

REST API

API

데이터, 서비스를 이용할 수 있도록 하는 도구

REST

(Representational State Transfer)

네트워크를 통해 컴퓨터끼리
통신할 수 있는 아키텍처 스타일

이러한 아키텍처 스타일은 왜 필요한가?

Soap

(Simple Object Access Protocol)

보안, 메시지 전송 등에 있어
REST 보다 많은 표준이 정해져 있다.

○ 능동적으로 새로운 기술을 학습하는 개발자

웹 프로그래밍 기초 강의를 수강한 이후 시간, 공간 등 서비스의 한계를 기술을 통해 극복할 수 있다는 점에서 개발 분야의 비전을 느껴 직무를 전환하게되었습니다.

2023년에는 삼성청년SW아카데미에서 1년 간 교육과정과 스터디를 통해 기본 CS지식과 프로그래밍 언어, 프레임워크, 데이터베이스에 대한 지식을 쌓았습니다. 주력 언어는 Java이고 Spring Boot, Spring Data JPA를 주로 사용하고 있습니다. 이 밖에도 3번의 프로젝트를 통해 기획부터 배포까지 경험하였습니다.

저는 필요한 기술을 빠르게 학습하고 적용하는 역량을 가지고 있습니다.

3개의 프로젝트 중 하나에 대해 말씀드리자면 6~7주라는 짧은 프로젝트 기간동안 기획부터 배포까지 이루어져야 했습니다. 서비스를 기획하며 OAuth, Flask, pandas 등 새로운 기술이 필요하다 판단했고 이를 학습하여 프로젝트 기간 내에 소셜로그인, 데이터 분석 및 시각화, 앱 서버 알림 등을 구현하였습니다. 짧은 시간 내에 필요한 기능을 구현하고자 노력했고 프로젝트는 기능성에서 좋은 평가를 받아 우수상이라는 성과를 달성하였습니다.

저는 소통과 협업을 중시합니다.

프로젝트를 수행하며 소통이 이뤄지지 않을 때 일어나는 비효율을 경험하였습니다. 백엔드에서 처리한 내용을 프론트에서도 작성하거나 요구사항명세에 대한 해석이 달라 기능을 다르게 개발하는 문제였습니다. 모든 개발자는 같은 생각을 가지고 있지 않으며 언급되지 않은 내용은 서로 암묵적으로 다르게 생각할 수 있음을 알 수 있었습니다. 이를 해결하기 위해 기능 모듈화, API 명세서, 트러블 슈팅, 코드 리뷰 등 프로젝트를 문서화하였습니다. 개인마다 경험한 내용을 기록하고, 이를 공유함으로써 프로젝트를 잘 마무리하였으며 이후로도 소통과 협업을 통해 목표를 이루는 것을 중시하고 있습니다.

저는 더 나은 코드를 작성하고자 노력합니다.

첫 프로젝트에서 서비스 로직을 개발하고 코드의 품질을 위해 Refactoring을 진행하였습니다. 하나의 로직에서 두 가지 역할이 이루어지거나 RESTful하지 못한 코드를 수정하였고 배포 이후 개념을 알게되었던 N+1문제를 BatchSize를 통해 개선하였습니다. 이후 코드의 품질을 높이는 방법에 대해 고민을 가지게 되었고 SonarLint 활용과 Junit5를 사용한 테스트 코드 작성, 클린 코드 등을 학습하며 코드의 품질을 높이하고자 노력하고 있습니다.

신입으로서 부족함을 채우고, 학습을 통해 보다 깊이 이해함으로써 성장을 추구하고 있습니다.

RESTful 하다



1. 일관된 인터페이스

2. Client - Server의 구분

3. Stateless

4. Cacheable

5. Layered System

6. Code on Demend

자원에 대한 식별

api/v1/member/1

```
member {  
  name : 권민우,  
  age : 29,  
  gender : male  
}
```


Naming

api/v1/member/1 **VS** api/v1/game/play

소문자를 사용하며 확장자, CRUD 사용하지 않는다.

api/v1/new/users

api/v1/delete/users



@Post

api/v1/users

표현을 통한 자원의 조작

HTTP / 1.0 200 OK

Content-Type : text/plain

Content-length : 5

ggok2

HTTP / 1.0 201 Created

Content-Type :

application/ json

자기 서술적 메시지

content-type

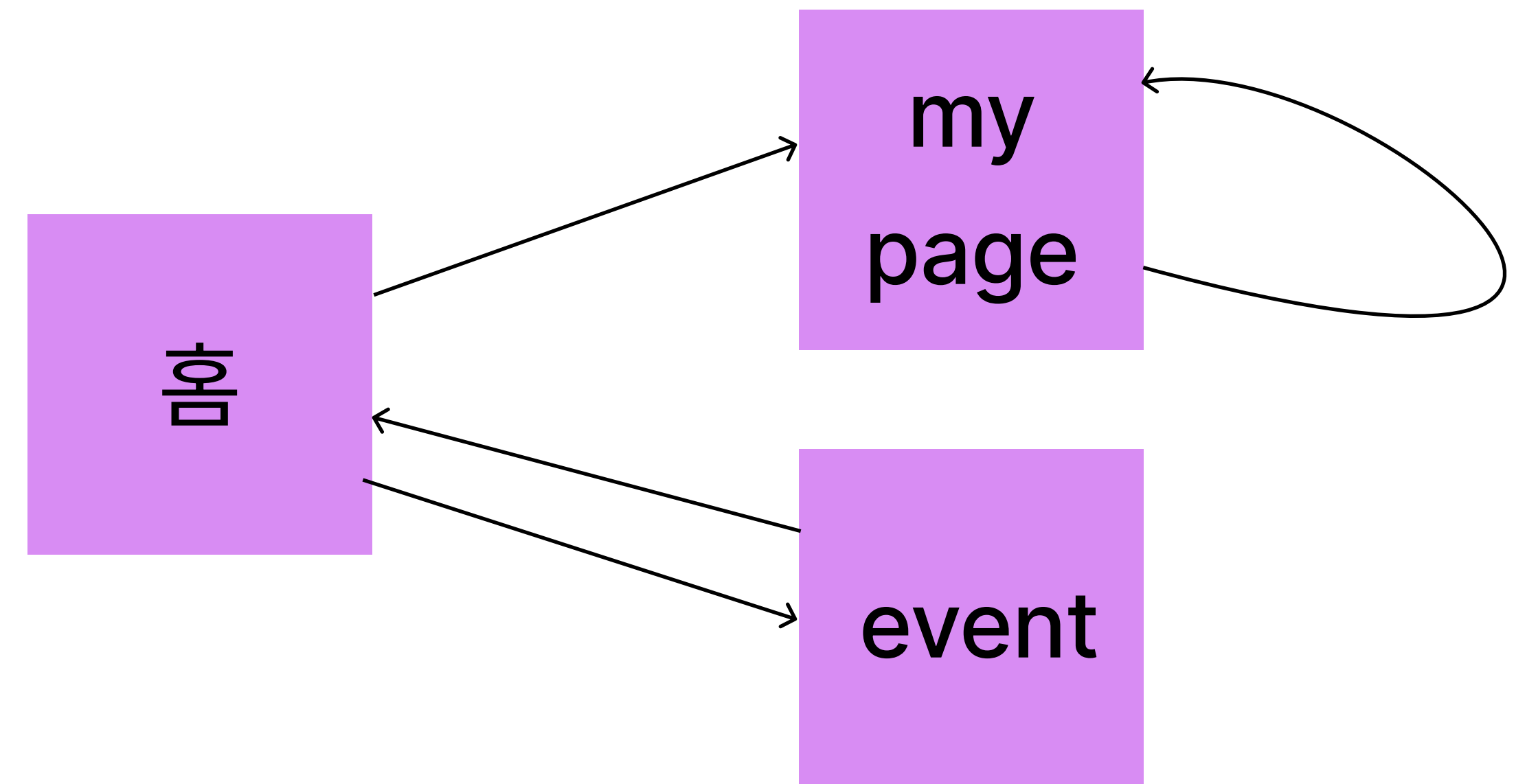
host

multipart

HATEOAS

(Hypermedia As The Engine of Application State)

HTML 등을 통해 클라이언트가
application의 상태를 변경 가능한
인터페이스를 제공해야한다.



REST 얼마나 중요한가

gRPC

(google remote procedure call)

클라이언트 서버 간 개방형 통신

http 2

당근마켓