|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 깆 | | | | |
| 입 사 지 원 서 | | | | |
|  | | | | |
|  | | | | |
|  | **성명** | **주현석** | **국적** | **대한민국** |
| **성별** | **남** | **생년월일**  **(연령)** | **1999-08-25(만 27세)** |
| **휴 대 폰** | **010-9120-5456** | **E-mail** | **wngustjr1306@naver.com** |
| **주 소** | **서울시 강동구 상암로 251 914동 504호** | | |

**[학력사항]**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **기간** | **학 교 명** | **학 과** | **졸업구분** | **소재지** | **학 점** |
| 2019.03.~ 2025.02. | 국립한국교통대학교 | 소프트웨어학과 | 졸업 | 충주 | 3.93/4.5 |
| 2015.03.~ 2018.02. | 보인고등학교 | 자연 | 졸업 | 서울 | - |

**[병역사항]**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **병역구분** | **복무기간** | **군별** | **계급** | **면제사유** |
| ■병역필 □미필 □면제 | 2021.07.08 ~ 2023.04.07 | 육군 | 이병 | - |

**[자격 및 면허 취득 사항]**

|  |  |  |  |
| --- | --- | --- | --- |
| **자 격 증 명** | **등급(점수)** | **취 득 년 월** | **발 급 기 관** |
| 정보처리기사 | - | 2024.06.12 | 한국산업인력공단 |
| TOEIC Speaking | 150(IH) | 2025.01.11 | ETS |

**[훈련이수사항]**

|  |  |  |
| --- | --- | --- |
| **교 육 기 간** | **훈 련 과 정** | **교 육 기 관** |
| 2025.01.13 ~ 2025.07.29 | (KDT)AWS와 Docker & Kubernetes를 활용한 Java Full-Stack 개발자 양성과정(7회차)(130일, 1040시간) | 쌍용교육센터 |

**[직무 능력사항]**

|  |  |  |  |
| --- | --- | --- | --- |
| **보 유 기 술** | | **개 발 환 경** | |
| **Back-end** | Java, JSP/Servlet, Mybatis,  Spring Framework, Spring Boot, Spring RESTful, Spring JPA | **OS** | Windows 10/11, Linux(Ubuntu) |
| **WAS** | NginX, Apache Tomcat |
| **Front-end** | HTML5, CSS,  JavaScript, jQuery, Vue.js Bootstrap, | **설계도구** | Figma, ERDCloud, draw.io, PlantText |
| **형상관리** | GitHub |
| **Database** | Oracle(ANSI-SQL, PL-SQL) | **기타** | AWS, RDB, |

**[포트폴리오 링크]**

|  |
| --- |
| **Github : https://github.com/githyunsuk**  **포트폴리오(Notion) :**  **기술 Blog :** |

|  |
| --- |
|  |
| 이 수 교 육 내 역 서 |
|  |

|  |  |
| --- | --- |
| **교육과정** | **(KDT)AWS와 Docker & Kubernetes를 활용한 Java Full-Stack 개발자 양성과정** |
| **교육기간** | **2025.01.13 ~ 2025.07.29 (130일/1040시간)** |

|  |  |
| --- | --- |
| **전공교과 이수내역** |  |
| **교육과목** | **주 요 교 육 내 용** |
| **Java** | 개발환경 설정 (JDK, IDE 설치 및 설정), 기본문법, 연산자, 제어문, 객체지향 프로그래밍 (Class, 상속, 다형성, 캡슐화), String, StringBuffer/StringBuilder, Array, 배열 조작, JCF(Collection, List, Set, Map), Exception Handling, 커스텀 예외처리, IOStream, NIO 파일처리, Thread, 동기화, 스레드 풀, Network 프로그래밍, Socket 통신 |
| **Oracle** | DML(select, insert, update, delete), DDL(create, alter, drop, truncate), DCL(grant, revoke), TCL(commit, rollback, savepoint), Subquery(단일행, 다중행, 상관 서브쿼리), Join(Inner, Outer, Self, Cross), Union, Intersect, Minus, 내장함수(문자, 숫자, 날짜, 변환, 집계), Constraints(PK, FK, Unique, Check, Not Null), Index 설계 및 최적화, Sequence, View, PL/SQL 기초, Stored Procedure |
| **JDBC** | JDBC 드라이버 유형과 특징, Connection Pool 설정과 관리, Statement vs PreparedStatement vs CallableStatement, ResultSet 처리 방법, Transaction 관리(commit, rollback), 데이터베이스 예외처리 |
| **HTML 5 & CSS 3**  **JavaScript & JavaScript library** | HTML5 시맨틱 태그와 웹 접근성, Form Validation과 HTML5 API, CSS3 선택자와 미디어쿼리, Flexbox, Grid 레이아웃, JavaScript ES6+ 핵심 문법, DOM 조작과 이벤트 처리, AJAX와 Promise, async/await, jQuery 셀렉터와 메서드, Bootstrap 그리드 시스템과 컴포넌트, JSON 데이터 처리, Vue.js 소개 및 설치, SPA 개념  템플릿 문법, v-bind, v-model, v-if, v-for, v-on. 이벤트 처리와 양방향 바인딩, component 의 등록법, Vue Router |
| **Servlet / JSP** | 웹 애플리케이션 아키텍처, 서블릿 생명주기와 컨테이너, Request/Response 객체 처리  Session, Cookie 관리, Filter, Listener 구현, JSP 기본 객체와 스코프, EL, JSTL 활용  파일 업로드/다운로드, MVC 패턴 구현, JSON/XML 처리 |
| **MyBatis / SpringBoot / JPA** | MyBatis 구조와 동작원리, MyBatis 설정파일 작성, XML 매퍼 작성법, 동적 SQL (if, choose, foreach), Spring-MyBatis 연동 설정  Spring boot 환경 설정, 특징 및 장점, property 설정법, Spring DI(의존성 주입의 개념, Component, @Service ,@Repository, @Autowired등),  Spring AOP(관점 지향 프로그램의 개념 ,point cut 식,@Aspect, @Before, @After, @AfterReturning,등 ), Spring MVC( Spring MVC개념 및 RESTful 개념, @Controller, @RestController,@RequestMapping, 등),  Spring Interceptor 작성방법 및 사용법, Spring Security 기본 설정, 인증/인가 처리 구현, JWT 토큰 기반 인증, JPA 기본 개념과 장점, 엔티티 매핑과 어노테이션, 식별자 생성 전략, 연관관계 매핑, 영속성 컨텍스트의 이해, 엔티티 생명주기, Spring Data JPA 활용, Repository 인터페이스 활용, 쿼리 메소드 작성법 |
| **AWS/ Linux/ Docker/Kubernetes** | Cloud의 종류, EC2의 인스턴스 유형, AMI를 사용한 인스턴스 생성, 키 페어, VPC, Elastic IO, EBS의 개념 및 스토리지 관리, 네트워크 및 방화벽 설정, MFA사용, RDS의 설정 및 사용. Ubuntu리눅스 기본 개념 및 환경 이해, 리눅스의 각종 명령어, 필수 패키지 및 소프트웨어 설치, 환경 설정 및 서비스 설정, IP tables의 이해, NginX설정 및 리버스 포워딩, Apache tomcat 설정 및 도메인 설정법. CI/CD의 이해, Jenkis의 역할 과 구조, 설치 및 github web hook설정, Job, Build, Pipeline, plugin 등, Docker의 기본 개념, 이미지, 컨테이너, Docker파일의 작성법 Docker 설치, Docker 명령어, version, run, images, ps, stop, start, rm, Kubernetes의 이해, 컨테이너 오케스트레이션 구조, 핵심구성요소 이해( Pod, Node,service ,ConfigMap, Volumne 등)설치 및 환경구성 |
| **Project** | 주제선정 후 조별 프로젝트 진행 (첨삭지도)  아이템 선정 → 개발범위설정 → story board or 화면설계서 작성 → DB설계 →  Coding → debug → 완료 후 시연 |

|  |  |  |
| --- | --- | --- |
|  | | |
| 프로젝트 수행 사항 | | |
|  | | |
|  | | |
| **작업기간** | 2025/04/28 ~ 2025/06/10 (6주) |
| **인력구성** | 7명 |
| **프로젝트 명** | 영화 예매 사이트 YEONFLIX |
| **개발환경**  **및**  **사용언어** | ◉OS : Windows 11  ◉Database : Oracle  ◉ Service System : Apache Tomcat 9.0  ◉언어/Tool : Java, JSP, HTML/CSS, JavaScript, jQuery, Eclipse, Golden7 |
| **프로젝트 소개** | CGV 영화 예매 사이트를 벤치마킹하여 제작한 영화 예매 웹사이트입니다. 관리자는 영화 및 상영 스케줄을 등록할 수 있으며, 사용자는 현재 상영 중인 영화 정보를 확인하고 원하는 상영 시간의 영화를 예매할 수 있습니다. 또한, 결제 시스템과 좌석 선점 기능을 구현하여 실제 예매 흐름을 반영하였습니다. 기존 CGV 예매 시스템을 단순히 모방하는 것이 아니라 기존 시스템의 연속 좌석 제한을 개선하여 여러 명이 예매하더라도 각자 원하는 좌석을 자유롭게 선택할 수 있도록 구현하여 차별성을 두었습니다. |
| **주요업무**  **및**  **상세역할** | -팀장으로서 프로젝트를 총괄하며 팀원 간 역할을 조율하고 일정 관리를 수행함  -프로젝트 초기 개발환경 구축 및 설정 작업 수행  -영화 예매 사이트의 핵심 기능인 예매 로직, 좌석 선점 기능, 결제 기능 등을 직접 구현  -예매, 상영 스케줄 조회, 예매 완료 등 사용자용 UI 및 관련 페이지 제작  -예매 관리, 상영 스케줄 관리 등 관리자용 UI 및 관련 페이지 설계 및 구현 |
| **느낀점** | ERD 설계가 완벽하지 않아 개발 도중 수정과 추가가 계속 발생하면서 이를 통해 설계 단계에서 더욱 꼼꼼하게 ERD를 구성해야 함을 느꼈습니다. 데이터베이스에 관한 공부가 부족하다는 점도 깨달아 단순 쿼리뿐만 아니라 복잡한 SQL까지 능숙하게 다룰 수 있도록 학습을 지속해야겠다고 생각했습니다. 또한 팀원들의 데드라인이 제대로 지켜지지 않아 프로젝트 진행이 지연되고 통합 작업이 늦어져 발표 직전까지 오류를 수정하느라 많은 시간을 소모했으며, 다음 프로젝트에서는 팀원들의 일정을 더 철저히 관리해야겠다고 다짐했습니다. |
| **결과물 링크** | 깃허브 : <https://github.com/githyunsuk/movie_prj> |

|  |  |  |
| --- | --- | --- |
|  | | |
| 프로젝트 수행 사항 | | |
|  | | |
|  | | |
| **작업기간** | 2025/3/24 ~ 2025/4/15 (4주) |
| **인력구성** | 7명 |
| **프로젝트 명** | JAVA 키오스크 프로젝트 |
| **개발환경**  **및**  **사용언어** | ◉OS : Window11  ◉Database : Oracle  ◉언어/Tool : Java, JDBC, Golden7 |
| **프로젝트 소개** | Java Swing을 활용해 제작한 키오스크 자바 애플리케이션입니다. KFC 키오스크를 벤치마킹하여 직관적인 UI를 구현했으며, 사용자가 메뉴를 선택한 뒤 사이즈, 추가 옵션 등을 자유롭게 커스터마이징하고 장바구니에 담아 결제할 수 있도록 구성했습니다. Oracle 데이터베이스와 연동하여 메뉴 정보, 주문 내역 등을 효율적으로 관리할 수 있습니다. |
| **주요업무**  **및**  **상세역할** | -팀장으로서 프로젝트를 총괄하며 팀원 간 역할을 조율하고 일정 관리를 수행함  -프로젝트 초기 개발환경 구축 및 설정 작업 수행  - erd 설계 및 데이터베이스 구축  - 키오스크의 핵심 기능인 메뉴 주문, 옵션 선택 및 변경 기능을 직접 구현  - 메인 페이지, 메뉴판, 장바구니, 결제, 영수증 등 주요 사용자 UI를 설계하고 기능을 구현 |
| **느낀점** | 인원이 많은 팀 프로젝트는 처음이라 업무 분담에 미숙한 점이 많았으며 앞으로 남은 두 개의 프로젝트에서도 팀장을 맡게 되어 이번 경험을 바탕으로 팀원들의 역량을 잘 파악하고 보다 효율적으로 업무를 분배해야겠다고 생각했습니다. 또한 변수명이나 코드 스타일이 팀원마다 달라 통합 과정에서 어려움이 있었고 이를 통해 개발 전 설계 단계에서 코딩 컨벤션을 미리 정해두는 것이 중요하다는 점을 깨달았습니다. 비록 Swing이 실무에서 많이 사용되지는 않지만, 이번 프로젝트를 통해 Java에 대한 이해를 심화할 수 있었으며 객체지향 개념도 명확하게 정립할 수 있었다고 느꼈습니다. |
| **결과물 링크** | 깃허브 : https://github.com/githyunsuk/group\_prj |

|  |  |  |
| --- | --- | --- |
|  | | |
| 프로젝트 수행 사항 | | |
|  | | |
|  | | |
| **작업기간** | 2023/9/22 ~ 2023/12/8 (10주) |
| **인력구성** | 4명 |
| **프로젝트 명** | AI 기반 커피 추천 앱 ‘커피 어때’ |
| **개발환경**  **및**  **사용언어** | ◉OS : Android 14 (API 34)  ◉Database : MySQL  ◉ Service System : Amazon EC2  ◉언어/Tool : Java, Android Studio, Firebase, Python |
| **프로젝트 소개** | AI 기반으로 커피를 추천해주는 안드로이드 앱입니다. AI 기술을 활용해 사용자가 특정 브랜드에서 즐겨찾기한 커피 메뉴의 성분 데이터를 분석하여 이를 통해 메뉴 간의 유사도를 정량적으로 계산하고, 사용자가 자주 선택하는 메뉴와 비슷한 맛과 특징을 가진 다른 브랜드의 커피를 추천합니다. 기존 커피 메뉴 앱들이 단순히 인기 순위나 최신 트렌드를 기반으로 추천하는 것과 달리 본 앱은 개별 사용자의 취향과 선호를 반영한 맞춤형 추천 서비스를 제공하여 차별성을 두었습니다. |
| **주요업무**  **및**  **상세역할** | - 팀장으로서 프로젝트 관리(PM) 역할을 수행하며 팀원 간 소통과 일정을 조율  - 프로젝트 초기 개발환경 구축 및 설정 작업 수행  - 로그인/회원가입, 홈 화면, 메뉴 화면, 즐겨찾기 화면, 마이페이지 등 주요 사용자 인터페이스를 설계하고 구현  - 팀원이 Python으로 설계한 AI 모델을 JAVA로 이식  - 데이터베이스 ERD 설계 작업 |
| **느낀점** | 처음으로 진행한 팀 프로젝트였던 만큼 부족한 점이 많았으며, 팀장 역할도 처음이라 프로젝트 관리 측면에서 일정 조율과 팀원 간 협업에 미흡한 부분이 있었다고 느꼈습니다. 앞으로는 PM 역할에 대해 더 깊이 이해하고 프로젝트를 주도적으로 이끌 수 있는 역량을 갖춰야겠다고 생각했습니다. 사용자 기능 대부분을 구현하며 많은 양의 개발을 담당하긴 했지만, AI 모델 설계나 데이터베이스 구축·연결 등 핵심 기능에 깊이 관여하지 못한 점이 아쉬웠으며, 단순 CRUD 기능을 넘어서 프로젝트의 중심 기능까지 책임질 수 있도록 지속적으로 학습하고 실력을 쌓아가야겠다고 생각했습니다. |
| **결과물 링크** | 깃허브 : https://github.com/githyunsuk/capstone\_team8 |

|  |
| --- |
|  |
| 자 기 소 개 서 |
|  |
|  |

|  |  |
| --- | --- |
| **성장과정** | **[우연한 선택에서 진짜 꿈으로]**  코딩과의 첫 만남은 대학교 전공 선택과 함께 시작되었습니다. 처음에는 단순히 전망이 좋다는 이유로 성적에 맞춰 선택한 학과였지만 막상 코딩을 접하고 나니 저에게 특별한 의미로 다가왔습니다. 어릴 적부터 레고나 블록 놀이를 통해 작은 조각들을 하나씩 쌓아가며 완성품을 만드는 과정에서 큰 즐거움을 느꼈던 저에게 코딩은 마치 그때의 경험을 되살려주는 듯했습니다. 작은 코드 조각들이 모여 하나의 완성된 프로그램으로 탄생하는 과정은 블록을 쌓아 성을 만들던 그 설렘과 성취감을 그대로 재현해 주었습니다.  특히 웹 개발을 접하면서 각각의 웹 기술들이 유기적으로 결합한 코드를 통해 만들어진 결과물이 브라우저에서 즉시 시각적으로 확인되고 다른 사람들과 손쉽게 공유될 수 있다는 점에서 웹 개발의 매력에 빠지게 되었습니다.  이를 통해 코딩이 단순한 학습 과목이 아닌 제가 진정으로 열정을 가질 수 있는 분야임을 깨달았고, 자연스럽게 웹 개발자라는 꿈을 품게 되었습니다. 우연한 선택으로 시작된 전공이 결국 제 인생의 방향을 결정하는 소중한 발견의 계기가 되었습니다. 입사 후에도 이러한 열정과 몰입을 바탕으로 사용자에게 가치 있는 웹 서비스를 개발하는 백엔드 개발자가 되겠습니다. |
| **업무 관련 역량** | **[전공 과정을 통한 배움]**  1) 대학교 학사 과정을 이수하면서 컴퓨터구조, 자료구조와 알고리즘, 데이터베이스, 네트워크 등의 기초 과목과 기초통계학, 이산수학, 선형대수 같은 수학 과목을 수강하였습니다.  2) 프로그래밍 언어로 Python, java, c언어, c++ , c#등의 언어 과목을 수강하였으며 Python과 java를 심화 학습하여 주력 언어로 두고 있습니다.  3) 그 외에도 인공지능, 기계학습, 서버운영 및 관리 등의 심화 과목을 수강하였습니다.  **[국비 교육을 통한 배움]**  1) 7개월간의 국비 교육을 통해 Java Full-Stack 개발자 양성과정을 수강하였습니다.  2) 개발 역량  **- 개발언어**: JAVA 8, Servlet/JSP(JSTL), HTML5, CSS3, JavaScript, JQuery, SQL  **- 데이터베이스**: Oracle  **- 사용기술**: JDBC, ORM(MyBatis), Ajax & JSON, Bootstrap, 다양한 API  **- 개발 툴**: Eclipse, STS, Golden7등  **- 운영체제 및 개발 서버**: Windows 11, Tomcat 8.0.5, Oracle Database 12c, SVN |
| **성격 장단〮점** | **[장점 : 소통으로 팀워크를 완성하다]**  저는 소통에 자신이 있습니다. 과거 1년 반 동안 장애 아동 봉사활동을 하며 처음에는 아이들과 가까워지는 데 어려움을 겪었지만 먼저 다가가고 인내심 있게 대화를 시도한 결과, 점차 아이들이 마음을 열었고 아이들과 자연스러운 대화가 가능해졌습니다.  이러한 경험은 팀 프로젝트에서도 큰 도움이 되었습니다. 팀장으로 활동하던 프로젝트에서는 말이 없고 의견을 잘 드러내지 않던 팀원들이 있었지만, 저는 먼저 다가가 대화를 유도하고 부담 없이 자기 생각을 말할 수 있는 분위기를 만들기 위해 노력했습니다. 덕분에 팀원 모두가 적극적으로 참여하며 프로젝트를 성공적으로 마무리할 수 있었습니다.  입사 후에도 이러한 소통 능력을 바탕으로 팀 전체의 시너지를 높여 협력하겠습니다.  **[단점: 걱정 많은 성격이 철저한 계획으로 이어지다]**  평소 생각과 걱정이 많은 편입니다. 사소한 결정에도 다양한 경우의 수를 떠올리며 고민하다 보니 때로는 실행이 늦어지는 경우도 있었습니다. 하지만 이러한 성향을 단점으로만 두지 않고 사전에 철저히 계획하고 위험을 줄이는 방향으로 활용하고 있습니다. 이러한 접근 방식 덕분에 실제 업무 진행 과정에서 예상치 못한 변수들로 인한 시행착오를 줄이고 더 안정적이고 효율적인 결과를 얻을 수 있었습니다.  입사 후에도 신중함을 바탕으로 철저한 계획을 세우고 안정적이면서도 효율적으로 문제를 해결하겠습니다. |
| **입사 후 포부** | **[끊임없는 자기 계발을 통한 성장]**  입사 후에도 자기 계발을 멈추지 않겠습니다.  첫째, 빠른 학습 능력을 바탕으로 회사의 업무 환경과 프로세스에 신속하게 적응하여, 즉시 팀의 일원으로서 실질적인 성과를 내겠습니다. 새로운 환경에도 능동적으로 대응하며, 팀에 도움이 되는 구성원이 되고자 합니다.  둘째, 부족한 부분은 퇴근 후나 주말 시간을 활용해 스터디와 자기 계발을 통해 보완하겠습니다. 빠르게 변화하는 IT 기술을 놓치지 않기 위해 꾸준히 학습하고, 알고리즘 공부도 지속하며 이를 실제 업무에 적극 반영해 나가겠습니다.  셋째, 현재 보유한 기술 스택에 안주하지 않고 지속적으로 갈고닦아 전문성을 높이겠습니다. 기존 기술에 대한 깊이 있는 이해를 바탕으로 더욱 견고하고 효율적인 코드를 작성하며, 기술적 역량을 강화해 나가겠습니다. |