7	예제 서비스 개발을 통해 아키텍처 설계가 코드에 어떻게 반영되는지 이해함	<ul style="list-style-type: none"> o Spring Boot 기반의 애플리케이션 개발 <ul style="list-style-type: none"> - Swagger를 이용한 API 테스트 페이지 개발 - Mybatis와 JPA를 이용한 Data Access 방법 이해 	<ul style="list-style-type: none"> o 개발환경 구성: IntelliJ, Git, DB 등 o Swagger를 이용한 API 테스트 페이지 개발 o Mybatis를 이용한 개발 실습 o JPA를 이용한 개발 실습
8	예제 서비스 개발을 통해 아키텍처 설계가 코드에 어떻게 반영되는지 이해함	<p>마이크로서비스 개발2</p> <ul style="list-style-type: none"> o 예제 서비스 소스 설명: 프론트엔드 & 백엔드 	예제 서비스 개발 및 테스트
9	마이크로서비스 주요 패턴을 이해하고 Axon Framework을 이용하여 API Composition과 CQRS패턴을 개발함	<p>마이크로서비스 패턴 개발</p> <ul style="list-style-type: none"> o 마이크로서비스 주요 패턴 이해: API Composition, CQRS, Saga, Event Sourcing, External API, Transactional Outbox/Polling publisher/Transaction Log tailing o Axon Framework 이해 	예제 서비스에 API Compostion, CQRS 패턴 개발
10	클라우드 네이티브 애플리케이션을 컨테이너화하기 위한 컨테이너 기술 이해	<p>컨테이너 기술 이해</p> <ul style="list-style-type: none"> o 컨테이너 이해 o 컨테이너 제품과 도커 아키텍처 이해 	<ul style="list-style-type: none"> o 예제 서비스의 컨테이너 이미지 만들기 o 예제 서비스의 컨테이너 배포
11	컨테이너를 관리하기 위한 쿠버네티스 이해	<p>쿠버네티스 이해1</p> <ul style="list-style-type: none"> o 쿠버네티스와 쿠버네티스 아키텍처 이해 o 한장으로 이해하는 쿠버네티스 리소스 	예제 서비스의 쿠버네티스 환경 배포
12	컨테이너를 관리하기 위한 쿠버네티스 이해	<p>쿠버네티스 이해2</p> <ul style="list-style-type: none"> o Health Check, Resource 조정, Auto Scaling o 인증/권한관리, 통합로깅, 무중단 배포, 통합 모니터링 	<ul style="list-style-type: none"> o Failover & Self Healing o 인프라 자원 자동 증감 o Auto Scaling
			CI/CD

13	마이크로서비스의 빌드 및 배포를 자동화하는 방법 이해	<ul style="list-style-type: none"> ○ CI/CD 파이프라인과 Tool Chains 이해 ○ 예제 CI/CD 파이프라인 설명 ○ Helm chart 사용법 	<ul style="list-style-type: none"> ○ Jenkins 기반의 CI/CD <ul style="list-style-type: none"> - 예제 서비스 배포를 위한 k8s yaml 파일 작성 - 기본 CI/CD파이프라인 제작 - 소스정적검사, Git연결 자동화, Slack연결 추가
14(증강)	<p>1. 제품 검증: 개발된 서비스가 고객의 문제를 해결 또는 완화해 줄 수 있는</p>	<p style="text-align: center;">제품 검증, 스크럼/칸반</p> <p>1. 제품 검증</p> <ul style="list-style-type: none"> ○ 제품 검증 목적과 수행 단계 이해 ○ 제품 검증 인터뷰지 작성법 ○ 제품 검증 방법: 인터뷰 수행, 수행결과 분석, 솔루션 피보팅 <p>2. 스크럼/칸반</p> <ul style="list-style-type: none"> ○ 데브옵스 조직 구성원의 역할과 마음가짐 ○ 스크럼 수행 가이드: Planning, DSU, Review, Retrospective ○ 칸반 수행 가이드: 스크럼과 칸반 유사점과 차이점, 칸반보드 작성법 	내용을 기입해주세요.

강사 소개

성명	이해경
주요 경력	<p>1) 학력 동국대학교 정보관리학 학사</p> <p>2) 회사/직무명</p> <ul style="list-style-type: none">- 유니콘주식회사/대표이사: 2023.05 ~ 현재- 한국IBM/컨설턴트, PM, 아키텍트, 애자일코치: 2007.02 ~ 2023.04- 코디얼/개발팀장: 2001.08 ~ 2005.10- 로터스코리아/개발자: 1999.08 ~ 2001.06- 동양시스템하우스/개발자: 1997.02 ~ 1999/07 <p>3) 기타 등등</p>
	<p>1) 강의 이력</p> <ul style="list-style-type: none">- IBM 지원 산업고등학교(P-Tech) Design Thinking 워크샵(2회): 2024.4, 2023.4- KB 플랫폼 부서원 대상 Chapter 활성화 방안 강의: 2023.8- KB PO(Product Owner) 대상 서비스 기획 워크샵: 2023.7

프로젝트 및 기타 활동

- 2) 논문 게재
- 3) 수상 내역
- 4) 강의 관련 프로젝트 진행 이력
 - KB 신입행원 DigiTech Bootcamp(4회): 2021~2023
 - 현대해상 U2C(Unix to Container) PoC: 2023.9
 - KB Wallet Agile Coaching (자동차검사예약 서비스 개발): 2022.12
 - KB 수신상품부 Agile Coaching (국민건강적금 개발): 2022.10
 - KB 기업뱅킹 Agile Coaching(부동산 임대료 관리 서비스 개발): 2022.5
 - KB 마이데이터 Agile Coaching(자산 포트폴리오 공유 및 상품 추천 서비스 개발): 2021.5
 - NS홈쇼핑 클라우드 네이티브 애플리케이션 부트캠프: 2021.3
 - MetLife 마이크로서비스 부트캠프(보험 가입설계 및 청약 서비스 개발): 2020.10
 - SKT Container Platform BMT: 2020.3
 - HANA은행 IBM Cloud Garage 2회: 2019.8, 2020.1
- 5) 기타 등등
 - 블로그 '온달의 해피클라우드' 운영중: 2024년 4월 현재 누적 방문수 61만명
 - '마이크로서비스 패턴 쉽게 개발하기' 출간: 2024.2

인터뷰

1. 자기소개 부탁드립니다.

안녕하세요?

저는 자신이 맞다고 생각하는 믿음을 실천하고, 다른 사람과 소통하고 협업하면서 일하고, 끊임없는 도전과 실패를 통해 성장하는 애자일 사상이 우리가 인생을 사는 훌륭한 철학이라고 믿습니다.

제 꿈은 더 많은 사람이 애자일하게 살고 일하고 도전하는 세상을 만드는 것입니다.

개발자로 시작하여 프로젝트관리자와 애플리케이션 아키텍트로 20여년을 보냈습니다.

PM과 아키텍트로 여러 프로젝트를 하면서 제 가장 큰 고민은 어떻게 하면 사용자를 위한 더 좋은 서비스를 만들 수 있을까요. 모험적으로 몇 개 프로젝트를 스크럼으로 진행해 보니 사용자 피드백을 반영하느라 개발이 힘들다는 단점이 있었지만, 좀 더 사용자가 원하는 서비스를 만들 수 있다는 확신을 얻었습니다.

이러한 경험이 애자일 코치로의 변신으로 이어졌고 지금은 애자일 사상에 기반하여 서비스를 기획/설계/개발하는 것을 도우며 살고 있습니다.

IBM에서 KB, 하나은행, MetLife, NS홈쇼핑, 코스콤 등의 고객들을 대상으로 15회 이상의 부트캠프와 실제 서비스 개발을 리딩 하였습니다.

실제 출시한 서비스로는 **マイ데이터 기반의 자산 포트폴리오 공유 서비스, 임대료 관리 서비스, 건강적금, 자동차검사 예약 서비스**가 있습니다.

그 경험으로부터 제가 배운 것은 서비스는 문제를 올바르게 정의하는 것이 가장 중요하다는 것입니다.

또한 변화에 민첩하고 안정적이며 비용도 절감할 수 있는 서비스가 되려면 **マイ크로서비스와 컨테이너화**가 정말 필요하다고 느꼈습니다.

저는 지금은 IBM을 나와 유니콘이라는 회사를 경영하고 있습니다.

제가 IBM을 나온 이유는 제품을 기반으로 한 서비스 개발보다 고객이 진정으로 원하는 서비스를 애자일 사상에 기반하여 개발하는 것 이 더 맞다고 믿었고, 그간의 경험을 더 많은 사람들에게 전달하고 싶어서였습니다.

이런 모험을 결심할 수 있었던 건 능력 있는 와이프와 어느새 다 커버린 아들과 딸의 전폭적인 지지가 큰 힘이 되었다는 말도 꼭 하고 싶습니다.

2. 강의에서는 주로 어떤 내용을 다를 예정이시고, 어떤 형태로 진행되나요?

부트캠프의 목표는 클라우드 네이티브 애플리케이션(이하 CNA: Cloud Native Application)의 목적인 변화에 민첩하게 대응하고 안정적이며 비용을 절감할 수 있는 서비스의 기획/설계/개발 방법을 쉽고 기억에 남게 제공하는 것입니다.

이번 부트캠프는 주로 개발자를 대상으로 매주 목/토요일 3시간씩 총 14주로 기획하였습니다. 시간으로 계산하면 42시간입니다.

기존에 기업 대상으로 할 때는 이 과정을 최소 160시간 이상으로 진행하였습니다.

짧은 시간에 압축적으로 전달하기 위해 개발자의 실무 활용도와 CNA 목적 부합도에 따라 강좌를 핵심강좌, 지원강좌, 일반강좌로 나누었습니다.

핵심강좌는 개발자의 실무 활용도와 CNA 목적 부합도가 모두 높은 강좌로서 크게 설계, 개발, 배포의 3가지 유형의 강좌가 있습니다.

뉴스처럼 핵심 개념과 수행 방법을 쉽게 전달하고 주로 워크숍과 실습으로 이해도를 높이고자 합니다.

첫째, 마이크로서비스 설계 강좌는 DDD(Domain Driven Design), 유저스토리, 외부 아키텍처 설계(MSA, Service Mesh, Spring Cloud, Istio), 내부 아키텍처 설계(Hexagonal Architecture, Layerd Architecture, UML)입니다.

둘째, 마이크로서비스 개발 강좌는 마이크로서비스 개발, 마이크로서비스 패턴 개발입니다.

셋째, 마이크로서비스 배포 강좌는 컨테이너와 쿠버네티스, CI/CD입니다.

지원강좌는 개발자의 실무 활용도는 낮지만, CNA 목적 부합도가 높은 강좌입니다.

이론강의로 진행되고 광고카피처럼 핵심 개념만 기억에 남게 전달하고 기존 산출물로 이해를 돋겠습니다.

문제정의와 검증 기법 강좌에서는 고객경험 인터뷰, 유저저니맵, 5WHY, 문제검증 인터뷰, JTBD 프레임워크를 다룹니다.

솔루션 구상과 비즈니스 모델 기획 강좌에서는 아이디에이션 기법인 Design Thinking과 비즈니스 모델 수립을 위한 린캔버스를 다룹니다.

솔루션 검증 기법 강좌에서는 개발될 서비스를 미리 검증하는 솔루션 검증 인터뷰에 대해 설명합니다.

사용성 검증 기법 강좌에서는 출시전에 서비스를 미리 검증하는 제품 검증 인터뷰에 대해 얘기하겠습니다.

일반강좌는 개발자의 실무 활용도가 높고 CNA 목적 부합도는 낮은 강좌입니다.

이론강의로 진행되고 쉽고 명확하게 핵심을 전달하겠습니다.

일하는 방식 변화 프레임워크 강좌에서 애자일, 마이크로서비스, 데브옵스, 클라우드에 대해 개략적으로 설명하겠습니다.

3. 본 강의에서 배운 내용을 수강생 분들이 어떻게 활용할 수 있을까요?

부트캠프 대상자의 유형은 비즈니스 파트에 있는 기획자와 IT 파트에 있는 개발자입니다.

최근 많은 기업은 서비스별로 기획자와 개발자를 한 팀으로 구성하고 있습니다.

서비스의 핵심 가치 달성을 목적으로 서로 소통하면서 점진적으로 서비스를 발전시켜 나가기 위해서입니다.

스타트업은 인력이 적어 기획과 개발을 한 사람이 하는 경우도 많습니다.

이렇게 한 팀으로 일하게 되면 서로의 분야에 대해 어느 정도의 이해가 필요합니다.

본 과정을 통해 기획자는 클라우드 네이티브 애플리케이션을 설계, 개발, 배포하는 기술에 대해 핵심을 이해할 수 있습니다.

개발자는 서비스를 기획하는 기법을 이해하여 좀 더 적극적으로 기획 과정에 참여할 수 있을 것입니다.

만약 기획자와 개발자가 다른 조직이라 하더라도 이러한 기법과 기술에 대한 이해는 서로의 소통을 크게 증진 시킬 수 있을 것입니다.

좀 더 구체적으로 설명하면 아래와 같습니다.

첫째, **클라우드 네이티브 애플리케이션의 내부와 외부 아키텍처를 설계하고 개발한 결과를 자동으로 클라우드에 배포할 수 있게 됩니다.**

둘째, 고객 중심으로 서비스를 기획/설계/개발하고 검증하여 서비스의 성공 확률을 높일 수 있게 됩니다.

셋째, 스크럼과 칸반으로 반복적으로 개발하면서 서비스의 완성도를 높일 수 있습니다.

4. 주로 어떤 분야에 있는 분들이 수강 대상에 적합할까요?

먼저, 고색 짱검스도 흔다수는 네이티브 애플리케이션의 기획/설계/개발 신 과정을 세워야고 싶은 기획자와 개발자에게 전하고 싶습니다.

많은 개발자의 고민은 개발은 하고 있는데 애플리케이션이 어떻게 설계되고 어떤 기술들이 포함되어 있는지 잘 모른다는 겁니다.

직접적인 원인은 빨리 개발해야 하므로 다른 사람이 설계하거나 개발한 소스를 응용하는데 급급하기 때문입니다.

하지만 좀 더 근원적인 원인은 짧은 시간에 범위가 넓은 영역을 선택과 집중 없이 한꺼번에 습득하려고 하기 때문입니다.

그래서 부트캠프에서는 서비스 개발의 전 과정에 대해 거시적으로 이해하고 중요 과정은 미시적으로 자세히 배우고 실습할 수 있도록 할 것입니다.

둘째, 변화에 민첩하고 안정적이며 비용도 절감할 수 있는 서비스를 만들고 싶은 스타트업 대표나 개발자에게도 매우 필요한 강의라고 생각합니다.

이들의 문제는 성공 확률이 높은 서비스를 어떻게 만들어야 할 지 모르는 것입니다.

그 이유는 고객 중심이 아닌 공급자 마인드로 생각하여 고객의 문제를 제대로 정의 안 하고 솔루션부터 만들려고 하기 때문입니다.

본 부트캠프에서 고객 중심으로 문제를 정의하고 솔루션을 구상하여 검증하는 가이드와 예제를 얻으실 수 있습니다.

마지막으로, 클라우드 네이티브 애플리케이션 설계 및 개발 역량이 필요한 아직 희망자 또는 직무 전환 희망자입니다.

이들의 가장 큰 문제는 어떻게 시작하고 무엇을 공부해야 할 지 모르는 것입니다.

그 이유는 공부할게 많은 것도 있겠지만 기준의 교재나 강의가 너무 기술적 용어들로만 설명되어 있기 때문입니다.

많은 기업에서 다양한 역할과 수준의 사람들을 코칭한 제 경험을 바탕으로 수강생의 눈높이에서 이해하기 쉬운 언어로 전달하도록 하겠습니다

5. 다른 강의와 어떤 차별 점이 있나요?

첫째, 풍부한 경력과 블로그 방문 수로 검증된 강사의 전문성입니다.

저는 최근 5년간 15회 이상의 서비스 기획/설계/개발을 리딩하였습니다.

또한 누적 방문 수가 61만 회가 넘는 기술 블로그 '온달의 해피클라우드'를 운영하고 있습니다.

최근에는 국내에서 최초로 마이크로서비스 패턴 개발서를 출간하기도 했습니다.

둘째, 서비스의 기획/설계/개발의 전 과정을 **End-to-End**로 배울 수 있는 국내 유일의 교육이라는 것입니다.

실제 서비스를 수강생이 직접 완성해 가면서 클라우드 네이티브 애플리케이션의 전 과정을 현실감 있게 체험함으로써 교육의 효과를 극대화할 수 있을 것입니다.

셋째, 기업 대상 코칭 경험을 바탕으로 실질적 도움이 되는 가이드와 풍부한 예제를 얻으실 수 있다는 겁니다.

교육 중에 실제 예제를 통해 보다 명확히 이해할 수 있을 뿐 아니라 교육 후에도 쉬운 가이드를 참고하여 여러분의 업무에 활용하실 수 있을 겁니다.

6. 마지막으로 수강생들에게 하고 싶으신 말이 있다면?

일본의 쇠고 무사인 뉴니클로의 야나이 다나시 외상은 자신의 성공 비결이 '빨리 실패하고, 빨리 깨닫고, 빨리 수습하는 것'이라고 밝았습니다.

고객 니즈에 맞는 서비스를 기획하거나 제대로 설계하는 방법을 배우고 싶다면 너무 고민하지 말고 저와 함께 시작하십시오.

본 부트캠프에서 어떤 분은 완전히 실패하거나 부분적으로 실패할 수 있습니다.

하지만 그 실패를 통해 배우고 성장할 수 있는 새롭게 일하는 방법을 배우실 수 있을 겁니다.

개발을 모른다고 겁내실 필요도 없습니다. 원하신다면 별도로 시간을 내서 도와드리도록 하겠습니다.

여러분과 함께 할 달디 달고 단 시간을 기대하고 있겠습니다.

가사합니다