

Source	내용
source1	<pre># Kubernetes 배포를 위한 조직 내부 데이터 정의 ## 배포 환경 > ### 공통 리소스 정의 - ConfigMap 이름: gift-config - API_GATEWAY_URI: http://api-gateway-svc:8080 - SERVICE_USER_URI: http://user-service-svc:8084 - SERVICE_PRODUCT_URI: http://product-service-svc:8081 - SERVICE_WISH_URI: http://wish-service-svc:8082 - SERVICE_ORDER_URI: http://order-service-svc:8083 - KAKAO_REDIRECT_URI: http://localhost:8080/members/login/oauth2/code/kakao - FRONT_DOMAIN: http://localhost:8080 - Secret 이름: gift-secret - JWT_SECRET: WW4ya2ppYmRkRkFXdG5QSjJBRmxMOFdYbW9oSk1DdmInUWdnYUV5cGE1RT0= - KAKAO_CLIENT_ID: YWMyMGQ4NGY5N2I2NzlmMGJiZDlhODA2NWJhODIzZjM= - KAKAO_CLIENT_SECRET: RnczSmIMNVBrcGV2WFZaOU9pVk04YmZiZ0hVUEQ1RUE= > ### MSA 애플리케이션 정의</pre>
source2	<pre>- KAKAO_CLIENT_SECRET: RnczSmIMNVBrcGV2WFZaOU9pVk04YmZiZ0hVUEQ1RUE= > ### MSA 애플리케이션 정의 - 서비스 설명 - 서비스 이름: 선물하기 서비스 (Gift Service) - 구성 마이크로서비스 목록: - api-gateway: 모든 MSA 요청에 대한 인증 및 라우팅을 담당하는 관문 서비스입니다. - user-service: 사용자 회원가입, 로그인, 정보 관리를 담당합니다. - product-service: 상품 정보, 재고, 옵션을 관리합니다. - wish-service: 사용자의 위시리스트를 관리합니다.</pre>
source3	<pre>- wish-service: 사용자의 위시리스트를 관리합니다. - order-service: 상품 주문 및 카카오톡 메시지 발송을 처리합니다. - gift-app: 사용자를 위한 웹 UI를 제공하는 BFF(Backend for Frontend) 서비스입니다. - 의존 서비스 - gift-app: api-gateway에 의존하여 모든 백엔드 API를 호출합니다. - api-gateway: user-service, product-service, wish-service, order-service로 요청을 라우팅합니다. - order-service: product-service (재고 차감), wish-service (주문 후 위시리스트 삭제), user-service (카카오 토큰 조회)에 의존합니다.</pre>

Source	내용
source4	<ul style="list-style-type: none"> - wish-service: product-service (상품 정보 조회)에 의존합니다. - user-service, product-service: 다른 서비스에 대한 직접적인 의존성 없이 독립적으로 실행됩니다. - 볼륨 구성 - 애플리케이션은 상태 비저장(Stateless)으로 설계되어, 별도의 영구 볼륨(Persistent Volume)을 사용하지 않습니다. 모든 데이터는 외부 데이터베이스에 저장됩니다. - 내부 서비스 포트 (Service Port) - api-gateway-svc: 8080 - user-service-svc: 8084 - product-service-svc: 8081 - wish-service-svc: 8082 - order-service-svc: 8083
source5	<ul style="list-style-type: none"> - user-service-svc: 8084 - product-service-svc: 8081 - wish-service-svc: 8082 - order-service-svc: 8083 - gift-app-svc: 8080 (외부 포트) -> 8085 (내부 컨테이너 포트) - 외부 노출 방식 - gift-app-svc: LoadBalancer 타입을 사용하여 외부 트래픽을 서비스의 8080 포트로 전달합니다. 이 트래픽은 gift-app 컨테이너의 8085 포트로 전달됩니다. - 기타 모든 서비스: ClusterIP 타입을 사용하여 클러스터 내부에서만 통신합니다. 외부 요청은 모두 api-gateway를 통해서만 내부 서비스로 전달됩니다.
source6	<pre> --- > ### 마이크로서비스 기본 정보 (api-gateway) - 컨테이너 이미지 및 버전 정보: sk124590/api-gateway:2.1 - 마이크로서비스 포트: 8080 - 환경 변수: - ConfigMap (gift-config) - SERVICE_USER_URI - SERVICE_PRODUCT_URI - SERVICE_WISH_URI - SERVICE_ORDER_URI - Secret (gift-secret) - JWT_SECRET - 리소스 설정: - requests: cpu: "100m", memory: "256Mi" - limits: cpu: "500m", memory: "512Mi" - 복제 수 (Replicas): 1 > ### 마이크로서비스 기본 정보 (user-service) - 컨테이너 이미지 및 버전 정보: sk124590/user-service:2.1 - 마이크로서비스 포트: 8084 </pre>

Source	내용
source7	<ul style="list-style-type: none"> - 컨테이너 이미지 및 버전 정보: <code>sk124590/user-service:2.1</code> - 마이크로서비스 포트: <code>8084</code> - 환경 변수: - Secret (gift-secret) - <code>JWT_SECRET</code> - <code>KAKAO_CLIENT_ID</code> - <code>KAKAO_CLIENT_SECRET</code> - ConfigMap (gift-config) - <code>KAKAO_REDIRECT_URI</code> - 리소스 설정: - requests: <code>cpu: "150m", memory: "512Mi"</code> - limits: <code>cpu: "1000m", memory: "1Gi"</code> - 복제 수 (Replicas): 1 > ### 마이크로서비스 기본 정보 (gift-app) - 컨테이너 이미지 및 버전 정보: <code>sk124590/gift-app:2.1</code> - 마이크로서비스 포트: <code>8085</code> - 환경 변수: - ConfigMap (gift-config)
source8	<ul style="list-style-type: none"> - 마이크로서비스 포트: <code>8085</code> - 환경 변수: - ConfigMap (gift-config) - <code>API_GATEWAY_URI</code> - <code>KAKAO_REDIRECT_URI</code> - Secret (gift-secret) - <code>JWT_SECRET</code> - <code>KAKAO_CLIENT_ID</code> - <code>KAKAO_CLIENT_SECRET</code> - 리소스 설정: - requests: <code>cpu: "100m", memory: "512Mi"</code> - limits: <code>cpu: "500m", memory: "1Gi"</code> - 복제 수 (Replicas): 1 > ### 마이크로서비스 기본 정보 (product-service) - 컨테이너 이미지 및 버전 정보: <code>sk124590/product-service:2.1</code> - 마이크로서비스 포트: <code>8081</code> - 환경 변수: 없음 - 리소스 설정: - requests: <code>cpu: "150m", memory: "512Mi"</code> - limits: <code>cpu: "1000m", memory: "1Gi"</code>

Source	내용
source9	<ul style="list-style-type: none"> - 환경 변수: 없음 - 리소스 설정: <ul style="list-style-type: none"> - requests: <code>cpu: "150m", memory: "512Mi"</code> - limits: <code>cpu: "1000m", memory: "1Gi"</code> - 복제 수 (Replicas): 1 > ### 마이크로서비스 기본 정보 (wish-service) - 컨테이너 이미지 및 버전 정보: <code>sk124590/wish-service:2.1</code> - 마이크로서비스 포트: <code>8082</code> - 환경 변수: <ul style="list-style-type: none"> - ConfigMap (gift-config) <ul style="list-style-type: none"> - <code>SERVICE_PRODUCT_URI</code> - 리소스 설정: <ul style="list-style-type: none"> - requests: <code>cpu: "100m", memory: "512Mi"</code> - limits: <code>cpu: "700m", memory: "1Gi"</code> - 복제 수 (Replicas): 1 > ### 마이크로서비스 기본 정보 (order-service) - 컨테이너 이미지 및 버전 정보: <code>sk124590/order-service:2.1</code>
source10	<ul style="list-style-type: none"> > ### 마이크로서비스 기본 정보 (order-service) - 컨테이너 이미지 및 버전 정보: <code>sk124590/order-service:2.1</code> - 마이크로서비스 포트: <code>8083</code> - 환경 변수: <ul style="list-style-type: none"> - ConfigMap (gift-config) <ul style="list-style-type: none"> - <code>SERVICE_PRODUCT_URI</code> - <code>SERVICE_WISH_URI</code> - <code>SERVICE_USER_URI</code> - <code>FRONT_DOMAIN</code> - Secret (gift-secret) <ul style="list-style-type: none"> - <code>KAKAO_CLIENT_ID</code> - <code>KAKAO_CLIENT_SECRET</code> - 리소스 설정: <ul style="list-style-type: none"> - requests: <code>cpu: "100m", memory: "512Mi"</code> - limits: <code>cpu: "700m", memory: "1Gi"</code> - 복제 수 (Replicas): 1 --- ## 조직 내부 정책 > ### 조직 내부 정책 - 레이블 규칙

Source	내용
source11	<ul style="list-style-type: none"> - 복제 수 (Replicas): 1 --- ## 조직 내부 정책 > ### 조직 내부 정책 - 레이블 규칙 - 모든 Kubernetes 리소스에는 다음 레이블을 필수로 포함해야 합니다. - app: [microservice-name] (e.g., app: user-service) - project: gift-service - tier: backend 또는 tier: frontend - 리소스 할당 기준 - 모든 컨테이너는 리소스 요청(requests)과 한계(limits)를 명시적으로 설정해야 합니다. - 최소 요청량: cpu: "50m", memory: "128Mi" - 최대 한계량: cpu: "1500m", memory: "2Gi"
source12	<ul style="list-style-type: none"> - 최소 요청량: cpu: "50m", memory: "128Mi" - 최대 한계량: cpu: "1500m", memory: "2Gi" - 초기 리소스 할당은 위에 정의된 값을 따르며, 운영 모니터링 데이터를 기반으로 분기별로 재조정합니다. - 보안 및 운영 정책 - 이미지 태그: 안정적인 배포를 위해 latest 태그 사용을 금지하고, 반드시 명시적인 버전 태그 (e.g., 2.1)를 사용해야 합니다. - Secret 관리: - ConfigMap: 일반적인 구성 변수, 서비스 URI 등 비민감성 데이터를 저장합니다. - Secret:
source13	<ul style="list-style-type: none"> - ConfigMap: 일반적인 구성 변수, 서비스 URI 등 비민감성 데이터를 저장합니다. - Secret: - API 키, DB 접속 정보, JWT 시크릿 키 등 민감 데이터를 저장합니다. - 모든 Secret 값은 사전에 Base64 인코딩해서 기입해야 하며, data 필드에 저장해야 합니다. - stringData 사용은 금지합니다.