

1200H

AWS기반 스마트오더(Rabbit MQ_양방향 메세징시스템) 자바풀스택 웹개발자 양성과정

취업분야 :

풀스택 개발자, 백엔드 개발자,
자바 개발자, 웹 개발자

사용프로그램 :

HTML5, CSS3, JavaScript,
Figma, React.js, Java, RabbitMQ,
SQL, Spring Boot, AWS, Azure,
DBMS, Git, GitHub

- 양방향 메세징 서비스 및 디지털 기술을 강화하기 위해 HTML5, CSS3, JavaScript, Java를 통해 프론트엔드와 백엔드를 융합하고 새로운 품의 레이아웃과 스마트오더 시스템등의 서비스를 창출하여 웹개발자로서의 취업을 목적으로 한다.
- UI 요구사항을 확인하여 설계한 UI 설계를 기반으로 화면을 구현하는 능력을 함양한다.
- 요구사항 분석부터 API 설계와 문서화, 자동화, 테스트, API의 확장을 통해 사용자와 개발자가 사용할 수 있는 웹 API의 활용능력을 함양한다.
- 컴퓨터 프로그래밍 언어로 각 업무에 맞는 소프트웨어의 기능에 관한 설계, 구현 및 테스트를 수행하고, 서비스를 개선하는 업무능력을 함양한다.

훈련순서	교과목명	세부내용	훈련개요
STEP 1	풀스택 개발 기초 프로그래밍	[HTML5, CSS3 기초] HTML 문서의 기본 구조 / 태그 이해 / 시멘틱 태그로 문서 구조화 / 드롭다운리스트, 로그인폼, 회원가입폼 만들기 / CSS 위치, 작성순서, 클래스 이름 규칙 / 클래스와 가상클래스의 조합, 속성과 속성 값 / 모바일웹/반응형웹을 위한 뷰포트메타태그 [JavaScript] 자바스크립트 문서의 기본 구조와 이해 / 변수, 연산자 등의 기초문법 파악 / 제어문인 조건문과 반복문 이해 / 함수, 객체, 생성자함수, 프로토타입 파악하기 / BOM과 DOM의 이해 [React] React-환경설정 / todo/list 경로 처리 / React-Router / 리액트와 API 서버 통신 / 파일 업로드를 위한 설정 / 상품 관련 수정/삭제 페이지와 컴포넌트 처리	[HTML5, CSS3] [JavaScript] [React]
STEP 2	풀스택 개발 구현 프로그래밍	[JAVA 프로그래밍] 자바 프로그램 개발 과정 / 변수와 타입 / 연산자와 연산식 / 조건문과 반복문 / 참조 타입 / 클래스 / 상속, 인터페이스 / 중첩 클래스와 중첩 인터페이스 / 기본 API 클래스 [Database & SQL] 오라클 DBMS 실습환경 구축 / 사용자 계정 및 테이블 스페이스 생성 / 데이터 모델링의 이해 / SQL 기본 및 활용 / SQL 최적화, 옵티마이저와 실행계획 [Spring Boot & Gradle] FrameWork 환경설정 / DI,AOP / Spring Web / Thymeleaf 엔진 / Junit / Spring IO	[JAVA] [Database & SQL] [Spring Boot & Gradle]
STEP 3	양방향 메세징 서비스 및 API연동	[Web socket] NetWork / 채팅서버 socket 구현 / 클라이언트 socket 구현 / 애노테이션 활용(@ServerEndpoint / @OnOpen / @OnMessage / @OnClose / @OnError) / 챗봇 시나리오 설계 [RabbitMQ] RabbitMQ 계정설정 / AMQP 메시지 구조 / 서버 푸시로 전환 / 웹소켓 연결 / AMQP 테스트 / 스마트 메시지 라우팅 / 애플리케이션 테스트와 추적 [API연동] API 요청 준비 / JSON, YAML / 스웨거 에디터로 OpenAPI 정의서 작성 / API 응답 기술하기 / 자원 생성 / 인증과 인가 / API 문서 준비와 호스팅 / 웹 애플리케이션 설계 / Open-API를 사용한 API 설계 / API 설계 우선 방식에 변경 워크플로 구축 / 웹 애플리케이션 통합	[Web socket] [RabbitMQ] [API연동]
STEP 4	양방향 메세징 서비스를 위한 Cloud CI/CD	AWS, Azure, Google Cloud의 주요 서비스 개요 / 서버리스 컴퓨팅 / PaaS 서비스 사용 / CI/CD 기초 / Kubernetes 소개 / 컨테이너 오케스트레이션 서비스 활용	AWS, Azure 개발 방법론
STEP 5	프로토타입제작	피그마의 특징 및 환경설정하기 / 툴 익히기 / 스타일 설정 / 컴포넌트를 이용한 오브젝트 생성 / 오토레이아웃과 Constraints를 이용한 자동화 / 프로토타입과 인터랙션을 이용한 애니메이션 처리 / 피그마를 이용한 반응형 웹 사이트 디자인	피그마를 이용한 반응형웹사이트 디자인 구현
STEP 6	Smart Order-화면구현 (Frontend)Project	[사용자용 주문시스템] 점포메인화면 → 지도 API 연동-점포목록 → 메뉴안내 → 주문하기 → 결제화면 → 카카오킬링	사용자용 주문 시스템 구현
STEP 7	Cloud 보안 및 테스트	Spring Data JPA / Spring Security / Spring Security Oauth / 웹 보안 개요 및 HTTPS 및 CORS 이해 / 사용자 인증 및 인가, 세션 및 쿠키 관리 / 간단한 테스트 코드 작성 / 테스트 자동화 / 테스트 확장 기술	Web Security 구축 및 테스트
STEP 8	Smart Order-구축 (Backend)Project	[점주용 주문관리시스템] 점포소개의 메뉴 수정, 등록, 삭제 → 상품목록 → 주문처리 → 주문 상세내역 → 대시보드 → 실시간주문내역 → 공공 API를 활용한 서비스	관리자용 주문 관리 시스템 구현