



허진한

Software Engineer

 [LeoHeo](#)

 010-9979-5385

 [LeoHeo.github.io](#)

 hjh5488@gmail.com

“어제보다 오늘이 더 나은 개발자가 되자”

간략한 소개

Java, Spring Framework, Node.js, Express.js, Vue.js, PostgreSQL, Firebase, AWS를 사용해서 개발하고 프로덕션 운영을 해봤습니다.

테스트코드의 중요성을 느껴서 꾸준히 프로젝트나 토이프로젝트에 테스트코드를 짜려고 노력 중이며, 최근에는 개인 시간에 주로 그동안 몰랐던 거에 대해서 공부하면서 기록 하는 걸 좋아합니다. 이런 기록들을 통해 6개월, 6년이라는 경력 숫자의 틀보다 이런 시간의 쌓임이 헛되지 않은 사람 그래서 앞으로의 시간도 그렇게 보낼 것이라 예상되는 사람이 되고 싶은 개발자 허진한 이라고 합니다.

보유기술 (관심분야)

- Front-end
 - javascript
 - Vue.js
- Back-end
 - Node.js
 - Express.js
 - Java
 - Spring Framework
 - AWS
- Database
 - PostgreSQL
 - MySQL
 - MongoDB
 - Firebase
- Environment
 - AWS
 - ubuntu
- Tooling/DevOps
 - Docker
 - Git / Git-flow / GitHub
 - Tmux
 - Slack
 - Asana
 - Travis CI

프로젝트

2018.01 ~ 2018.05

아이디어스터디

지문인식기 API 개발

- 한 일
 - Express.js + mongodb를 활용한 API 개발
 - Slack을 활용한 Travis CI, Sentry Error Tracking 구축
- 프로젝트 하면서 가장 중점을 두었던 사항
 - 모든 API에 대해서 Test Code를 작성
 - AWS 무중단 배포
 - git-flow(release, develop, feature) 기반의 세분화 개발
- 개발환경
 - Node.js
 - Express.js
 - Mongodb
 - AWS(ElasticBeanstalk, S3 + CloudFront, Route 53)
 - git, git-flow

프로젝트

2017.01 ~ 2017.12

아이디어스터디

독서실 관리자 프로그램 개발

- 한 일
 - Front-end(Vue.js + Vuex + Vue-Router)를 활용한 개발
 - Back-end(Express.js + Firebase + PostgreSQL)를 활용한 개발
 - Slack을 활용한 Travis CI, Sentry Error Tracking 구축
- 프로젝트 하면서 가장 중점을 두었던 사항
 - AWS Lambda를 활용한 각종 cron 작업
 - AWS 무중단 배포
 - git-flow(release, develop, feature) 기반의 세분화 개발
- 개발환경
 - Node.js
 - Express.js
 - PostgreSQL
 - Vue.js
 - Firebase
 - AWS(ElasticBeanstalk, S3 + CloudFront, Route 53, RDS, IoT, Lambda)
 - git, git-flow

프로젝트

2015.08 ~ 2016.01

KPMG(삼정 회계법인)

물류정보시스템(Omega) 개발

- 한 일
 - PDF Print Module 개발 및 ExtJS FrontEnd 구현
 - iText library사용 인사기록카드 PDF 출력 개발
 - 프로젝트하면서 가장 중점을 두었던 사항
 - ExtJS Grid 커스터마이징
 - 개발환경
 - Java
 - Spring
 - MS-SQL
 - ExtJS
 - svn
-

프로젝트

2014.02 ~ 2015.07

KPMG(삼정 회계법인)

FlexSolutions v3.0 – 인사급여 ERP

- 한 일
 - 인사, 휴가, 퇴직금 Module 개발
 - iText library사용 인사기록카드 PDF 출력 개발
 - 퇴직소득 계산 및 영수증 출력
- 프로젝트 하면서 가장 중점을 두었던 사항
 - v2.0에 없던 퇴직소득 계산 및 영수증 출력
- 개발환경
 - Java
 - Spring
 - MS-SQL
 - ExtJS
 - svn

토이 프로젝트

2016.06 ~ 2016.07

패스트캠퍼스

HOOMI API 개발- 이력서 관리

- 한 일
 - python-social-auth를 활용한 OAuth 구현
 - SSL letsencrypt 적용
 - 사진 업로드시 AWS S3 업로드
 - Slack을 활용한 Travis CI, Sentry Error Tracking 구축
- 프로젝트 하면서 가장 중점을 두었던 사항
 - 모든 API에 대해서 Test Code를 작성
 - git-flow(release, develop, feature) 기반의 세분화 개발
- 개발환경
 - Python
 - Django
 - PostgreSQL
 - AWS(EC2, Route 53, RDS)
 - git, git-flow

토이 프로젝트

2018.04 ~ 진행 중

Collect

- 한 일
 - e-commerce toy project (WIP)
 - Spring Data JPA 공부 중
 - Spring Security OAuth 구현해보기
 - Spring AWS SES 사용해보기
 - 프로젝트 하면서 가장 중점을 두었던 사항
 - 진행하면서 삼질했던 거 기록
 - git-flow(release, develop, feature) 기반의 세분화 개발
 - 개발환경
 - Java
 - Spring boot 2.0(Spring Data JPA, Spring Security)
 - MySQL
 - AWS(SES, ElasticBeanstalk)
 - git, git-flow
-

- 문제
 - AWS Lambda function들이 version control 안됨
 - 여러 functions들이 공통쓰는 npm 모듈을 재사용 못함
- 원인
 - Lambda 특성상 function마다 package.json이 필요해서 공통모듈을 재사용하지 못함
 - 소스를 업로드 하면 기존 걸 override하는 Lambda default 옵션
- 연구
 - apex라는 Lambda mangement하는 오픈소스 공부 & 삽질
 - 공부한 내용을 가지고 사람들 앞에서 발표
- 해결
 - apex로 deploy를 하면 Lambda version control 가능
 - npm 공통모듈이 있을경우 apex hook를 이용해서 webpack으로 번들링 후 deploy
- 평가
 - AWS Lambda version control 및 rollback 손 쉽게 가능
 - 공통된 npm 모듈 중복 제거

AWS IoT를 활용한 IoT 장비 DC 전원 제어 개선

- 문제
 - DC 전원을 제어하는 IoT 장비가 있었는데 API polling을 해서 실시간 전원제어가 안 되는데 실시간 전원제어가 필요하게 됨
 - polling 하는 기기가 많아질수록 AWS 비용이 많이 증가함
- 원인
 - polling을 하므로 전원 제거가 최대 5~8초가 걸려서 사용자들 불편호소
 - 어떤 IoT 기기는 하루에 10번 이내 전원제어를 하는데 24시간 polling을 하니깐 AWS 비용 측면에서 비효율 증가
- 연구
 - 하드웨어 하는 팀원과 회의를 하다가 mqtt라는 대안을 발견
 - AWS IoT에 대해서 qiita 글을 보고 공부 및 삽질
- 해결
 - AWS IoT + API Gateway를 만들고 IoT 장비 mqtt 사용으로 해결
 - 항상 polling을 하는 게 아니고 필요할 때만 네트워크 통신을 하므로 효율적
- 평가
 - AWS 기존 비용 대비 1/5 수준으로 감소
 - 실시간으로 IoT 장비들 전원제어가 가능해짐