



# 이동훈

## 준비된 BACK-END 개발자

### 자기 소개

안녕하세요. 저는 기억보다 기록을 믿고, 새로운 것에 도전함에 있어 흥미로움을 느끼는 개발자 이동훈입니다.

### 학력 및 교육

은행고등학교

졸업 | 2008년 3월 ~ 2011년 2월

경일게임 아카데미

수료 | 2021년 12월 ~ 2022년 3월

- 과정명 : 블록체인 기반 핀테크 응용 SW 개발자 양성

### 프로젝트 경험

#### HELPLESS-DID

2022년 8월 ~ 2022년 9월 (7주)

- 목표 : DID를 이용한 SSO 구현
- 역할
  - Login Server 구축
  - 메인사이트 Back-End
- Link : helpless-did.shop
- GitHub : <https://github.com/green-kong/DID>

#### Dodol 타입캡슐

2022년 5월 ~ 2022년 6월 (3주)

- 목표 : RN을 이용한 타입캡슐 모바일 앱 개발
- 역할
  - Back-End Server 구축
  - DB Scheme 및 관리
- GitHub : <https://github.com/green-kong/Dodol-server>

### 연락처

📍 경기도 성남시 분당구 불곡남로 17번길 5 102호

☎ 010-7749-8300

✉ ahdjd5@gmail.com

💻 [www.dev-kong.com](http://www.dev-kong.com)

📄 <https://kong-dev.tistory.com/>

### TECH STACK

- Language
  - Javascript
  - Typescript
- Framework
  - Express
  - Nest.js
- Database
  - mySql
  - MongoDB
  - Redis

## 프로젝트 경험

### Carrot World

2022년 3월 ~ 2022년 4월 (3주)

- 목표 : 바닐라 JS로 SPA 구현 및 socket.io를 이용한 실시간 경매 & 1:1 채팅
- 역할 :
  - Back-End Server 구축
  - DB Scheme 및 관리
- GitHub :  
[https://github.com/greenkong/carrot\\_world](https://github.com/greenkong/carrot_world)

### Team6 Board

2022년 2월 ~ 2022년 2월 (1주)

- 목표 : 로그인 기능 및 게시판 기능
- 역할 :
  - UI/UX 디자인
  - Front-End
  - DB Scheme 및 관리
- GitHub :  
[https://github.com/greenkong/team6\\_login\\_board](https://github.com/greenkong/team6_login_board)

# 자기소개서

## 10년차 요리사가 백엔드 개발자로 전향한 이유

저는 약 10년간 작은 중식당에서부터 시작해, 국내 유명호텔을 거쳐, 군 전역 이후에는 시드니의 여러 다이닝에서 요리사의 삶을 살고 있었습니다. 그런데 약 2년 전, 현재 개발자로 현업에 종사 중인 가까운 지인을 통해 개발자라는 새로운 길에 대해 알게 되었고, 관심을 두게 되었습니다. 개발자라는 직업에 대해 알아볼수록 정말 매력적인 직업이란 것을 알게 되었습니다.

그 중에서도 제가 하는 일이 저 또는 회사의 이익만을 위한 것이 아니라, 사람들의 삶을 편하게 만들어 줄 수 있다는 점이 가장 매력적이었고, 이러한 매력에 매료되어 개발자가 되기로 하였습니다.

개발자가 되기 위한 공부를 위해 부트캠프와 국비교육을 알아보던 도중, 현재 수료 중인 '블록체인 기반 핀테크 응용 SW 개발자 양성' 국비 교육과정을 접하게 되었습니다. 웹의 기초부터 프론트엔드, 백엔드 영역을 모두 경험해 볼 수 있었고, 뿐만 아니라 블록체인 기술까지 배워볼 수 있는 좋은 기회라 생각했습니다. 그리하여, 29살의 꼬트머리였던 2021년 12월부터 개발자가 되기 위한 공부를 시작했습니다.

## 저는 이런 개발자입니다.

### <꾸준히 기록하는 개발자>

저는 개발을 공부하기로 마음먹은 날부터 블로그를 시작했습니다. 매일매일은 아니더라도 새로운 것을 접하거나, 해결하기 어려웠던 문제를 해결했을 때에는 꼭 기록을 남겼습니다. 처음에는 단순히 배웠던 것을 까먹었을 때를 대비하기 위해 블로그를 시작했지만, 학습한 내용을 블로그에 작성하는 것이 개념을 정리하는 데에 큰 도움이 되었습니다. 또한, 저와 같은 수업을 듣는 동기들이 어려운 부분을 이해하는 데에 있어 제 블로그가 많은 도움이 된다고 했을 때 뿌듯함을 느꼈습니다.

### <함께 성장하는 개발자>

저의 첫 팀 프로젝트는 일주일이라는 짧은 시간 동안 진행되는 작은 토이 프로젝트였습니다. 팀장을 맡아 프로젝트를 진행하게 되었는데, 기술적인 문제보다는 팀원 간의 실력 차에 따른 어려움이 더욱 컸습니다. 총 4명의 팀원 중 유독 한 명의 실력이 많이 뒤쳐져 있었습니다. 팀장으로서 이번 프로젝트를 통해 팀원들의 역량도 높이고 멋진 결과도 내고 싶었기에, 주어진 기간 동안 오전에는 실력이 뒤쳐진 팀원과 함께 페어 프로그래밍을 진행했고, 이후의 시간에는 제 몫의 작업을 했습니다. 페어 프로그래밍을 하면서 해당 팀원의 실력이 부쩍 늘어 프로젝트 막바지에는 스스로 힘으로 한가지의 기능을 구현할 수 있었습니다. 저 또한 지식을 공유하면서 혼자 학습할 때보다 빠른 속도로 실력이 늘 수 있었습니다.

### <끊임없이 도전하는 개발자>

저는 도전에 대한 두려움보다는 흥미를 느끼는 편입니다. 수료 중인 과정에서 현재까지 총 3번의 팀 프로젝트를 진행하였는데, 3번 모두 수업 때 다루지 않았던 내용을 접목해 챌린징한 프로젝트를 진행하였고 결과 또한 만족스러웠습니다. 첫 번째 프로젝트에서는 비동기 통신에 대해 배우기 전이었지만, 구글 검색을 통해 XMLHttpRequest와 fetch에 대해 공부하고 비동기 통신을 구현했습니다. 두 번째 프로젝트에서는 React를 사용하지 않고 SPA를 구현하는 것에 도전하였고 주소창에 Hash(#)를 이용하여 React의 Router와 비슷한 기능을 구현할 수 있었습니다. 세 번째 프로젝트는 React를 이용한 앱 개발이 과제였고, "모바일 타임캡슐"을 주제로 진행했습니다. 타임캡슐의 특성상 위치기반 기술을 사용했는데, 웹보다 정확한 위치를 탐지할 수 있는 모바일 앱이 적합하다고 판단하여, 수업에서 배우지 않았던 expo와 React-Native를 이용해 모바일 앱 개발을 도전하였습니다. 그뿐만 아니라, 코드의 가독성을 높이기 위해 정적 타입 언어인 typescript를 도입하였습니다. 이처럼 저는 배운 것만을 활용하는 것이 아니라, 개발하려는 서비스에 따라 더욱 적합한 기술이 있다면 끊임없이 도전하는 자세를 갖고 있습니다.

## <주체적인 개발자>

저는 배움과 도전에 있어 주체적으로 행동합니다. 현재 수료 중인 과정에 자료구조 / 알고리즘에 대한 수업이 없어 많은 아쉬움을 느끼고 있었습니다. 그래서 같은 수업을 듣는 동기 중 자료구조 / 알고리즘에 흥미가 있는 친구들 모아 스터디를 개설했습니다. 모든 문제를 다 풀어내지는 못했지만 서로 부족한 부분을 채워주며, 일주일에 4문제 이상 푸는 것을 목표로 하였고, GitHub를 통해 서로의 코드를 공유하고 토론하는 시간을 가졌습니다. 또한 2022K-디지털 트레이닝 해커톤에 참가하였습니다. Local 2 You 라는 팀명으로 블록체인 기술을 활용하여 지역의 농수산물 수확 과정 및 유통과정을 투명화하여 소비자와 생산자 간의 신뢰도를 높이고, 더 나아가 지역 경제 활성화를 목표로 하는 서비스를 기획하였습니다. 이처럼 교육과정 내에서 주체적으로 팀을 구성하고 여러 활동에 도전했습니다.

## 기술적으로 어려웠던 부분을 해결한 경험

마지막으로 진행했던 프로젝트(HELPLESS-DID)에서 Refresh Token을 사용하는 과정에서 기술적인 어려움을 겪었습니다.

Refresh Token을 도입한 이유는 stateless한 서버의 특성상 Access Token의 탈취를 막을 수 없으며, 또한 탈취의 여부조차 서버에서는 알 수 없기 때문이었습니다. 보안강화를 위해 Access Token의 만료시간을 2시간으로 설정하는 것도 생각해 보았으나, 사용자의 편의성이 매우 감소할 것으로 생각하여 Refresh Token을 도입하기로 하였습니다.

Access Token의 만료시간을 2시간으로 짧게 설정하고, Refresh Token의 만료시간을 2주로 설정하여, Access Token이 있으면, 검증 이후에 Access Token을 통한 로그인을 진행하고, Access Token이 만료되고, Refresh Token만 남아있는 경우는 Refresh Token 검증 이후에 Access Token과 Refresh Token을 재발급하는 로직을 구성하였습니다.

Refresh Token은 DB에 저장하여 관리하였는데 mySql에서는 특정 row에 만료시간을 설정하여 삭제하는 것이 불가능하여, Redis에 저장하여 TTL을 2주로 설정함으로 Refresh Token의 만료시간이 지나면 자동으로 삭제되게 처리 하였습니다.

또한 XSS 공격을 통한 Refresh Token의 탈취를 방어하기 위해 Refresh Token을 쿠키로 설정하고, httpOnly 설정을 해주었습니다.

경일게임아카데미에서 수료한 과정에서는 Refresh Token을 다루지 않았기에 막연한 기분이었으나, 검색을 통해 개념정리부터 차근차근 작업을 진행했습니다.

이 과정을 통해 처음 접한 개념일지라도 코드로 구현할 수 있다는 자신감을 얻게 되었고, 보안과 사용자 편의성에 있어서 깊게 고민할 수 있는 계기가 되었습니다.