

Semi Second Project

Group 6: 고가용성을 고려한 만보기 서비스



Title Page

Member&TechStack X +

← → Q Group Members



전종혁

Development



심유진

Infrastructure



이담

Infrastructure



이원준

Infrastructure



Title Page

Member&TechStack X +

← → Q Infra Tech Stack



AWS



Route53



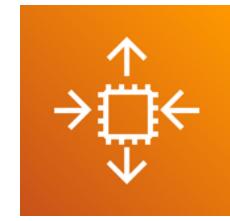
AMI



Application Load Balancer



IAM



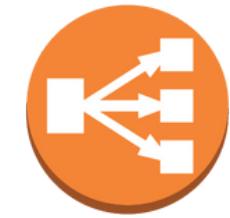
AutoScaling Group



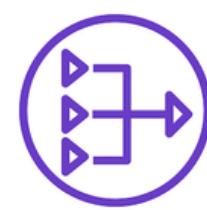
Internet Gateway



RDS



Network Load Balancer



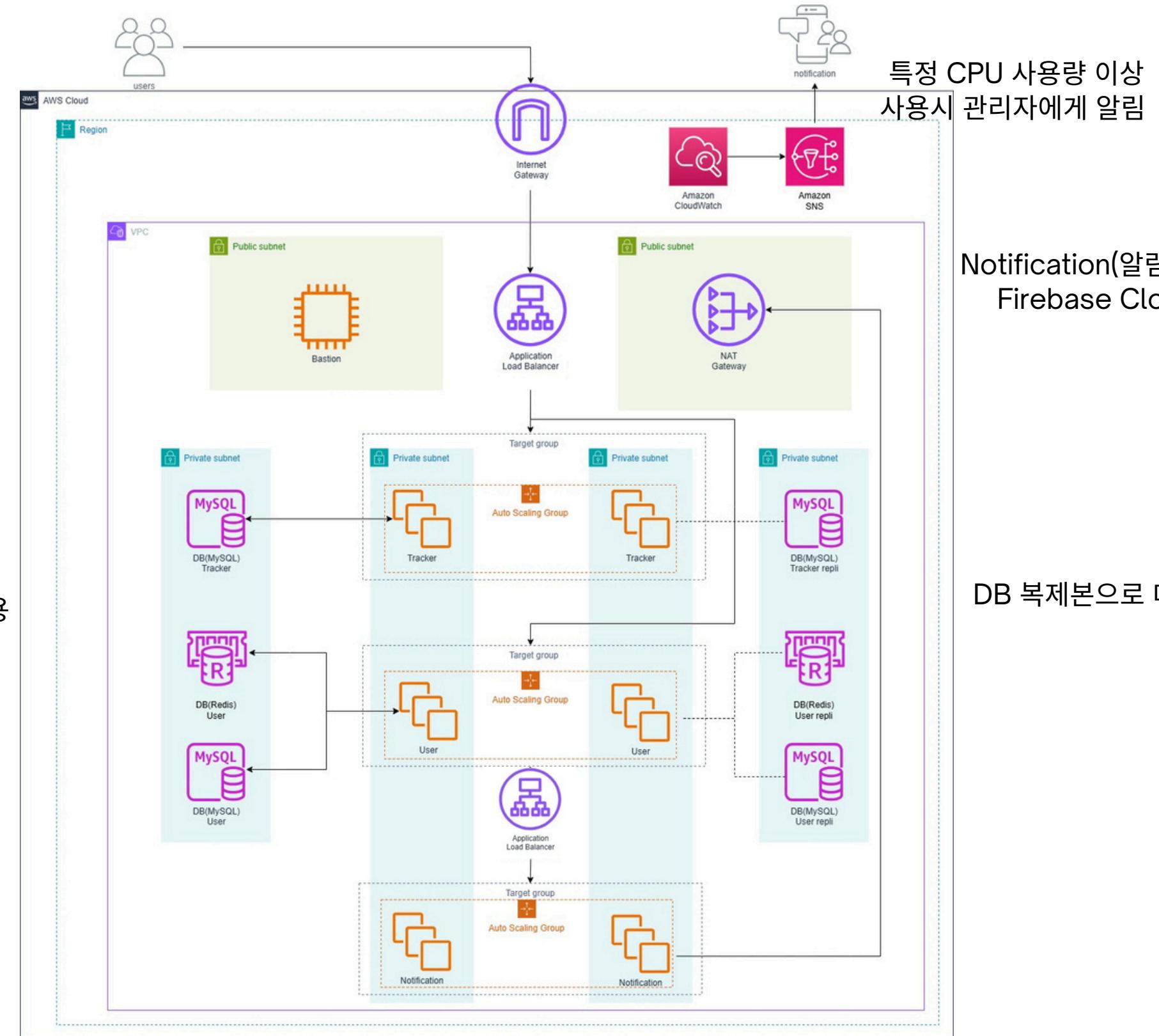
NAT Gateway



CloudWatch

Public subnet에는 Bastion과 NAT Gateway 존재
Bastion을 통해 private subnet 접근 가능

Instance와 DB subnet영역을 구분지어 위치
고가용성 확보를 위해 원본과 복제본을 다른 가용 영역에 배치



DB 복제본으로 대기(Standby)상태



Title Page

Member&TechStack X +

← → Q Development Tech Stack



Flutter



iOS
CoreMotion



Dart



Swift

Frontend (mobile)



MySQL



Redis

Database



Nestjs



TypeScript

Backend



Firebase Cloud Messaging



Login



Sign in

External Server



← → G Q Concerns related to development

gRPC vs REST

Runner
main

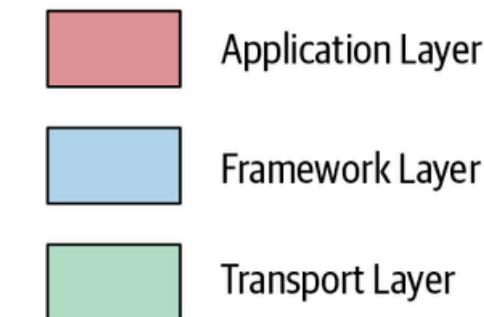
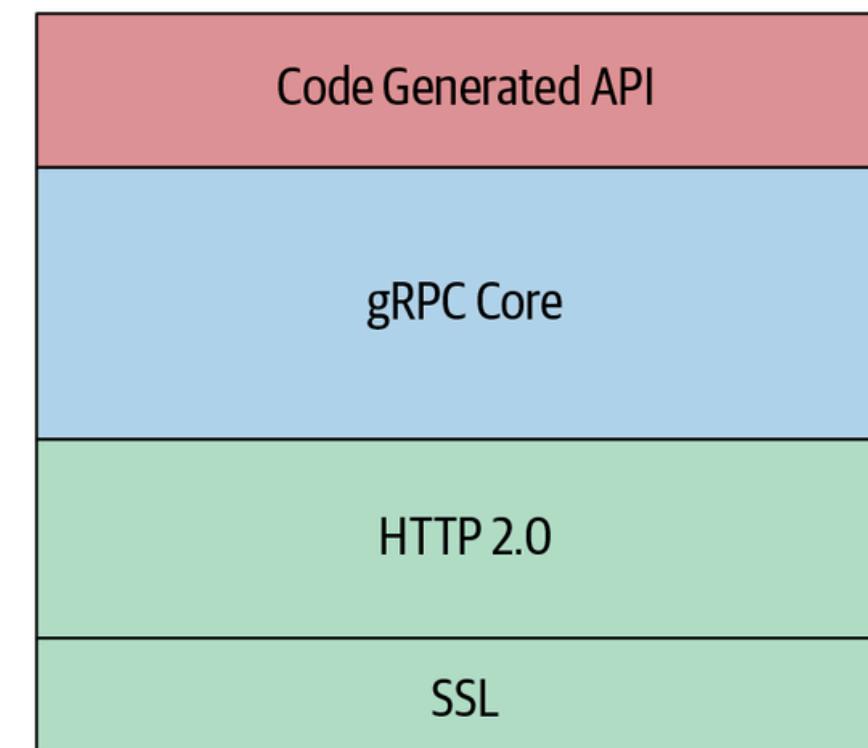
Any iOS Finished running Runner on Jonghyuck's iPhone ①

CoreMotion CMPedometer

CoreMotion CMPedometer CMPedometer

```
231 open class CMPedometer : NSObject {
232     open class func isPedometerEventTrackingAvailable() -> Bool
293
294
295     /*
296     * authorizationStatus
297     *
298     * Discussion:
299     *     Returns the current authorization status for pedometer.
300     */
301 @available(iOS 11.0, *)
302 open class func authorizationStatus() -> CMAuthorizationStatus
303
304
305     /*
306     * queryPedometerDataFromDate:toDate:withHandler:
307     *
308     * Discussion:
309     *     Queries for the user's pedestrian activity in the given time range. Data
310     *     is available for up to 7 days. The data returned is computed from a
311     *     system-wide history that is continuously being collected in the
312     *     background. The result is returned on a serial queue.
313     */
314 open func queryPedometerData(from start: Date, to end: Date, withHandler handler: @escaping
    CMPedometerHandler)
315
316
317     /*
318     * startPedometerUpdatesFromDate:withHandler:
319     *
320     * Discussion:
321     *     Starts a series of continuous pedometer updates to the
322     *     handler on a serial queue. For each update, the app
323     *     will receive the cumulative pedestrian activity since the
324     *     start date specified and the timestamp associated with the
325     *     latest determination. If the app is backgrounded and resumed
326     *     at a later time, the app will receive all of the pedestrian
327     *     activity accumulated during the background period in the
328     *     very next update.

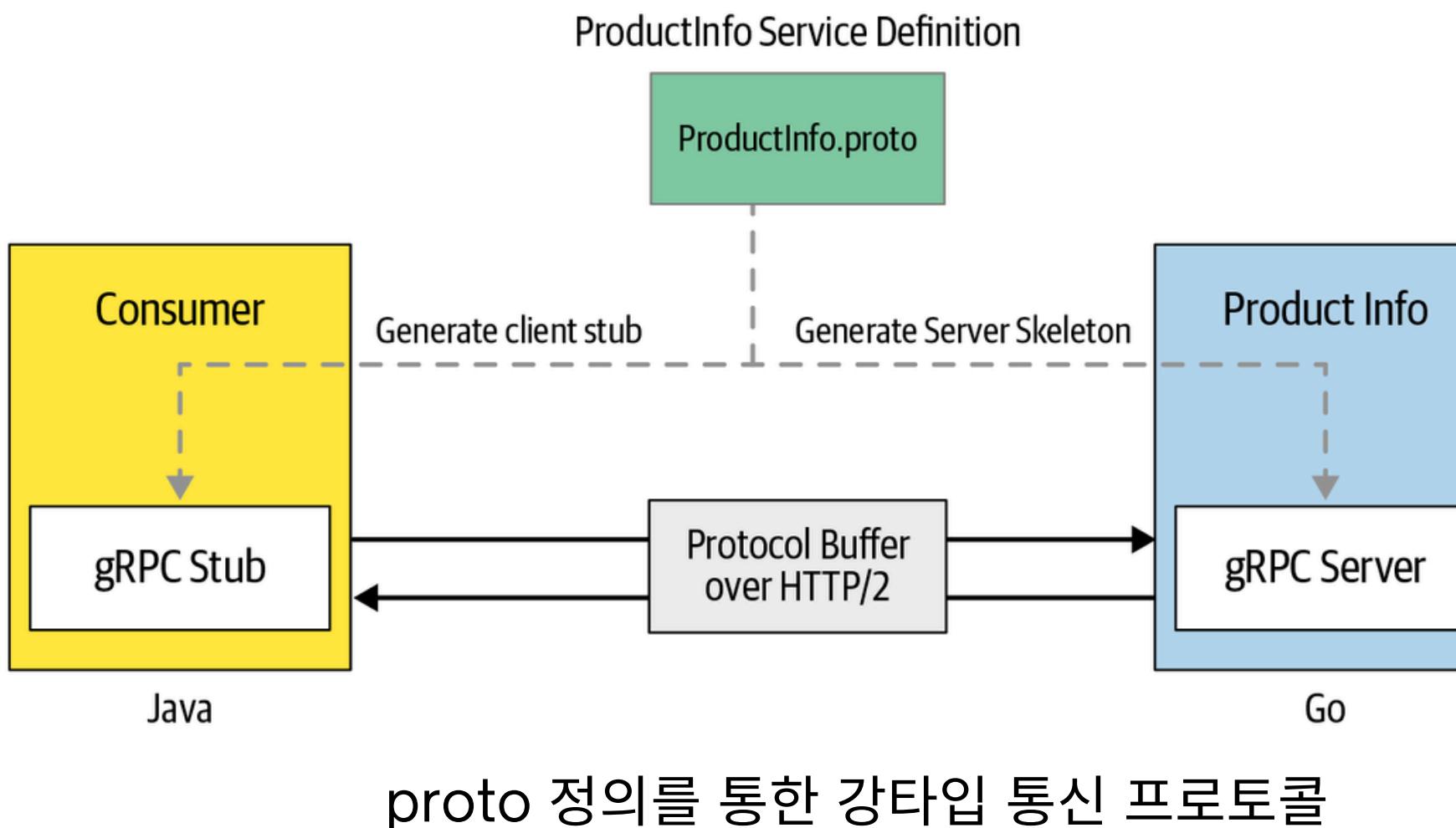
```



gRPC는 실시간 양방향 스트리밍을 지원
만보계의 실시간 카운트를 서버에 전송

← → Q Concerns related to development

gRPC vs REST



```
syntax = "proto3";

package step;

// 서비스 정의
service StepService {
    rpc RecordSteps(StepInput) returns (StepResponse);
    rpc GetStepsByUuids(GetStepsByUuidsInput) returns (StepResponse);
    rpc StreamSteps(GetStepsByUuidsInput) returns (stream StepInput);
    rpc StreamUserSteps(stream StepInput) returns (StepResponse);
}

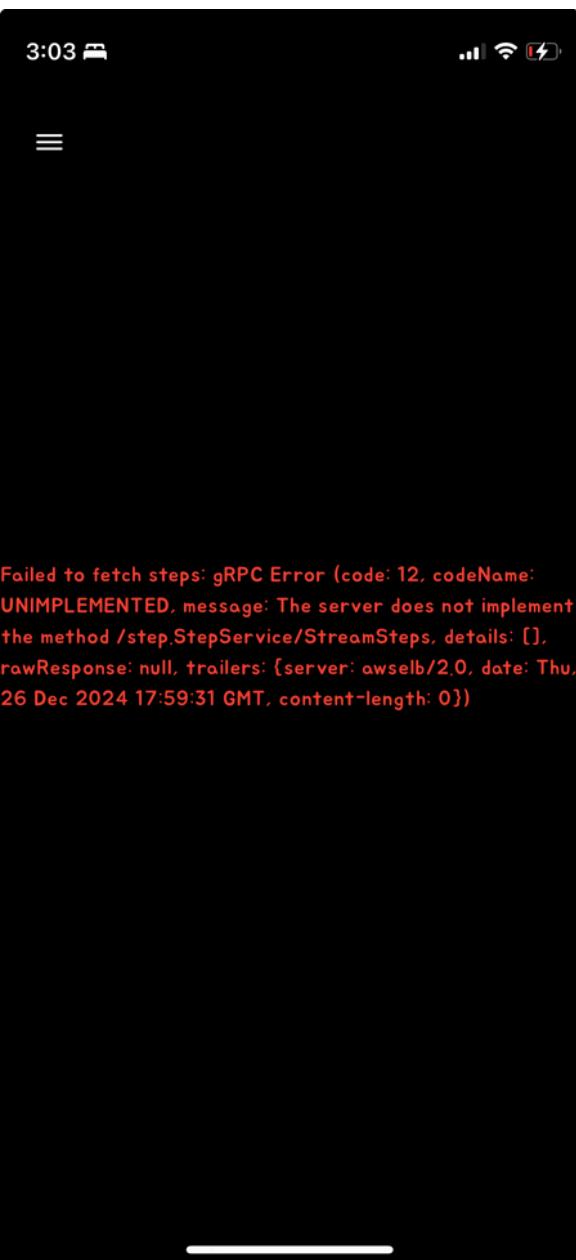
// 사용자 걸음수 기록 요청 메시지
message StepInput {
    string userId = 1; // 사용자 UUID
    int32 steps = 2; // 걸음수
    string date = 3; // 날짜
}

// 여러 사용자 걸음수 요청 메시지
message GetStepsByUuidsInput {
    repeated string userIds = 1; // 사용자 UUID 리스트
    string date = 2; // 특정 날짜
}

// 단일 사용자 걸음수 응답 메시지
message StepResponse {
    string message = 1; // 성공 메시지
}
```

← → Q Concerns related to development

gRPC vs REST



```
/// 사용자 걸음수 실시간 전송 (StreamUserSteps)
Future<void> streamUserSteps(Stream<StepInput> stepsStream) async {
    await for (final response in _grpcClient.streamUserSteps(stepsStream)) {
        print('Server response: ${response.message}');
    }
}
```

```
@GrpcMethod('StepService', 'StreamSteps')
streamSteps(data: { userId: string }): Observable<{ date: string; steps: number }> {
    return new Observable((subscriber) => {
        const interval = setInterval(async () => {
            const steps = await this.getStepsByUserUseCase.execute(data.userId);
            const latestStep = steps[steps.length - 1]; // 가장 최근의 데이터
            subscriber.next({ date: latestStep.date, steps: latestStep.steps });
        }, 1000);
    });
}
```

← → Q Concerns related to development

Monolith vs MSA

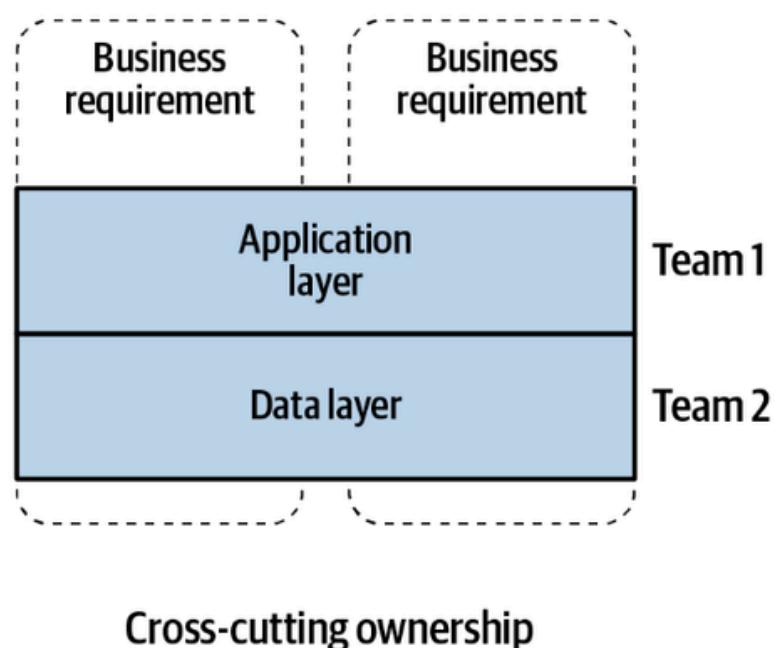
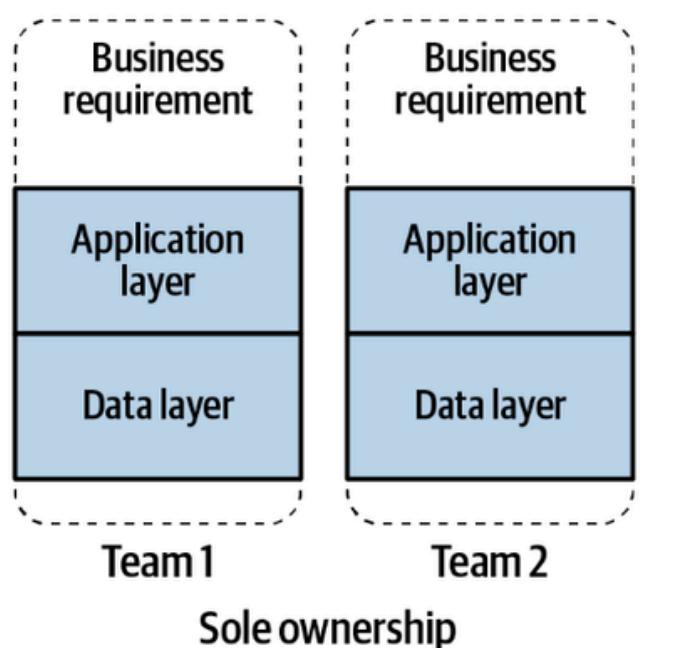
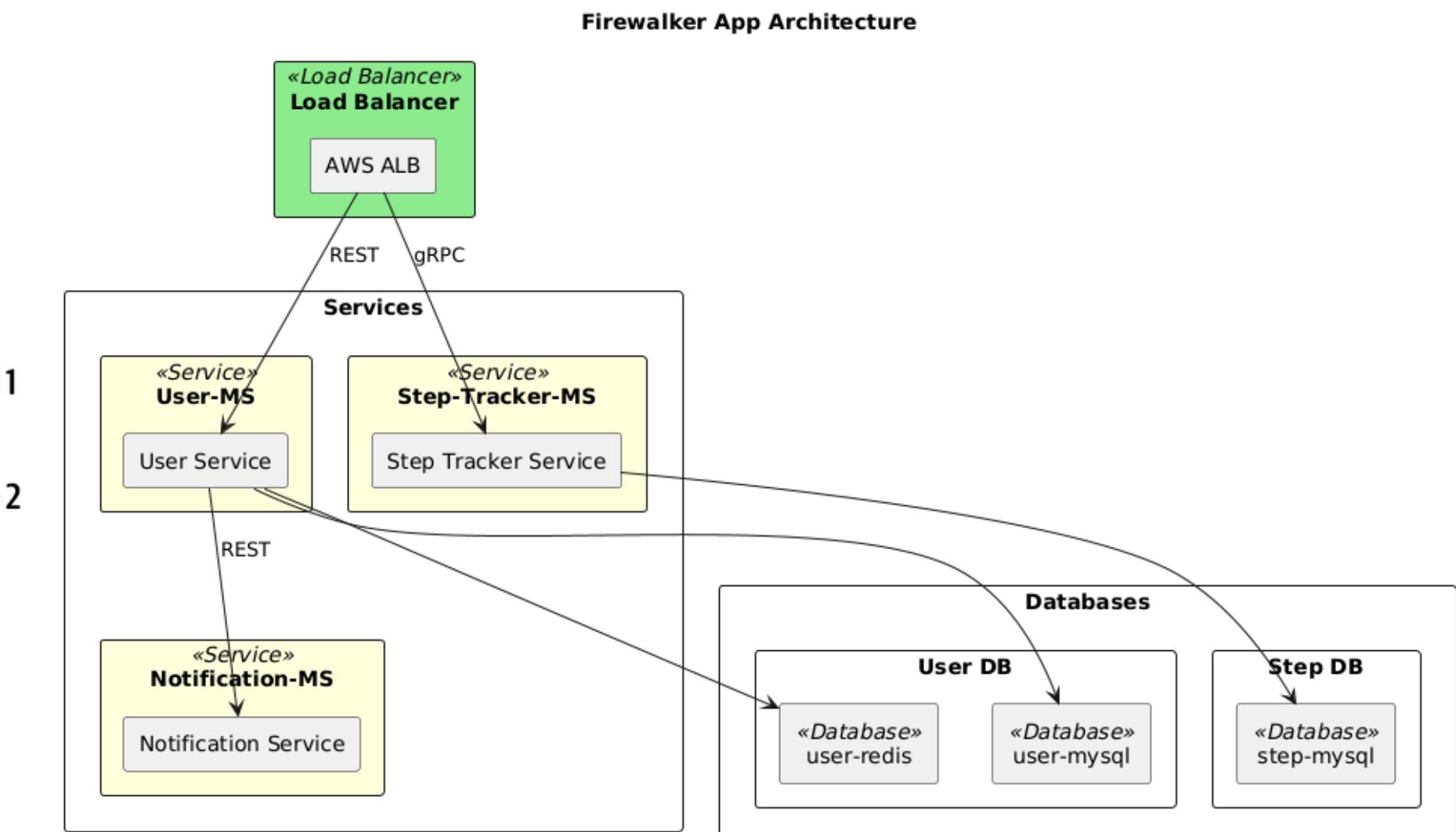


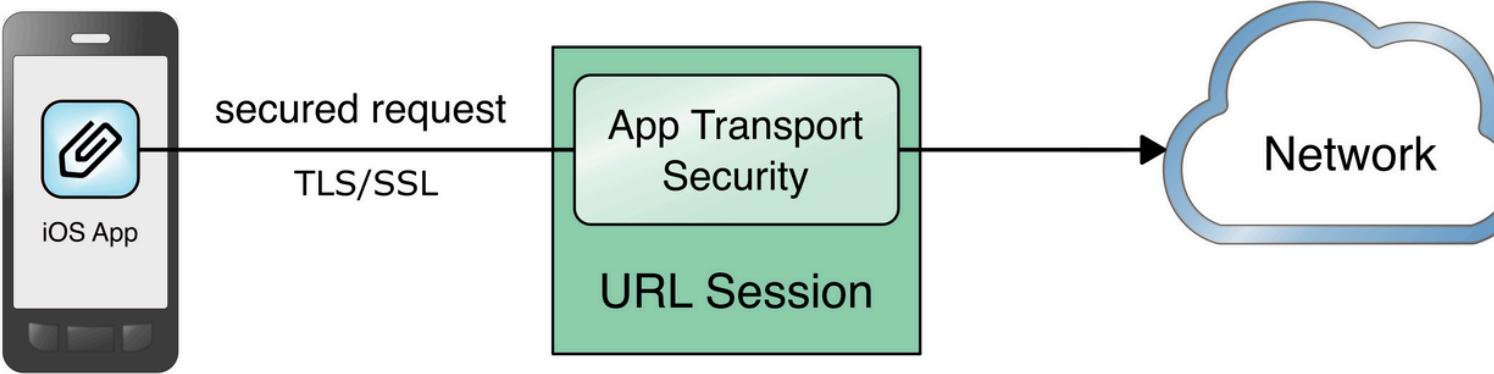
Figure 1-2. Alignment on business contexts versus on technological contexts



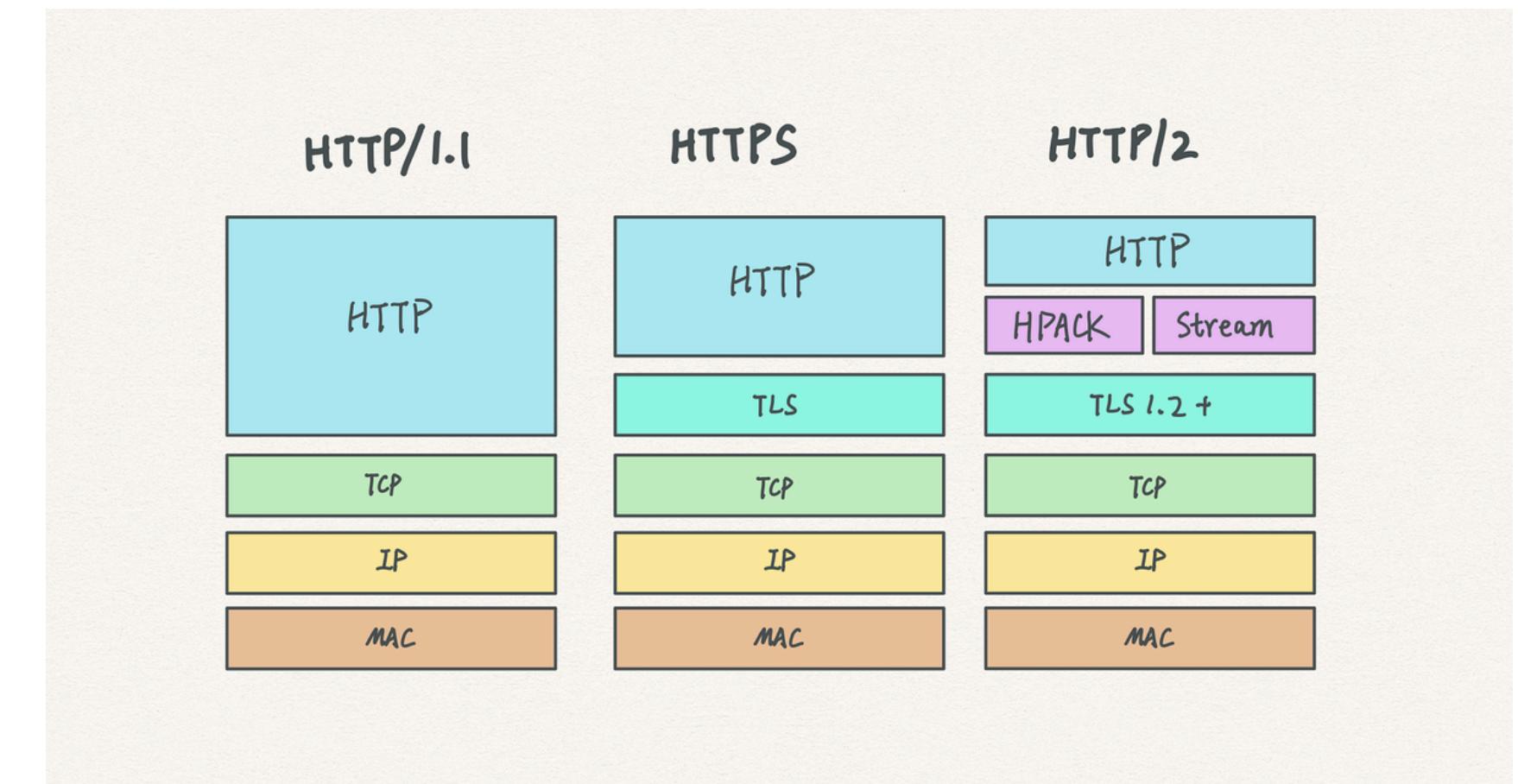
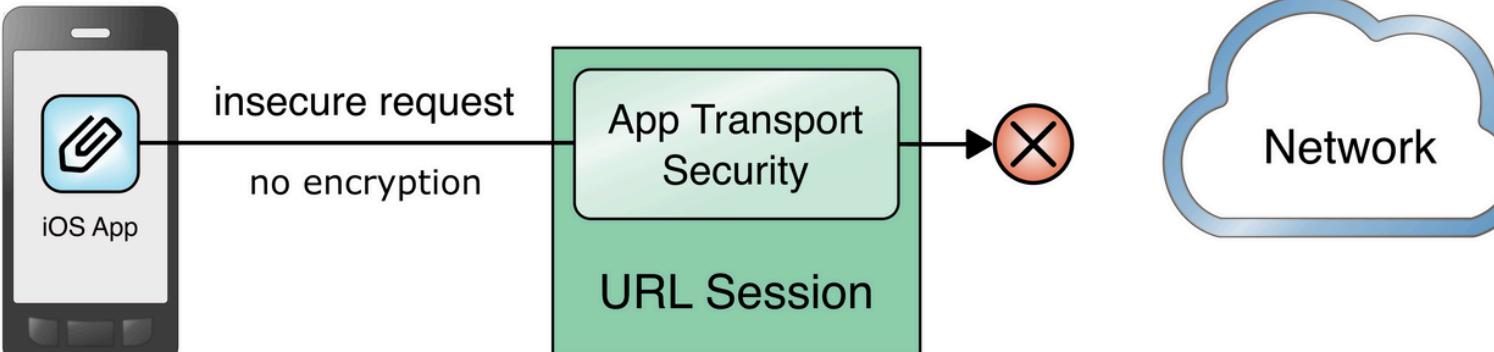
← → Q Concerns related to development

HTTPS

ATS Compliant Scenario



ATS Noncompliant Scenario



[Title Page](#)[Member&Tech
Stack](#)[Development](#)[Infra](#)[X](#)[+](#)[**← → Q**](#) [What is Terraform](#)

Terraform을 사용한 인프라 구성

인프라를 코드로 관리하고, 재사용 하기 위해 Terraform을 통한 IaC 구현

```
Plan: 55 to add, 0 to change, 0 to destroy.
aws_eip.eip_natgw: Creating...
aws_vpc.test_vpc: Creating...
aws_eip.eip_natgw: Creation complete after 1s [id=eipalloc-0c37b3825ffe7fb9]
aws_vpc.test_vpc: Creation complete after 1s [id=vpc-027c324158182d7d0]
aws_security_group.mysql_sg: Creating...
aws_security_group.alb_sg: Creating...
aws_security_group.user_sg: Creating...
aws_subnet.test_pub02: Creating...
aws_subnet.test_pub01: Creating...
aws_security_group.redis_sg: Creating...
aws_lb.test_alb: Creation complete after 3m3s [id=arn:aws:elasticloadbalancing:ap-northeast-2:054037133389:loadbalancer/app/test-alb/b7cb0c3b0ad14bb7]
aws_route53_record.www: Creating...
aws_lb_listener.tracker_listener: Creating...
aws_lb_listener.user_listener: Creating...
aws_lb_listener.user_listener: Creation complete after 0s [id=arn:aws:elasticloadbalancing:ap-northeast-2:054037133389:listener/app/test-alb/b7cb0c3b0ad14bb7/06d405db86951df8]
aws_lb_listener.tracker_listener: Creation complete after 0s [id=arn:aws:elasticloadbalancing:ap-northeast-2:054037133389:listener/app/test-alb/b7cb0c3b0ad14bb7/7244bfcfffc06c04]
aws_db_instance.trackerdb: Creation complete after 3m23s [id=db-GI654F4X4W5RNWKHP56VJVP52M]
aws_db_instance.trackerdb: Creation complete after 3m23s [id=db-GI654F4X4W5RNWKHP56VJVP52M]
aws_db_instance.userdb: Creation complete after 3m24s [id=db-UVUWDOGEYXZYXVS6B23HGQQNPQ]
aws_route53_record.www: Still creating... [30s elapsed]
aws_route53_record.www: Creation complete after 38s [id=Z082827615UQ174WWGA8_www.sesacyujin.site_A]

Apply complete! Resources: 55 added, 0 changed, 0 destroyed.
```



← → Q Advantages of Terraform

Terraform 사용 vs 콘솔에서의 수동 인프라 생성

테라폼 사용		테라폼 사용 x
배포 속도	자동화로 신속한 배포 가능	수동 작업으로 인해 느린 배포 속도
재사용성	재사용 가능	재사용 어려움
버전 관리	Git 등으로 코드 변경 이력 관리 가능	변경 이력 관리 불가능



Title Page

Member&Tech
Stack

Development

Infra

X

+

HTTPS Protocol

gRPC HTTPS, ACM

```
# acm certificate
data "aws_acm_certificate" "acm_certificate" {
  domain      = "sesacyujin.site"
  types        = ["AMAZON_ISSUED"]
  most_recent = true
}
```

```
port          = "50051"
protocol     = "HTTPS"
ssl_policy   = "ELBSecurityPolicy-2016-08"
certificate_arn = data.aws_acm_certificate.acm_certificate.arn
```

실시간 양방향 통신을 위해 gRPC Protocol 사용
HTTP/2 (HTTPS) 기반으로 동작

HTTPS의 경우 애플리케이션의 보안 강화하기 위한
ACM 인증서 필수
ALB에 **ACM 인증서 적용**

인증서 ID	도메인 이름	유형	상태
d9a66e25-920e-4a3a-a665-4ef0cf33fe29	sesacyujin.site	Amazon 발급	발급됨

terraform code로 acm 생성

[Title Page](#)[Member&Tech
Stack](#)[Development](#)[Infra](#)[X](#)[+](#)[← → Q Cloud watch](#)

SNS 생성 및 이메일 구독 설정

AWS Notification - Subscription Confirmation

보낸사람 AWS Notifications <no-reply@sns.amazonaws.com>

받는사람 0375_@naver.com

2024년 12월 25일 (수) 오후 5:25

영어 → 한국어 번역하기

You have chosen to subscribe to the topic:

[arn:aws:sns:ap-northeast-2:054037133389:test-sns](#)

To confirm this subscription, click or visit the link below (If this was in error no action is necessary):

[Confirm subscription](#)

Please do not reply directly to this email. If you wish to remove yourself from receiving all future SNS subscription confirmation requests please send an email to [sns-opt-out](#)



Simple Notification Service

Subscription confirmed!

You have successfully subscribed.

Your subscription's id is:

[arn:aws:sns:ap-northeast-2:054037133389:test-sns:fd753e02-e115-460b-bfbe-9d4150b38383](#)

If it was not your intention to subscribe, [click here to unsubscribe](#).

구독 확인으로 해당 주제에 대한 문제 발생시 자동 이메일 알람

[Title Page](#)[Member&Tech
Stack](#)[Project1](#)[Infra](#)[X](#)[+](#)[**← → Q**](#) [Cloud watch](#)

문제 발생시 알람

ALARM: "tracker-alarm" in Asia Pacific (Seoul)

Alarm Details:

- Name: tracker-alarm
- Description:
- State Change: INSUFFICIENT_DATA -> ALARM
- Reason for State Change: Threshold Crossed: 1 out of the last 1 datapoints
- Timestamp: Wednesday 25 December, 2024 14:24:03 UTC
- AWS Account: 340752825860
- Alarm Arn: arn:aws:cloudwatch:ap-northeast-2:340752825860:ala

설정한 기준이 비정상일 경우 이메일로 알람이 전달됨

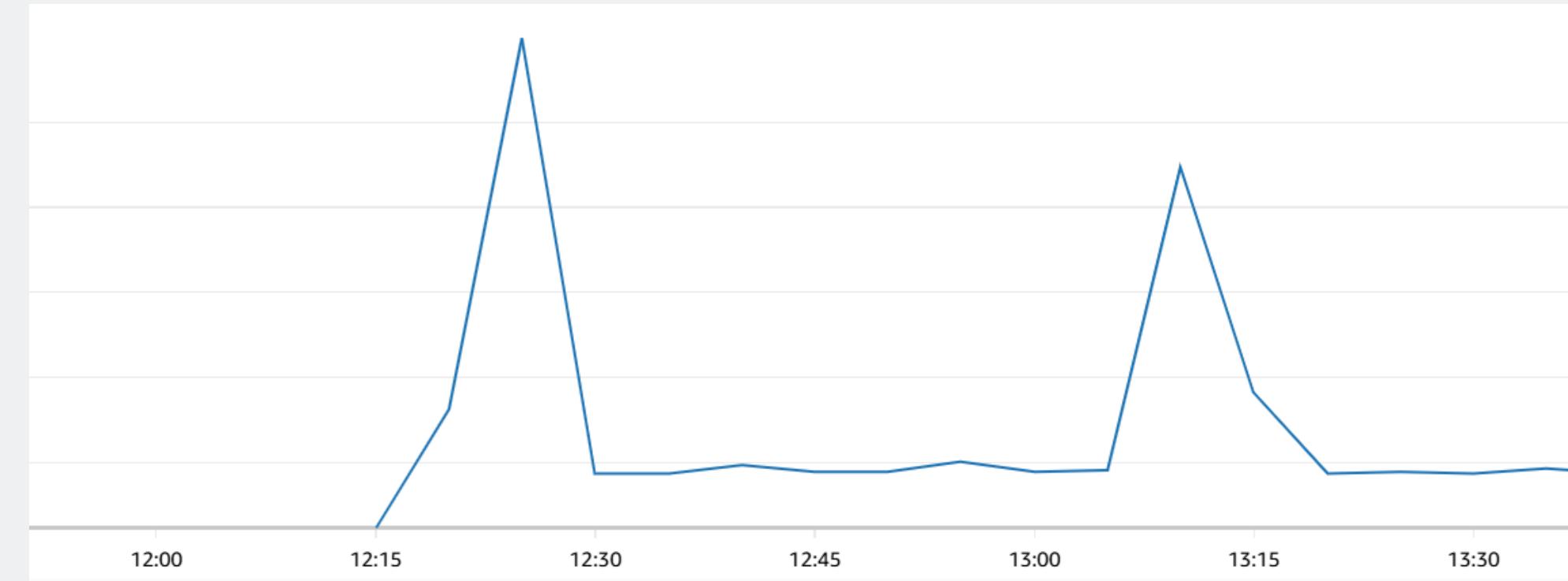
[Title Page](#)[Member&Tech
Stack](#)[Project1](#)[Infra](#)

X

+

[← → Q Cloud watch](#)

Monitoring



시간대별로 사용량을 실시간 관찰 가능
기준값을 넘는 지점의 시간대 파악 가능



Title Page

Member&Tech
Stack

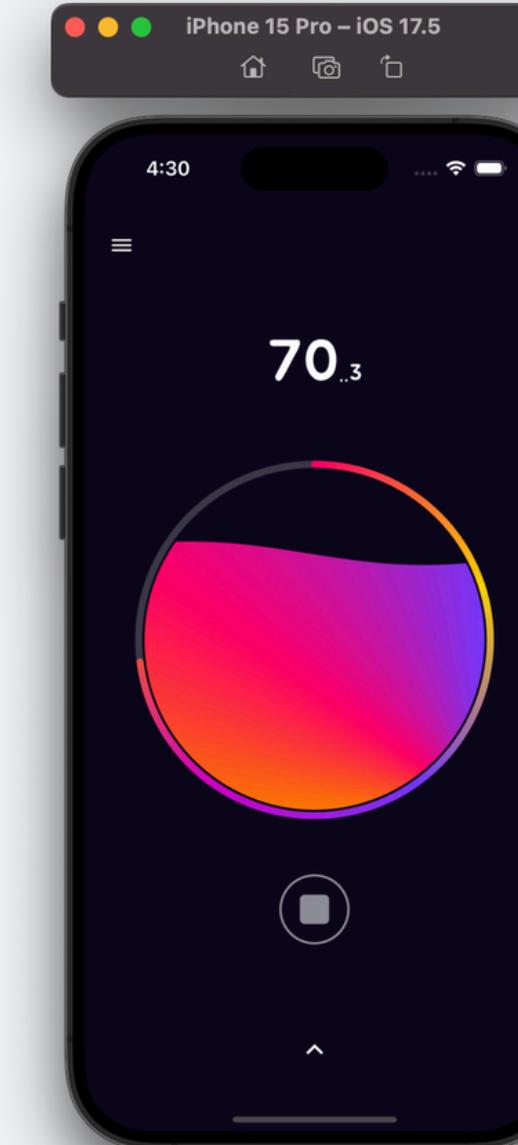
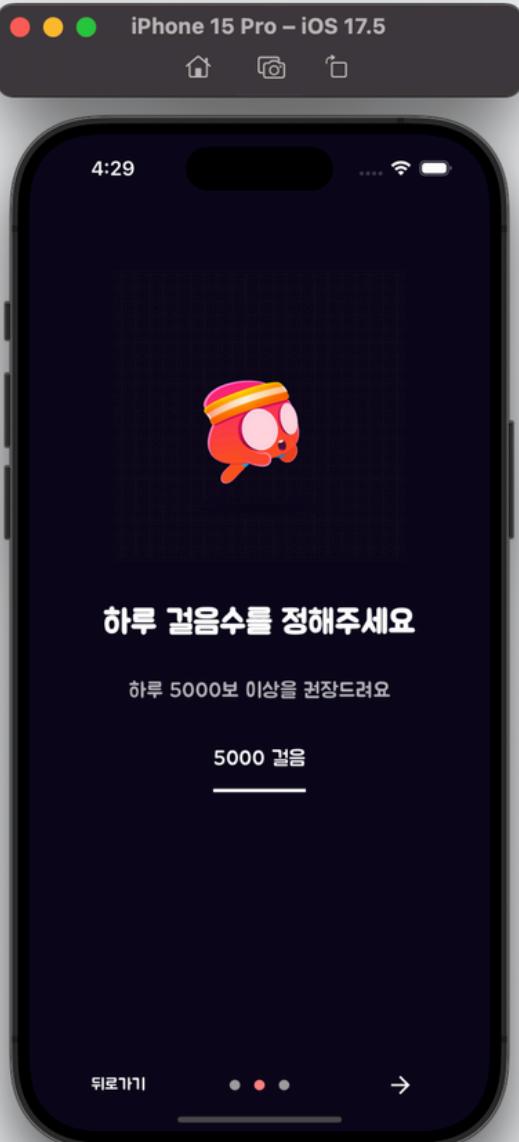
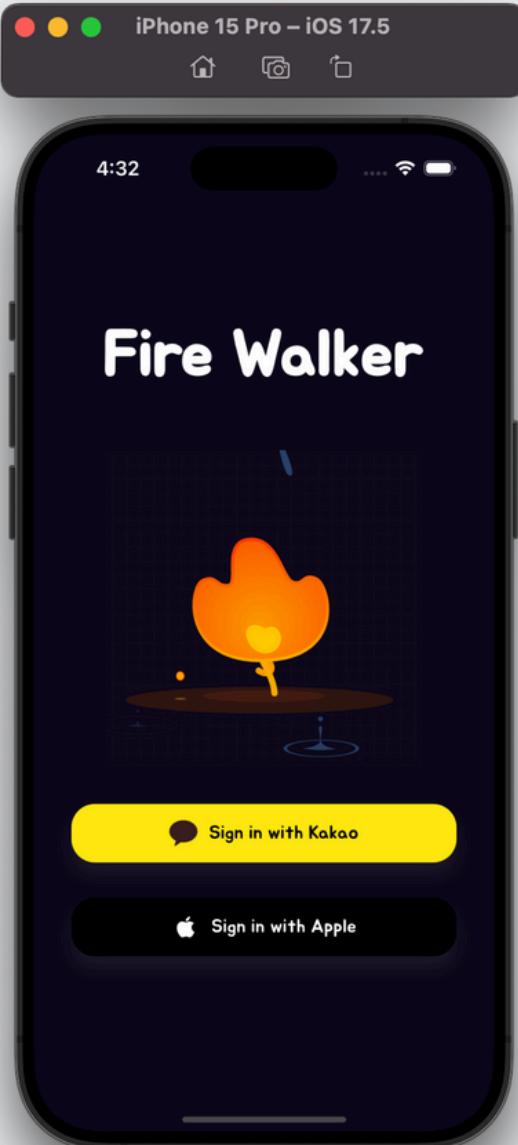
Development

Infra

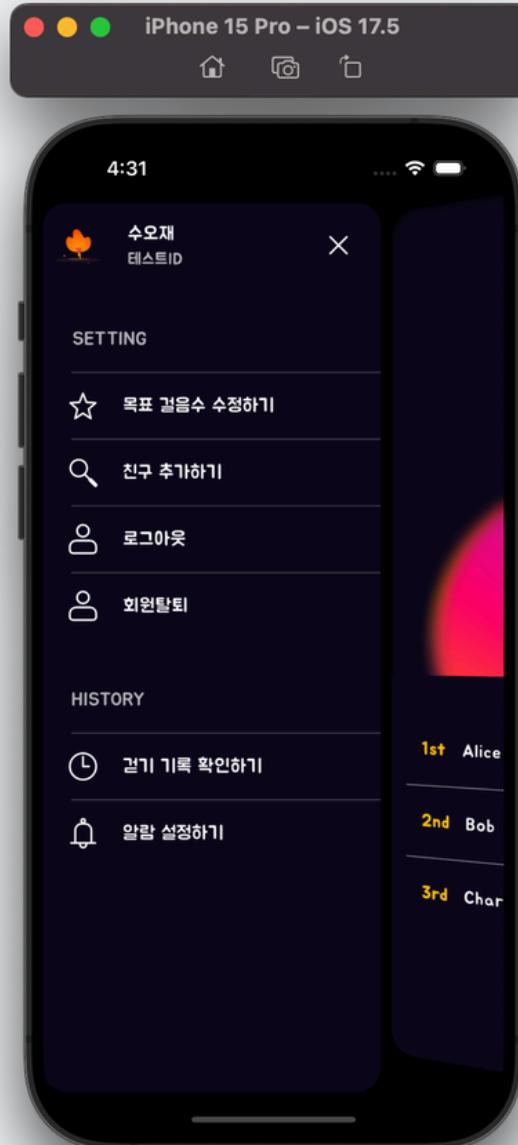
Demonstration



← → Q Application Test



Rank	User	Steps
1st	Alice	12000 steps
2nd	Bob	11000 steps
3rd	Charlie	10500 steps



← → Q Trouble Shooting



Terraform · Depends on

```
aws_lb.test_alb: Creating...

Error: creating ELBv2 application Load Balancer (test-alb): operation error Elastic Load Balancing v2: CreateLoadBalancer, https response error StatusCode: 400, RequestID: 07d4e3b5-2c23-4576-8db2-d9a6d1a9e8fa, InvalidSubnet: VPC vpc-0e1c75dab7e247f9b has no internet gateway

with aws_lb.test_alb,
on 09.alb.tf line 39, in resource "aws_lb" "test_alb":
39: resource "aws_lb" "test_alb" {
```

Public Routing Table에서 depends on Internet Gateway를 명시하지 않을 경우 IGW 생성 지연으로 인한 오류가 발생

```
resource "aws_route_table" "public_rt" {
  vpc_id = aws_vpc.test_vpc.id

  route {
    cidr_block = "10.0.0.0/16"
    gateway_id = "local"
  }

  route {
    cidr_block = "0.0.0.0/0"
    gateway_id = aws_internet_gateway.igw.id
  }

  depends_on = [aws_internet_gateway.igw]
}
```

리소스 간의 의존성을 명시적으로 정의하여
Terraform 실행 시 발생할 수 있는 오류를 방지



AMI Instance Settings

```
resource "aws_instance" "tracker" {
    ami                         = data.aws_ami.ubuntu_node.id
    instance_type                = "t2.micro"
    subnet_id                    = aws_subnet.test_pri01.id
    key_name                     = "key"
    security_groups              = [aws_security_group.tracker_sg.id]

    root_block_device {
        volume_size = 8
    }

    tags = {
        Name = "tracker"
    }
}

resource "aws_instance" "tracker" {
    ami                         = "ami-0d17fd2ad58b6eb4e"
    instance_type                = "t2.micro"
    subnet_id                    = aws_subnet.test_pri01.id
    key_name                     = "key"
    security_groups              = [aws_security_group.tracker_sg.id]

    root_block_device {
        volume_size = 8
    }

    tags = {
        Name = "tracker"
    }
}
```

기존의 AMI 이미지로 인스턴스 생성시 자동으로 최신 ubuntu 이미지 가져오기 가능

테스트용으로 다른 계정에서 AMI를 공유 받은후 같은 코드를 테라폼으로 진행했을때 문제점 발생

-> 공유된 ami의 ID를 정확히 명시해야 코드가 실행 되는 것을 확인

[Title Page](#)[Member&Tech Stack](#)[Development](#)[Infra](#)[Demonstration](#)[Trouble Shooting](#)[Conclusion](#)[X](#) [+](#)[← → Q Conclusion](#)

아쉬운점



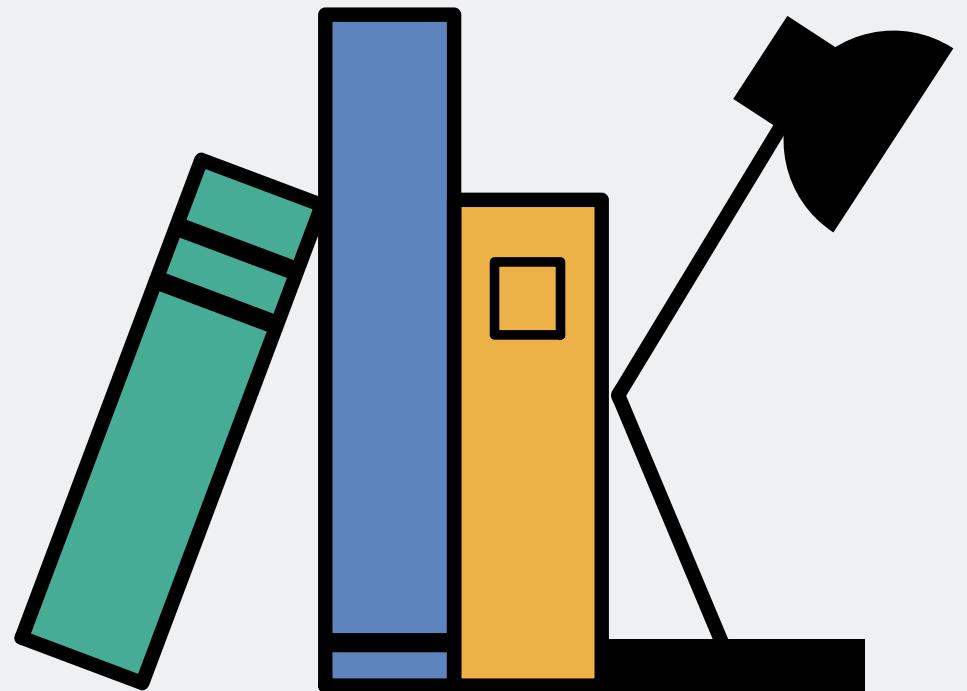
짧은 프로젝트 기간



컨테이너화된 애플리케이션 관리를 위한 쿠버네티스 사용



테라폼 모듈화

[Title Page](#)[Member&Tech Stack](#)[Development](#)[Infra](#)[Demonstration](#)[Trouble Shooting](#)[Conclusion](#)[Last Page](#)[← →](#) [Q](#) [🔍](#)

Thank you!