



포팅 메뉴얼

☰ 태그

1. 개발 환경

[EC2 서버 인스턴스 사양](#)

[UI / UX Design](#)

[형상 관리](#)

[OS](#)

[이슈 관리](#)

[커뮤니케이션](#)

[Mobile App](#)

[Watch App](#)

[BackEnd](#)

[Infra](#)

[IDE](#)

[Database](#)

[Monitoring](#)

2. 설정 파일

[BackEnd](#)

[application.yml](#)

3. CI/CD 구축

[Docker](#)

[DockerFile](#)

[Jenkins](#)

[Credential](#)

[New Item Option](#)

[Jenkins Pipe-line](#)

[GitLab Webhook](#)

[Argo CD \(Helm \)](#)

[Chart.yaml](#)

[values.yaml](#)

[deployment.yaml](#)

[hpa.yaml](#)

[service.yaml](#)

[Argo CD 설정](#)

[Argo CD 결과물](#)

BackEnd 서버 배포 현황
Kubernetes 내부 Pod 현황

화면

메인 화면

보상 화면

뽑기 화면

운동 기록 상세 화면

운동 프로그램 목록 화면

운동 프로그램 추가 화면

친구 화면

강아지 집 화면

환경 설정 화면

1. 개발 환경

EC2 서버 인스턴스 사양

- CPU 정보 : Intel Xeon(R) Core 4개
- RAM : 16GB
- Disk : 300GB

UI / UX Design

- Figma

형상 관리

- GitLab(Dev Code)
- GitHub(HelmChart)

OS

- Ubuntu 20.04
- Windows 10

이슈 관리

- Jira

커뮤니케이션

- Mattermost
- Notion
- Discord

Mobile App

Tech	Version
Flutter	
SecurityStore	

Watch App

Tech	Version
Kotlin	
Retrofit	

BackEnd

Tech	Version
Java JDK	17
Spring boot	3.2.5
Gradle	
Swagger	

Infra

- Jenkins
- Docker
- MicroK8S
- Argo CD
- Helm Chart : v3

IDE

- IntelliJ
- Visual Studio Code
- Android Studio

Database

- MySQL

Monitoring

- Grafana + Prometheus

2. 설정 파일

BackEnd

application.yml

```
server:
  port: 8081

spring:
  application:
    name: member-service

  datasource:
    url: jdbc:mysql://${DATABASE_URL}:${DATABASE_PORT}/${DATABASE_NAME}
    username: ${DATABASE_USERNAME}
    password: ${DATABASE_PASSWORD}

  jpa:
```

```

hibernate:
  ddl-auto: update
  show-sql: true
  generate-ddl: true
  properties:
    hibernate:
      format_sql: true
  defer-datasource-initialization: true

eureka:
  client:
    register-with-eureka: true
    fetch-registry: true
    service-url:
      defaultZone: https://eureka.gaenari.kr/eureka

springdoc:
  api-docs:
    enabled: true
    version: openapi_3_1
  enable-spring-security: true
  default-consumes-media-type: application/json
  default-produces-media-type: application/json

token:
  expiration_time: 86400000 # 60*60*24*1000
  secret: ${SECRET_TOKEN} # 임의의 값

feign:
  url:
    member-service: http://member-service.member.svc.cluster.
    inventory-service: http://inventory-service.inventory.svc
    program-service: http://program-service.program.svc.clust
    exercise-record-service: http://exercise-record-service.e
    challenge-service: http://challenge-service.challenge.svc
  client:
    config:
      default:

```

```

        connectTimeout: 5000
        readTimeout: 5000
    circuitbreaker:
        enabled: true

resilience4j:
    circuitbreaker:
        configs:
            default:
                failure-rate-threshold: 50 # 실패율 임계값 설정 (50%)
                slow-call-rate-threshold: 80 # slowCall 발생 시 서버 스레드 점유율 임계값 설정 (80%)
                slow-call-duration-threshold: 5s # 위와 같은 이유로 5초를 초과하여 호출이 지연된 경우
                register-health-indicator: true # 회로 차단기의 상태를 헬스 체크로 등록
                wait-duration-in-open-state: 10s # HALF_OPEN 상태로 빨리 전환할 때까지 대기할 시간
                automaticTransitionFromOpenToHalfOpenEnabled: true # HALF_OPEN 상태에서 자동으로 OPEN으로
                sliding-window-type: COUNT_BASED # 윈도우 타입: COUNT_BASED (호출 횟수) 또는 TIME_BASED (시간)
                sliding-window-size: 100 # 최근 100번의 호출을 분석하여 결정을 내림
                minimum-number-of-calls: 10 # 최소 호출 횟수, 이 수를 넘어야 결정을 내림
        instances:
            default:
                baseConfig: default
            member-service-circuit-breaker:
                baseConfig: default
            inventory-service-circuit-breaker:
                baseConfig: default
            program-service-circuit-breaker:
                baseConfig: default
            exercise-record-service-circuit-breaker:
                baseConfig: default
            challenge-service-circuit-breaker:
                baseConfig: default
    timelimiter:
        configs:
            default:
                timeoutDuration: 6s # slowCallDurationThreshold보다는 짧게 설정
                cancelRunningFuture: true
        instances:
            default:

```

```

    baseConfig: default
member-service-circuit-breaker:
    baseConfig: default
inventory-service-circuit-breaker:
    baseConfig: default
program-service-circuit-breaker:
    baseConfig: default
exercise-record-service-circuit-breaker:
    baseConfig: default
challenge-service-circuit-breaker:
    baseConfig: default

```

3. CI/CD 구축

Docker

DockerFile









```

FROM openjdk:17-ea-jdk-slim
VOLUME /tmp
ARG JAR_FILE=./build/libs/*.jar
COPY ${JAR_FILE} MemberService.jar
ENTRYPOINT ["java", "-jar", "MemberService.jar", "--spring.pro

```

Jenkins

Credential

		System	(global)	gitlab-xinx-token	jaesin463@gmail.com/***** (GitLab Access Token with ID)
		System	(global)	docker-xinx-token	xinx/***** (DockerHub Access Token with ID)
		System	(global)	github-xinx-token	jaesin463/***** (Github Access Token with ID)
		System	(global)	sonarqube-token	Sonarqube 토큰 연결

New Item Option

☒ Do not allow concurrent builds
| ☐ Abort previous builds ?

☐ Do not allow the pipeline to resume if the controller restarts

☐ GitHub project

GitLab Connection

Git_connections



☐ Use alternative credential

☐ Pipeline speed/durability override ?

☐ Preserve stashes from completed builds ?

☐ Throttle builds ?

☐ 오래된 빌드 삭제 ?

☐ 이 빌드는 매개변수가 있습니다 ?

Build Triggers

☐ Build after other projects are built ?

☐ Build periodically ?

- ☒ Enable [ci-skip] ?
- ☒ Ignore WIP Merge Requests ?

Labels that launch a build if they are added (comma-separated) ?

- ☒ Set build description to build cause (eg. Merge request or Git Push) ?
- ☐ Build on successful pipeline events

Pending build name for pipeline ?

- ☐ Cancel pending merge request builds on update ?

Allowed branches

- ☒ Allow all branches to trigger this job ?
- ☐ Filter branches by name ?
- ☐ Filter branches by regex ?
- ☐ Filter merge request by label

Secret token ?

901f4848fa4270ebc4eff8b804e454ec

☒ Build when a change is pushed to GitLab. GitLab webhook URL: <http://jenkins:8080/project/Fcm-Service> ?

Enabled GitLab triggers

- ☒ Push Events ?
- ☐ Push Events in case of branch delete ?
- ☒ Opened Merge Request Events ?
- ☐ Build only if new commits were pushed to Merge Request ?
- ☐ Accepted Merge Request Events ?
- ☐ Closed Merge Request Events ?

Rebuild open Merge Requests ?

Never ▼

- ☒ Approved Merge Requests (EE-only) ?
- ☒ Comments ?

Comment (regex) for triggering a build ?

Jenkins please retry a build

고급 ▼

- ☐ GitHub hook trigger for GITScm polling ?
- ☐ Poll SCM ?

Jenkins Pipe-line

```
pipeline {
  agent {
    kubernetes {
      yaml '''
        apiVersion: v1
        kind: Pod
        spec:
          containers:
            - name: docker
              image: docker:latest
              command:
                - cat
              tty: true
      '''
    }
  }
}
```

```

        volumeMounts:
        - mountPath: /var/run/docker.sock
          name: docker-sock
      volumes:
      - name: docker-sock
        hostPath:
          path: /var/run/docker.sock
    ...
  }
}

environment{
  DOCKER_HUB_CREDS = 'docker-xinx-e-token'
}

stages{

  stage('Checkout'){
    steps{
      container('docker'){
        checkout scmGit(branches: [[name: 'deploy
      }
    }
  }

  stage('Gradle & Docker Build'){
    steps{
      dir('fcm-service'){
        sh 'ls'
        sh 'chmod +x ./gradlew'
        sh './gradlew clean build -x test'

        container('docker'){
          sh "docker build -t xinx-e/gaenari-fcm
        }
      }
    }
  }
}

```

```

stage('Docker Hub Login'){
  steps{
    container('docker'){
      withCredentials([usernamePassword(credentialId: 'docker-hub-username', username: 'xinxe', password: 'xinxe1234567890')]){
        sh "echo ${DOCKER_PASSWORD} | docker login --password-stdin"
      }
    }
  }
}

stage('Docker Image Push'){
  steps{
    container('docker'){
      sh "docker push xinxe/gaenari-fcm:${env.BUILD_NUMBER}"
    }
  }
}

stage('Helm Chart'){
  steps{
    dir('fcm'){
      checkout scmGit(branches: [[name: '*/main', branch: '*/main']])
      sh 'git checkout -b main'
      sh "sed -i 's/appVersion:.*\$/appVersion: ${BUILD_NUMBER}/'"
      sh 'git config --global user.email "jaesi@nongae.com"'
      sh 'git config --global user.name "xinxe"'
      sh 'git config --global --add safe.directory /'
      sh "git add fcm/Chart.yaml"
      sh "git commit -m '[UPDATE] Fcm ${BUILD_NUMBER}'"

      withCredentials([gitUsernamePassword(credentialId: 'github-token', username: 'xinxe', password: 'xinxe1234567890')]){
        sh "git push -u origin main"
      }
    }
  }
}
}

```

```

post {
  always {
    container('docker') {
      sh 'docker logout'
      sh 'docker rmi xinxe/gaenari-fcm:${BUILD_NUMB
    }
  }

  success {
    script {
      def Author_ID = sh(script: "git show -s --pre
      def Author_Name = sh(script: "git show -s --p
      mattermostSend (color: 'good',
      message: "Fcm-Service 빌드 성공!: #${env.BUILD_
      endpoint: 'https://meeting.ssafy.com/hooks/b6
      channel: 'BOT')
    }
  }

  failure {
    script {
      def Author_ID = sh(script: "git show -s --pre
      def Author_Name = sh(script: "git show -s --p
      mattermostSend (color: 'danger',
      message: "Fcm-Service 빌드 실패..: #${env.BUILD
      endpoint: 'https://meeting.ssafy.com/hooks/b6
      channel: 'BOT')
    }
  }
}
}

```

GitLab Webhook

https://jenkins.gaenari.kr/project/Program-Service <small>Push events SSL Verification: enabled</small>	Test ▾ Edit Delete
https://jenkins.gaenari.kr/project/Discovery-Service <small>Push events SSL Verification: enabled</small>	Test ▾ Edit Delete
https://jenkins.gaenari.kr/project/Apigateway-Service <small>Push events SSL Verification: enabled</small>	Test ▾ Edit Delete
https://jenkins.gaenari.kr/project/Inventory-Service <small>Push events SSL Verification: enabled</small>	Test ▾ Edit Delete
https://jenkins.gaenari.kr/project/Member-Service <small>Push events SSL Verification: enabled</small>	Test ▾ Edit Delete
https://jenkins.gaenari.kr/project/Exercise-Record-Service <small>Push events SSL Verification: enabled</small>	Test ▾ Edit Delete
https://jenkins.gaenari.kr/project/Gaenari <small>Push events SSL Verification: enabled</small>	Test ▾ Edit Delete
https://jenkins.gaenari.kr/project/Challenge-Service <small>Push events SSL Verification: enabled</small>	Test ▾ Edit Delete
https://jenkins.gaenari.kr/project/Fcm-Service <small>Push events SSL Verification: enabled</small>	Test ▾ Edit Delete

Argo CD (Helm)

Chart.yaml

```

apiVersion: v2
name: member
description: Member's Helm chart for Kubernetes
type: application
version: 0.1.0
appVersion: 42
maintainers:
  - name: Jaesin Lee
    email: jaesin463@gmail.com

```

values.yaml

```

member:
  enabled: true
  replicas: 1
  ports:
    - name: member
      port: 80
      targetPort: 8081

```

```
image:
  repository: xinxe/gaenari-member
```

deployment.yaml

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: member-deployment
spec:
  replicas: {{ .Values.member.replicas }}
  selector:
    matchLabels:
      app: member
  template:
    metadata:
      labels:
        app: member
    spec:
      containers:
        - name: member
          image: {{ .Values.member.image.repository }}:{{ .Char
          env:
            - name: DATABASE_USERNAME
              valueFrom:
                secretKeyRef:
                  name: member-app-secrets
                  key: database.username
            - name: DATABASE_PASSWORD
              valueFrom:
                secretKeyRef:
                  name: member-app-secrets
                  key: database.password
            - name: DATABASE_URL
              valueFrom:
                secretKeyRef:
                  name: member-app-secrets
                  key: database.url
```

```

      - name: DATABASE_PORT
        valueFrom:
          secretKeyRef:
            name: member-app-secrets
            key: database.port
      - name: DATABASE_USER_DB_NAME
        valueFrom:
          secretKeyRef:
            name: member-app-secrets
            key: database.dbName
      - name: SECRET_TOKEN
        valueFrom:
          secretKeyRef:
            name: member-app-secrets
            key: secret.token
    ports:
      - containerPort: {{ (index .Values.member.ports 0) }}
    volumeMounts:
      - name: tz-seoul
        mountPath: /etc/localtime
    volumes:
      - name: tz-seoul
        hostPath:
          path: /usr/share/zoneinfo/Asia/Seoul

```

hpa.yaml

```

apiVersion: autoscaling/v2
kind: HorizontalPodAutoscaler
metadata:
  name: member-hpa
spec:
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: member-deployment
  minReplicas: 1
  maxReplicas: 3

```

```
metrics:
  - type: "Resource"
    resource:
      name: "cpu"
      target:
        type: "Utilization"
        averageUtilization: 50
```

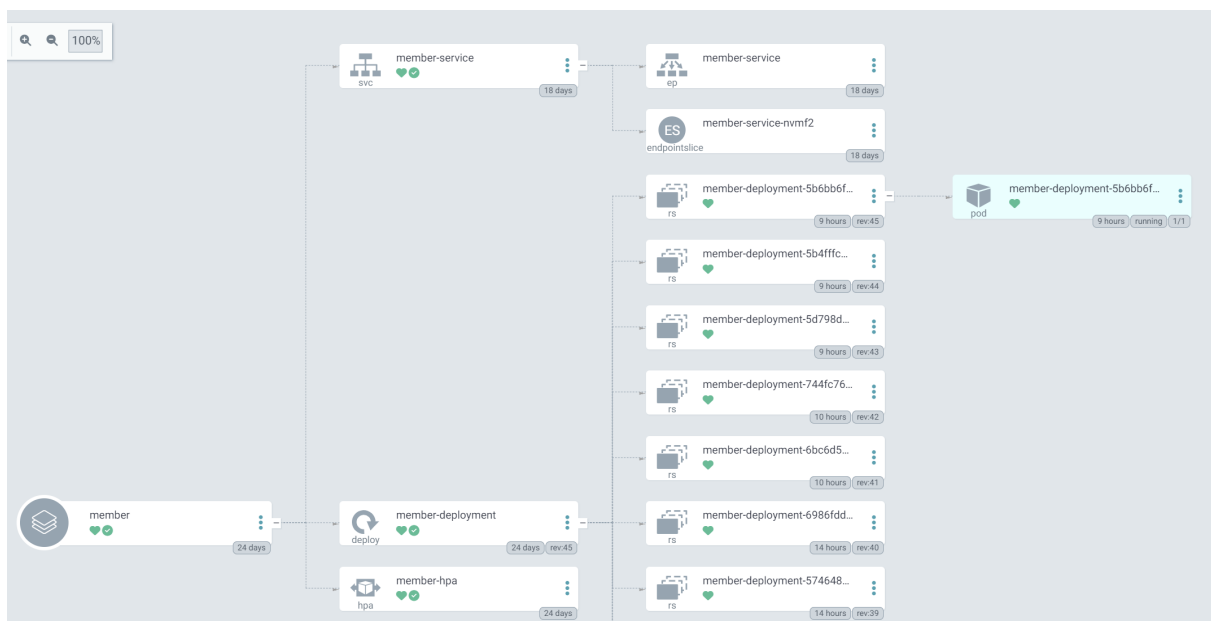
service.yaml

```
apiVersion: v1
kind: Service
metadata:
  name: member-service
spec:
  selector:
    app: member
  ports:
    - protocol: TCP
      port: {{ (index .Values.member.ports 0).port }}
      targetPort: {{ (index .Values.member.ports 0).targetPort }}
  type: ClusterIP
```

Argo CD 설정

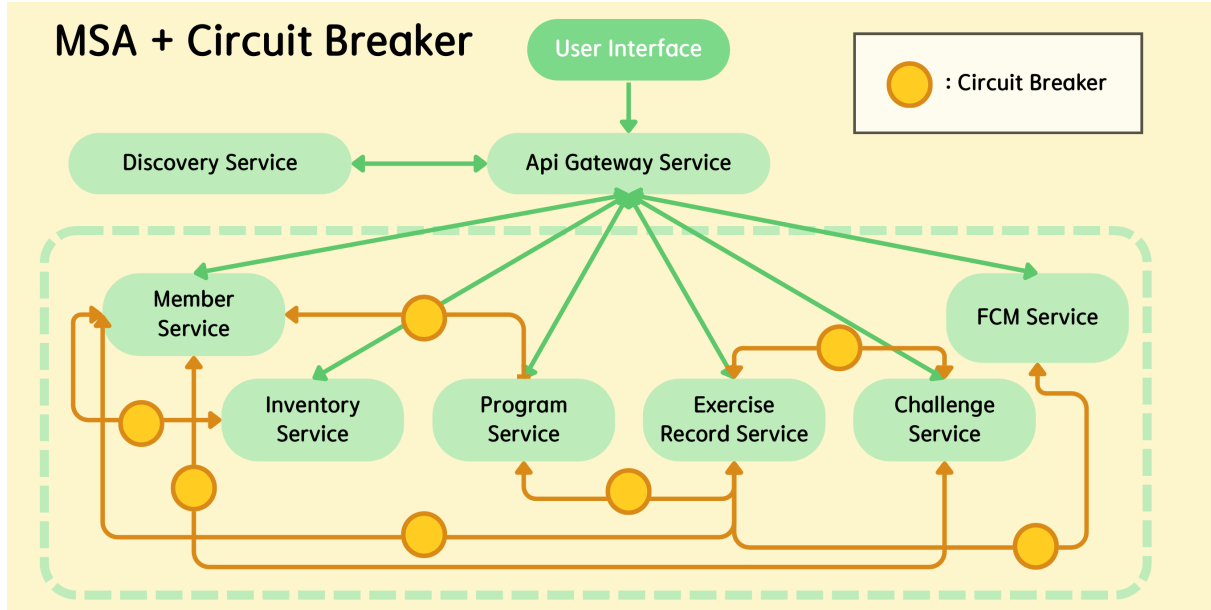
MEMBER		EDIT
PROJECT	default	
ANNOTATIONS		
CLUSTER	in-cluster (https://kubernetes.default.svc)	
NAMESPACE	member	
CREATED AT	04/25/2024 21:21:40 (25 days ago)	
REPO URL	https://github.com/jaesin463/gaenari-helm-chart.git	
TARGET REVISION	HEAD	
PATH	member	
SYNC OPTIONS	<input checked="" type="checkbox"/> CreateNamespace	
RETRY OPTIONS	Retry disabled	
STATUS	Synced to HEAD (15a434d)	
HEALTH	Healthy	
LINKS		
IMAGES	xinxe/gaenari-member:42	

Argo CD 결과물



BackEnd 서버 배포 현황

MSA + Circuit Breaker



Kubernetes 내부 Pod 현황

```
ubuntu@ip-172-26-5-77:~$ kubectl get pod -A
```

NAMESPACE	NAME	READY	STATUS	RESTARTS	AGE
argocd	argocd-cd-argocd-application-controller-0	1/1	Running	1 (5d18h ago)	26d
argocd	argocd-cd-argocd-applicationset-controller-557899997c-2zgwb	1/1	Running	1 (5d18h ago)	26d
argocd	argocd-cd-argocd-dex-server-597bf5b45d-fjlqp	1/1	Running	1 (5d18h ago)	26d
argocd	argocd-cd-argocd-notifications-controller-6fd445cdb8-gjgvs	1/1	Running	1 (5d18h ago)	26d
argocd	argocd-cd-argocd-redis-6595dcfcf5-2wqvg	1/1	Running	1 (5d18h ago)	26d
argocd	argocd-cd-argocd-repo-server-787c9cfd4-v5rb4	1/1	Running	2 (5d18h ago)	26d
argocd	argocd-cd-argocd-server-6fdb84579-5kmzz	1/1	Running	5 (20h ago)	26d
cert-manager	cert-manager-7cf97bbd47-pw2l6	1/1	Running	9 (7h26m ago)	26d
cert-manager	cert-manager-cainjector-99677759d-bmsfw	1/1	Running	8 (7h26m ago)	26d
cert-manager	cert-manager-webhook-8486cb8479-94slx	1/1	Running	3 (5d18h ago)	26d
challenge	challenge-deployment-6fffd75694-nqmb4	1/1	Running	0	16h
container-registry	registry-6c9fcc695f-7pbl	1/1	Running	1 (5d18h ago)	27d
default	myclient-7nq99	1/1	Running	1 (5d18h ago)	16d
discovery	discovery-deployment-7b79bf5f-s5s87	1/1	Running	0	5d18h
exercise-record	exercise-record-deployment-5c8d6dc8db-q2twn	1/1	Running	0	16h
fcm	fcm-deployment-6c66c494c5-pg7sf	1/1	Running	0	7h24m
gateway	gateway-deployment-7668f87c58-bjm5b	1/1	Running	0	3d21h
ingress-nginx	ingress-nginx-admission-create-9zltb	0/1	Completed	0	26d
ingress-nginx	ingress-nginx-controller-7d498fdd84-l94g6	0/1	CrashLoopBackOff	10482 (2m47s ago)	26d
ingress	nginx-ingress-microk8s-controller-p7mfv	1/1	Running	1 (5d18h ago)	27d
inventory	inventory-deployment-65b9cd5c9d-s88kp	1/1	Running	0	16m
jenkins	discovery-service-3-chxm9-jzl0l-w2m4h	0/2	Unknown	0	5d19h
jenkins	jenkins-0	2/2	Running	8 (5d18h ago)	26d
kafka	my-cluster-entity-operator-7f6cd75856-n4h8z	2/2	Running	9 (5d18h ago)	16d
kafka	my-cluster-kafka-0	1/1	Running	5 (5d18h ago)	16d
kafka	my-cluster-kafka-1	1/1	Running	4 (5d18h ago)	16d
kafka	my-cluster-kafka-2	1/1	Running	5 (5d18h ago)	16d
kafka	my-cluster-zookeeper-0	1/1	Running	4 (5d18h ago)	16d
kafka	my-cluster-zookeeper-1	1/1	Running	4 (5d18h ago)	16d
kafka	my-cluster-zookeeper-2	1/1	Running	3 (5d18h ago)	16d
kafka	strimzi-cluster-operator-86df856f74-s9hm5	1/1	Running	8 (16h ago)	16d
kube-system	calico-kube-controllers-77bd7c5b-trvhz	1/1	Running	1 (5d18h ago)	27d
kube-system	calico-node-skxwv	1/1	Running	1 (5d18h ago)	12d
kube-system	coredns-864597b5fd-jk6s2	1/1	Running	1 (5d18h ago)	23d
kube-system	dashboard-metrics-scraper-5657497c4c-nlm4w	1/1	Running	1 (5d18h ago)	23d
kube-system	hostpath-provisioner-756cd956bc-54grn	1/1	Running	13 (7h26m ago)	27d
kube-system	kubernetes-dashboard-54b48fbf9-l7j7v	1/1	Running	3 (16h ago)	23d
kube-system	metrics-server-848968bdcd-tdc64	1/1	Running	5 (5d18h ago)	27d
member	member-deployment-5b6bb6f6cc-b5wr5	1/1	Running	0	11h
monitoring	grafana-75fbbcf578-s9zlm	1/1	Running	1 (5d18h ago)	21d
monitoring	prometheus-alertmanager-0	1/1	Running	3 (5d18h ago)	21d
monitoring	prometheus-kube-state-metrics-6b7d7b9bd9-fpzbv	1/1	Running	1 (5d18h ago)	21d
monitoring	prometheus-prometheus-node-exporter-lkmf2	1/1	Running	2 (5d18h ago)	21d
monitoring	prometheus-prometheus-pushgateway-568fbf799-qwvlq	1/1	Running	1 (5d18h ago)	21d
monitoring	prometheus-server-579dc9cfd4-p4cbd	2/2	Running	2 (5d18h ago)	21d
mysql	mysql-0	1/1	Running	4 (2d18h ago)	26d
program	program-deployment-5b9b69c99c-vff5q	1/1	Running	0	5h
rabbitmq	rabbitmq-0	1/1	Running	1 (5d18h ago)	17d
sonarqube	sonarqube-postgresql-0	1/1	Running	1 (5d18h ago)	5d19h
sonarqube	sonarqube-sonarqube-lts-0	1/1	Running	1 (5d18h ago)	5d19h

화면

메인 화면

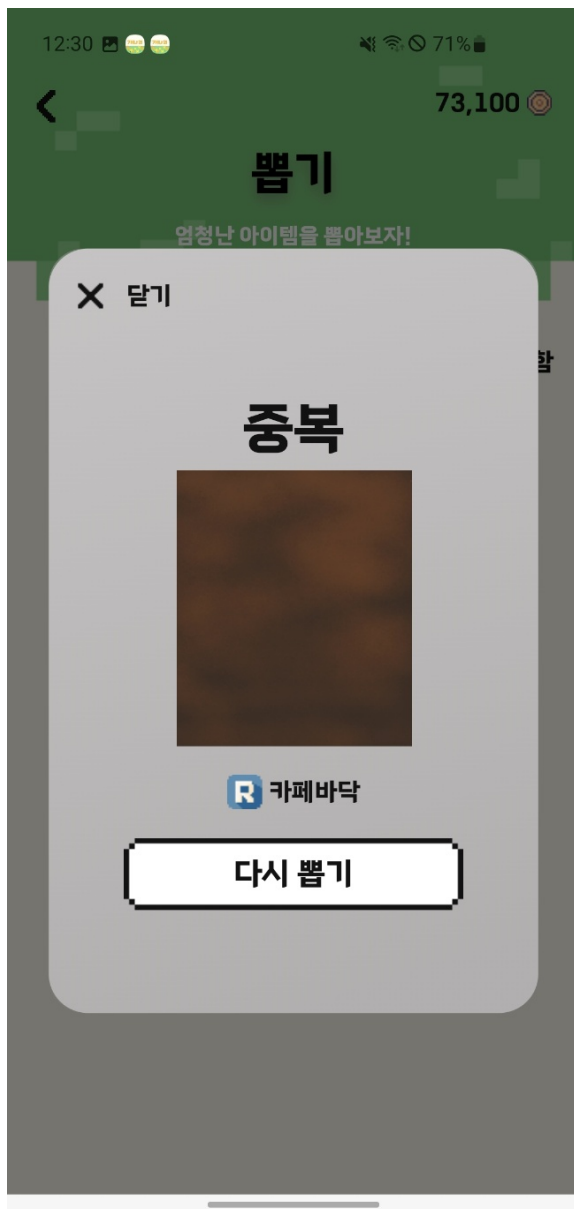


보상 화면



뽑기 화면

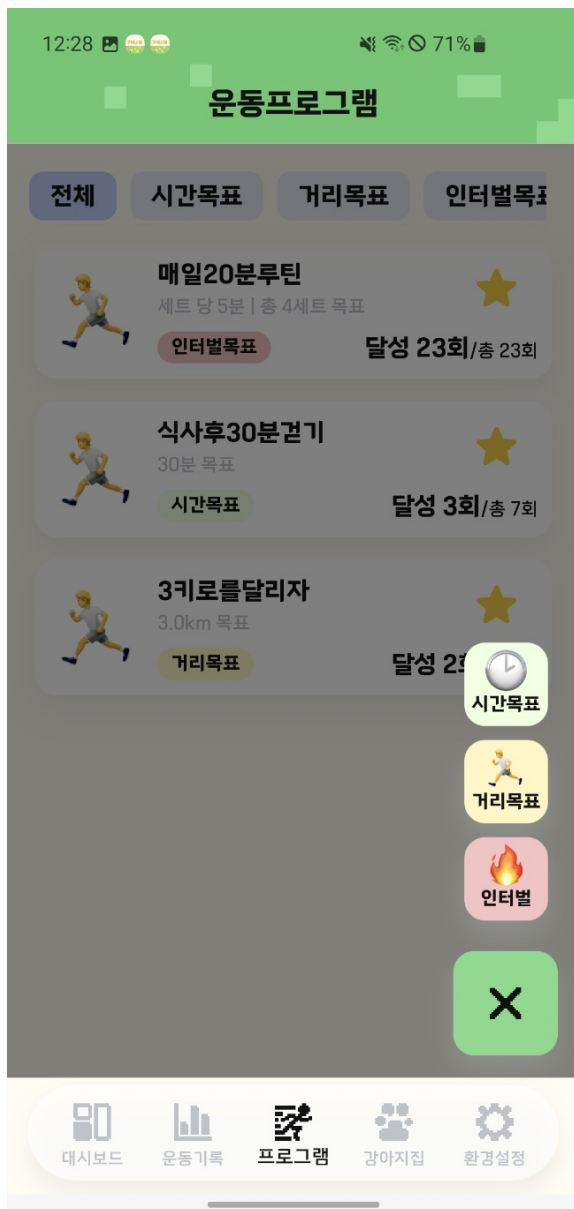




운동 기록 상세 화면

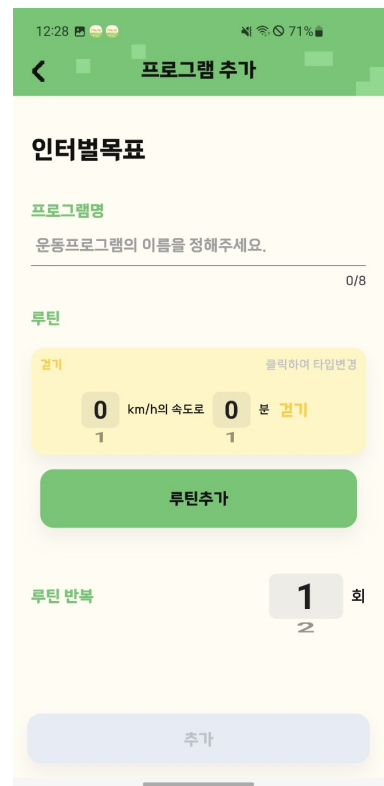
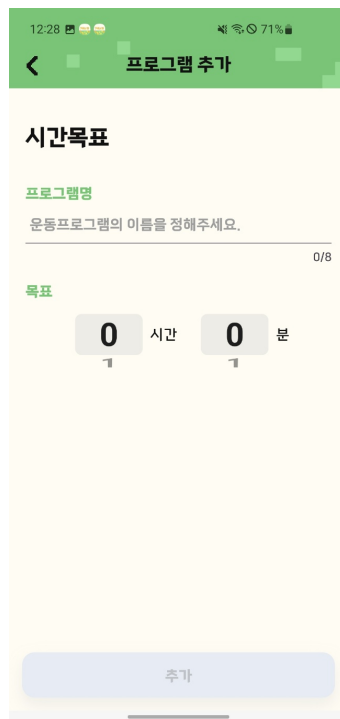
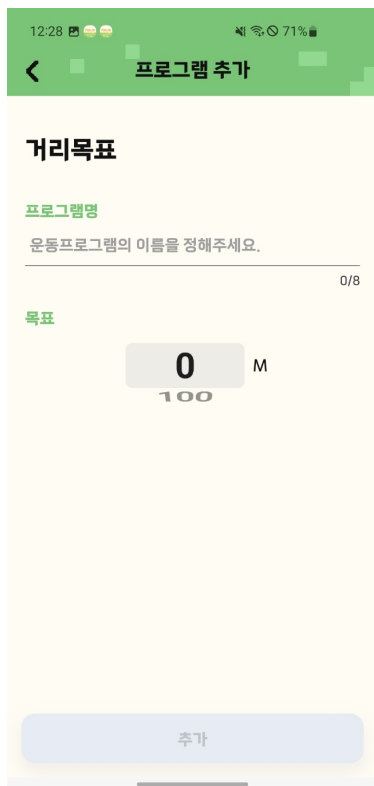


운동 프로그램 목록 화면



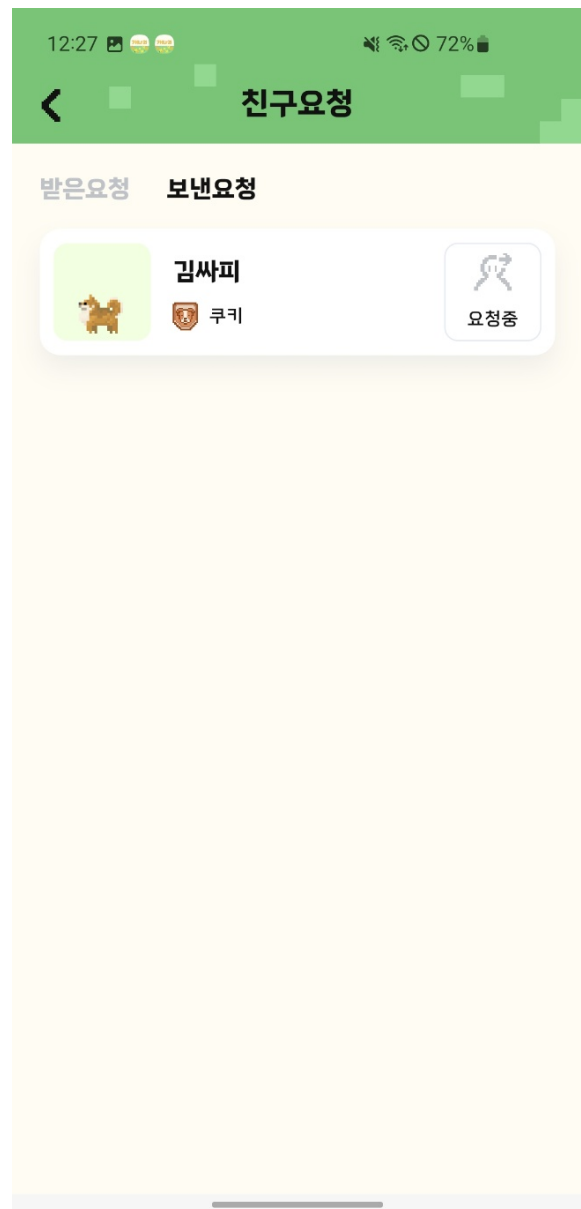
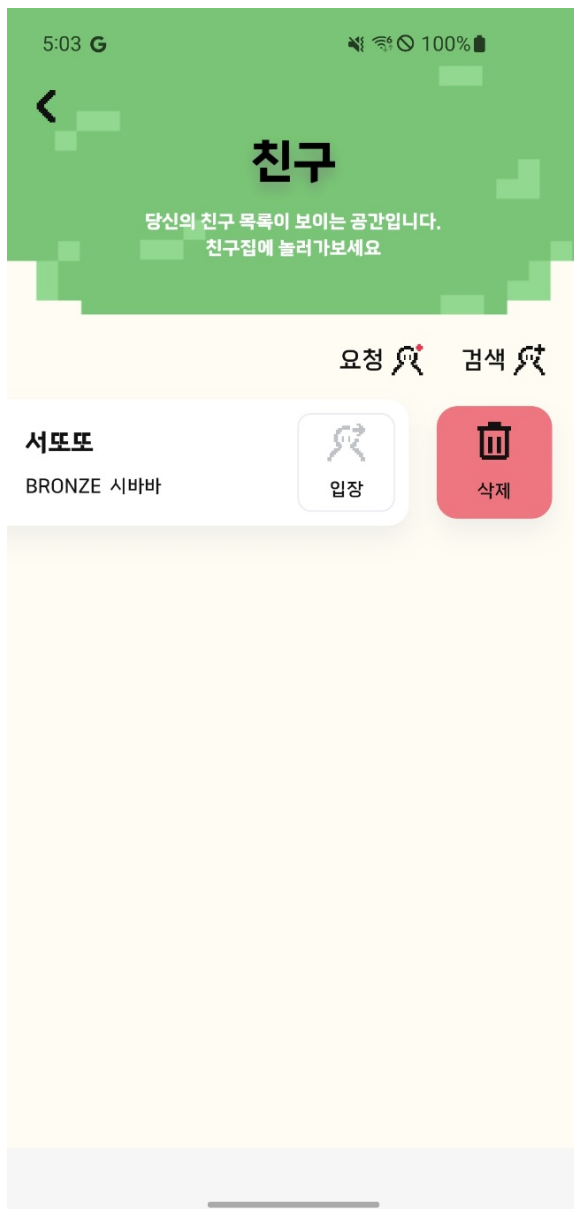


운동 프로그램 추가 화면



친구 화면





강아지 집 화면



환경 설정 화면

