이황로

Software Engineer

(+82) 10-4503-7743
hwangro123@gmail.com

소개

개발을 좋아합니다.

친절한 개발자를 목표로 개발을 즐기고 있습니다. 새로움을 즐기고 어려움을 극복하면서 개발을 하고 있습니다.

스타트업에서 웹 프론트엔드/백엔드 업무 모두 경험을 했고 수많은 새로운 기능을 개발했으며 버그를 수정해왔습니다. 그리고 서비스가 불안정할 때 MySQL, Redis 등 시스템 퍼포먼스를 개선하여 안정화 시킨 경험이 있고 제가 좋아하는 업무입니다.

평소 소통과 신뢰를 중시하며 사소한 문제라도 팀원과 소통하는 편입니다. 특히 코드리뷰를 하면서 사소한 부분이라도 궁금한 부분에 대해서 소통합니다. 코드리뷰를 통해 잠재적 버그와 좋은 코드품질 유지, 견고한 서비스를 만들기 위해 노력하고 있습니다.

개발할 때 현재 개발 상황과 가까운 미래를 생각하여 오버엔지니어링이 되지 않도록 항상 고민합니다.

최근에는 테크 리드로써 팀원들이 즐겁게 개발하면서 성장하기 위해 어떻게 팀을 운영해야 할지 고민을 많이 하고 여러 시도를 했습니다.

- 팀원 컨디션 확인 및 팀 운영 개선 사항 등을 확인하기 위한 정기적 1 on 1
- 업무에 대한 작은 부담을 주어서 책임감을 가지게 하기 위한 데일리 미팅
- 코드 안정성과 코드 읽기 능력 향상, 책임 분배를 위한 코드 리뷰
- PT 능력을 기르고 자신만의 경험을 공유하기 위한 DEV 세션 운영
- 본인의 경험과 지식을 잘 정리하기 위한 회사 블로그 운영
- 특정 개발자에 의존하지 않고 안정적으로 운영하기 위해 Bus Factor 방식으로 운영

기술

- 서버: Java, Python
- 프론트엔드 : Javascript, CSS, HTML5, SASS, Reactjs(Nextjs), jQuery, Webpack, Babel
- 프레임워크 : Spring Boot
- 데이터베이스 : MySQL, MongoDB, Redis, BigQuery
- 테스트 : JUnit 5, Spock Framework
- 인프라
    - AWS : EC2, S3, API Gateway, Lambda, RDS, CloudWatch, CloudFront, MQ 등
    - GCP : Compute Engine, Cloud Storage, Cloud Run Function, Cloud SQL, Cloud Logging, Cloud Task 등

- 기타 : Airflow, Docker, Crawler(Selenium, Playwright)

경력

Metric Studio Inc. (NNT), 서울시 - *Tech Lead*

2023년 5월 - 2024년 12월

- NNT Tech 팀 운영
    - 1 on 1 : 팀원과 정기적으로 미팅하여 팀원의 애로사항 및 팀 개선 항목 파악 및 개선
    - 코드 리뷰 문화 확립 : 코드 안정성, 에러 핸들링, 기능 책임 분배 등 개선
    - DEV 세션 운영 : PT 능력 및 자신만의 개발 경험 공유를 위해 DEV 세션 운영
    - Bus Factor 방식 : 특정 개발자에 의존하지 않고 안정적으로 운영하기 위해 Bus Factor 방식으로 운영
- 프로젝트 온보딩
    - 프로젝트 온보딩을 위한 고객사와 온/오프라인 미팅 참여
    - 고객사 개발환경에 맞는 개발 가이드라인 제공 및 가이드 문서 작성
    - 업무를 명확히 하고 빠르게 진행하기 위해 내부 담당자와 고객사와의 커뮤니케이션 진행
- 프로젝트 PM
    - Hyundai DDM 2.0 (w/ INNOCEAN)
        - 프로젝트 관리 : Hyundai DDM 1.0 프로젝트를 빠르게 팔로업 후 PM 으로써 프로젝트 관리 및 개발 참여
        - 일정 관리 : 프로젝트 팔로업 후 개발 항목에 대해서 일정을 체계적으로 관리
        - 개발 참여 : 데이터 파이프라인 개발, 개발 가이드 문서 및 인수인계 문서 작성 등
    - Innocean Ads Hub 1차 프로젝트 PM
        - 프로젝트 관리 : WBS 작성, 프로젝트 업무 관리, 내/외부 업체와의 커뮤니케이션, 프로젝트 산출물 작성 및 정리 등
        - 일정 관리 : WBS 기반으로 일정 관리. 정기적 미팅을 통한 업무 파악
        - 개발 서포트 : 인프라 설계 및 개발 방법론 수립, 개발 가이드 제공
    - LGU+ Marketing Data Warehouse
        - 프로젝트 관리 : WBS 작성, 프로젝트 업무 관리, 내/외부 업체와의 커뮤니케이션, 마케터 업무 서포트 등
        - 일정 관리 : WBS 기반으로 일정 관리. 정기적 미팅을 통한 업무 파악
        - 개발 서포트 : 인프라 설계 및 개발 방법론 수립
    - JYP Three Six 커머스 데이터 통합 및 매출 대시보드 개발
    - 버거킹 마케팅 성과 데이터 파이프라인 개발
    - 챌린저스 마케팅 성과 데이터 파이프라인 개발
- NNT Consulting 서포트
    - Content SEO 를 위한 서브폴더/서브 도메인 워드프레스 서버 셋업 가이드 작성

    - Content SEO 를 위한 서브폴더/서브 도메인 셋업을 위한 온/오프라인 미팅 참여 및 가이드 제공
    - 마케팅 성과 데이터 리포트 자동화 파이프라인 구축
    - 사내 시스템으로 관리 중인 GCP/AWS 기반 서버리스 아키텍처 설계 및 개발
    - 사내 시스템 및 고객사 프로젝트 관리 및 비용 최적화를 위한 개발 가이드 구성
- 시스템 관리 및 기술
    - 클라우드 비용 절감을 위한 퍼포먼스 개선
    - GCP : Cloud Quotas, Cloud Billing Budget, BigQuery 개선
    - Python, Java, Spring boot, Airflow
    - 크롤링 : Selenium, Playwright

타파스 엔터테인먼트 코리아**,** 서울 - *Software Engineer*

2019년 6월 - 2023년 4월

- 주도적으로 개발 진행
- 대부분의 코드리뷰에 적극적으로 참여(잠재적 버그 발견, 코드 품질 향상)
- Spring, Java 기반 서버 개발
    - Spring, java, MySQL, MongoDB, Redis, AWS(ec2, s3, rds, mq, etc...), 등
    - 다양한 피처 개발 및 버그 수정, 퍼포먼스 개선
- TDD
    - Spock framework 을 이용한 테스트코드 작성
    - 새로운 프로젝트에 Junit 5 도입.
- 웹 프론트엔드 개발
    - jQuery, javascript, html, css, grunt
    - 메인 화면 개편
    - 대시보드 전체 개편(리드)
    - 신규 기능/기존 기능 개편시 서버/웹 프론트 동시 개발
- 트래픽 장애 해결을 위한 MySQL 쿼리 개선
    - MySQL Slow Query 모니터링 및 튜닝
    - 트랜잭션 최적화를 위해 쿼리 용도에 따라 Isolation Level 을 조정 및 트랜잭션 분리
    - 트랜잭션을 리팩토링하여 퍼포먼스 향상
    - 서비스 및 쿼리 용도에 따라 캐시 처리 및 Replica DB 활용
- Redis 최적화
    - 캐시의 용도를 파악 후 TTL 조정하여 Redis 메모리 최적화
    - 서비스 장애를 유발하는 레디스 명령어를 개선하여 서비스 안정화 도모
- 새로운 기술 도입
    - 임시저장 기능을 위한 MongoDB 도입
    - Redis pub/sub 단점을 보완하기 위해 RabbitMQ 도입
- 재무/잉크/결제 업무 팀장

(주**)**엑스트라이버**,** 서울 - *Software Engineer*

2018년 2월 - 2018년 11월

- 트립스토어 앱 관리를 위한 어드민 개발 및 유지보수

    - 어드민 리뉴얼 : AngularJS 4 에서 Freemarker로 변경
    - 홈 카테고리 관리 기능 구현
    - 태그 관리 기능 구현
    - 프로모션 관리 기능 구현
    - 여행상품에 적용되는 다양한 이벤트 관리 기능 구현(특가 등)
    - 기타 다양한 기능 구현
- 트립스토어 앱 신규기능 API 개발
    - 홈카테고리 API 구현
    - 기타 API 구현
- 스케줄러 서버 구현
    - 사용자가 여행 예약하면 어드민페이지에 알람
    - 평일 예약현황을 이메일로 전송
    - 태그 및 프로모션, 특가 적용 및 해제 기능
- 페이스북 및 크리테오 리타겟팅 모듈 개발
- 개발환경
    - Spring boot + Kotlin 기반 서버 프로그래밍
    - Spring boot batch + Quartz 조합으로 배치 및 스케쥴러 구현
    - AWS Aurora DB로 데이터 관리
    - AWS ElasticCache로 캐시관리

(주**)**디자이어랩**,** 서울 - *Software Engineer*

2017년 5월 - 2017년 11월

<브랜드 의류를 모아 볼 수 있는 메타서비스>

- YOIL 앱 관리 어드민 개발 및 유지보수
    - 크롤링한 상품 데이터를 자동으로 가공하도록 구현
    - 어드민 페이지 기능 개선
    - 커머스 관리용 어드민 개발
- YOIL 앱 신규 기능 API 개발
- 카테고리 체계도 재정의
    - neo4j를 활용하여 카테고리를 그래프 형태로 재정의
- 페이스북 및 크리테오 리타겟팅 모듈 개발
- 개발환경
    - MSA 기반으로 개발
    - Spring boot + Java 기반 서버 프로그래밍
    - AWS Aurora DB, MongoDB, DynamoDB로 데이터 관리
    - AWS Elasticache로 캐시 관리
    - EC2 + ELB + Ansible + Jenkins 기반 서버 운영 및 배포

(주**)**디에스멘토링**,** 서울 - *Software Engineer*

2016년 1월 - 2017년 3월

- LDAP 성능관리 솔루션
    - 서버개발
        - Java 8, Spring Boot
        - MongoDB

        - MariaDB
        - Redis 등
    - 프론트엔드 개발
        - Web Frontend(HTML/CSS,AngularJS,Bootstrap)
        - 대부분의 Front-End 구현
        - Bootstrap, Javascript 활용
    - Gitbucket을 이용한 형상 관리
    - 실시간 모니터링 구현
    - Websocket으로 실시간 모니터링 데이터 통신
    - D3로 모니터링 뷰 구현
    - 로그인을 비롯한 대부분의 CRUD 개발
- 정부 디렉토리 사업
    - 개발환경
        - Web Backend(Java,, 전자정부프레임워크, Jeus, Tibero)
        - Web Frontend (HTML/CSS, JQuery ,JSP)
    - SVN을 이용한 소스관리
    - LDAP을 이용한 직원 및 임시조직 관리구현
    - 직원권한에 따라 소속직원 추가, 수정, 삭제 구현
    - 임시조직 생성, 수정, 삭제 구현
    - LDAP을 기반으로 조직도 구현
    - 트리구조의 조직도 구현
    - 사용자권한에 따라 자신이 속한 조직도만 보이도록 구현
    - LDAP을 기반으로 직원권한에 따른 검색기능 구현
    - 사용자권한에 따라 기관 및 소속직원만 검색되도록 구현

학력

조선대학교, 전라남도 광주 - 학사

2009년 3월 - 2016년 2월

- 정보통신 공학과
- 삼성 소프트웨어 멤버십 (광주) (24-2기)

링크

- 개인 사이트
    - https://hwangrolee.github.io
- **LinkedIn**
    - https://www.linkedin.com/in/hwangro-lee-68082a185/
- **Medium**
    - https://medium.com/@lhr0419
- **Github**
    - https://github.com/hwangrolee

프로젝트

블록체인 기반 주거 증명 프로젝트 - *Web Frontend Engineer*

2019년 2월 - 2019년 4월

- 이더리움연구회 dApp분과에서 진행한 프로젝트.
- React.js기반 웹 프론트엔드 개발.

에어레시피 - *Backend Engineer*

2018년 11월 - 2019년 3월

- AWS Serverless 아키텍처 기반 인프라 구성, API 개발, 서버 운영

코인킷 - *Full Stack Engineer (Backend, Web Frontend)*

2017년 11월 - 2018년 10월

- 구매한 블록체인 코인을 원하는 실물 화폐단위로 확인할수 있는 블록체인 자산 관리 서비스.
- 거래소에 등록된 암호화폐의 실시간 가격을 원하는 실물 화폐단위로 환산하여 볼 수 있음
- 두 개의 거래소에 등록된 암호화폐 가격을 비교할 수 있음
- 암호화폐 자산을 등록하여 자산을 모아볼수 있고 원하는 실물화폐로 환전하여 볼 수 있음
- 암호화폐 자산에 등록한 자산의 손익/손실을 금액과 손익율/손실율로 볼 수 있음
- 암호화폐의 가격이 떨어지면 알람을 받을 수 있음
- GCP 기반 아키텍처 구성
- 서버 전체 구현
- 웹 전체 구현

문제해결

GCP BigQuery 과금 이슈 해결

- 이슈 사항
  - BigQuery 의 비용이 예상 비용보다 많이 나옴
- 이슈 원인 확인 과정
  - BigQuery 비용을 자세히 보기 위해 GCP Billing Report 를 SKU 단위로 조회하여 어느 작업에서 비용이 발생했는지 확인
  - BigQuery 쿼리 처리량이 늘어서 비용이 발생한 것으로 확인되어 BigQuery Monitoring 기능을 활용하여 문제가 되는 쿼리 확인
- 해결 사항
  - 쿼리 튜닝 및 로직 변경
    - 쿼리에서 where 절에 date, account_id 값을 조건으로 사용 중이었고 date 필드로 파티션된 테이블이기 때문에 account_id 로는 비용이 절감되지 않고 있었던 것으로 확인

        - BigQuery 는 파티션 필드 외에 다른 필드 사용시 비용 절감이 되지 않는다.
        - BigQuery 비용 개선을 위해 파티션을 date 필드로 지정해두었고 이 이슈를 해결하기 위해서는 where 절에서 account_id 를 제거하고 date 필드로만 쿼리가 실행되도록 실행하고 account_id 조건에 따라 반복 실행되는 로직을 제거하여 해결.
        - BigQuery 테이블 설계와 쿼리 조건이 적절히 일치하지 않아 발생한 비용 증가 문제를, 파티션 필드(date)만 활용하도록 쿼리 조건을 수정하고 반복 로직을 제거하여 해결함. 이를 통해 쿼리 효율성을 높이고 운영 비용을 효과적으로 절감할 수 있었음.
    - GCP Quotas 설정
        - GCP Quotas 를 활용하여 BigQuery 일 사용 할당량 설정
        - GCP Quotas 를 활용하여 Cloud Run Function 할당량 설정

콘텐츠 **SEO** 를 위한 워드프레스 서브 폴더 연결

- 이슈 사항
    - 고객사 환경에 따라 워드프레스 서버를 고객사 웹 서비스 서브폴더 URL 로 연결하기 위한 가이드 및 직접 해결
- 해결 사항
    - 도메인에 활용 중인 CDN(CloudFront, CloudFlare 등) 을 활용하여 서브 폴더 적용이 가능한지 파악 - 가이드 문서 제공
    - 웹 서버 (NginX, Apache 등) 파악 후 Reverse Proxy(URL rewrite 등) 기능을 활용할 수 있는지 파악 후 해당 웹 서버 공식 문서를 기반으로 문서 제공
    - 고객사 워드프레스 서버에 직접 원격 접속하여 이슈 확인하여 해결
    - 개발 환경에 따라 서브 폴더로 연결이 안될 수 있기 때문에 이 경우 서브 도메인으로 운영 할 수 있도록 커뮤니케이션 진행

Redis Out of Memory (OOM)

- 이슈 사항
    - WEB/API 에서 활용하용 캐시용 레디스가 거의 매일 OOM 에러가 발생
- 해결 사항
    - 원인 파악을 하고 해결하기 전까지 매일 퇴근 시간에 캐시용 레디스를 정리하는 배치 개발
        - 캐시용 레디스에 의해 서비스에 이슈가 생기면 안되기 때문에 임시 해결 사항
    - 캐시를 사용하는 WEB/API 를 확인하여 사용에 따라 캐시로직 제거 혹은 TTL 을 조정
    - 레디스 서버 메모리 증설 및 레디스 메모리 모니터링을 위해 Datadog 활용

Web/API 응답시간 지연 **(1) - MySQL** 튜닝

- 이슈 사항
    - Web/API 서버가 간헐적으로 다운되어 서비스 장애 발생
- 해결 사항
    - MySQL Slow Query 기능 활성화하여 쿼리 확인

    - Slow Query 중 빠르게 튜닝이 가능한 쿼리는 바로 쿼리 튜닝 혹은 인덱스 추가
    - Slow Query 중 실제로 느린 쿼리가 아닌 다른 이슈로 인해 느리게 동작하는 것을 확인하여 비대한 트랜잭션의 경우 트랜잭션 분리 작업 진행
    - 트랜잭션 분리 작업 중 기능에 따라 정합성이 중요하지 않은 쿼리는 Isolation Level 을 Read Uncommitted 로 변경하고 Read Uncommitted는 레플리카 디비를 보도록 데이터소스 수정하여 테이블 락이 발생하지 않도록 개선
    - MySQL 커넥션 풀 조정

Web/API 응답시간 지연 **(2) - Redis** 튜닝

- 이슈 사항
    - Web/API 서버가 간헐적으로 다운되어 서비스 장애 발생
- 해결 사항
    - 데이터 임시 저장을 위해 사용 중인 레디스에서 사용 중인 명령어 중 O(N) 시간 복잡도를 가진 명령어를 모두 제거하고 최대한 O(1) 명령어로 개선
    - 레디스 설정 튜닝

WEB/API <> Batch 의존성 해결

- 이슈 사항
    - Web/API 에서 사용 중인 서비스 로직 중 Batch 로 보내는 요청이 제대로 동작되지 않는 경우가 발생
    - Web/API <> Batch 간 통신을 Redis Pub/Sub 사용 중이어서 배치에서 제대로 처리 못한 경우 해당 요청이 유실되는 상황 발생
- 해결 사항
    - AWS MQ(RabbitMQ) 도입하여 요청이 유실되지 않도록 개선