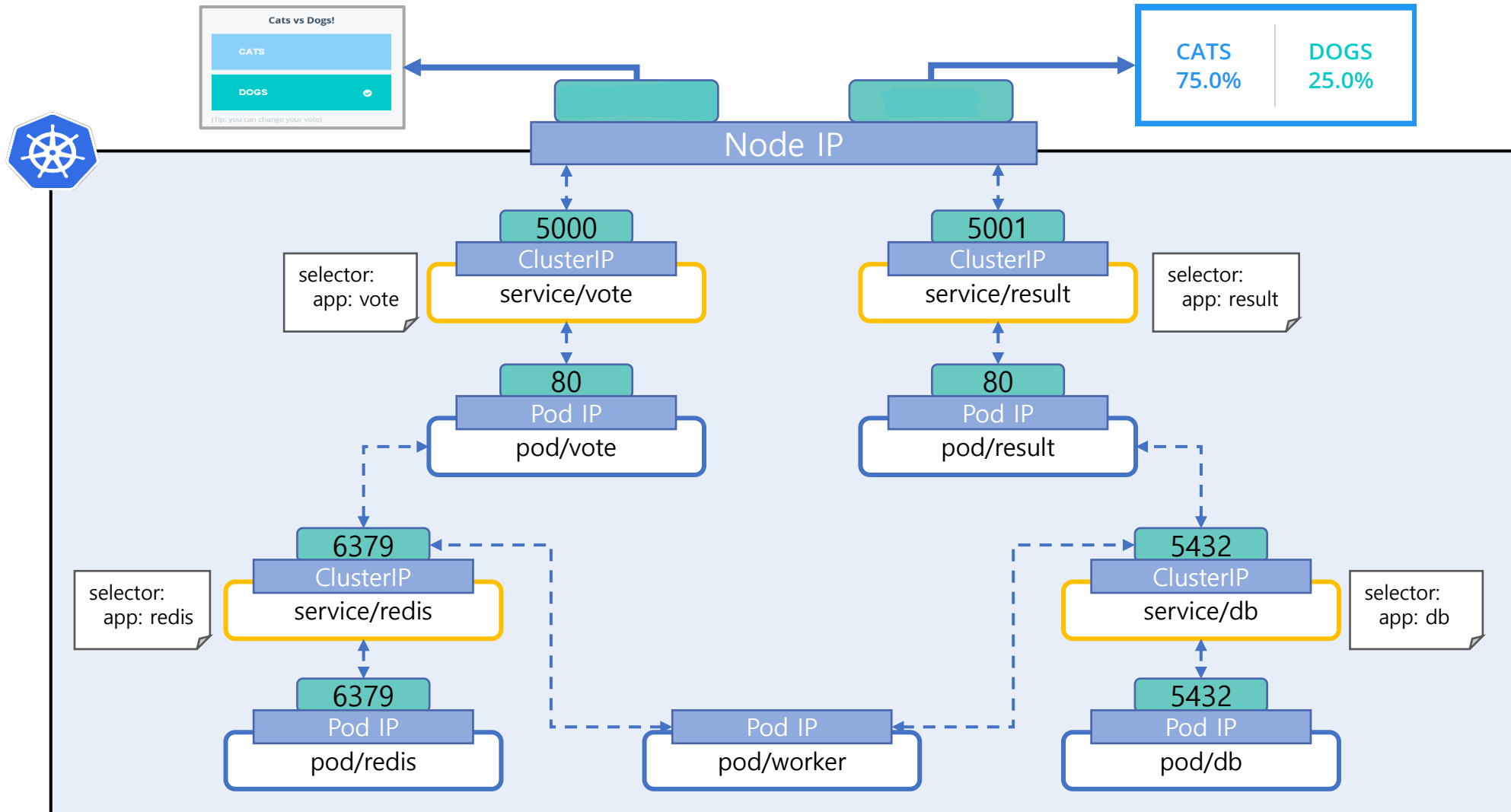


Micro Service Architecture 실습

Voting app example

전체 구조 - voting app example



실습

실습 시연내용을 보시고 각자 해당 내용을 실습 하십시오

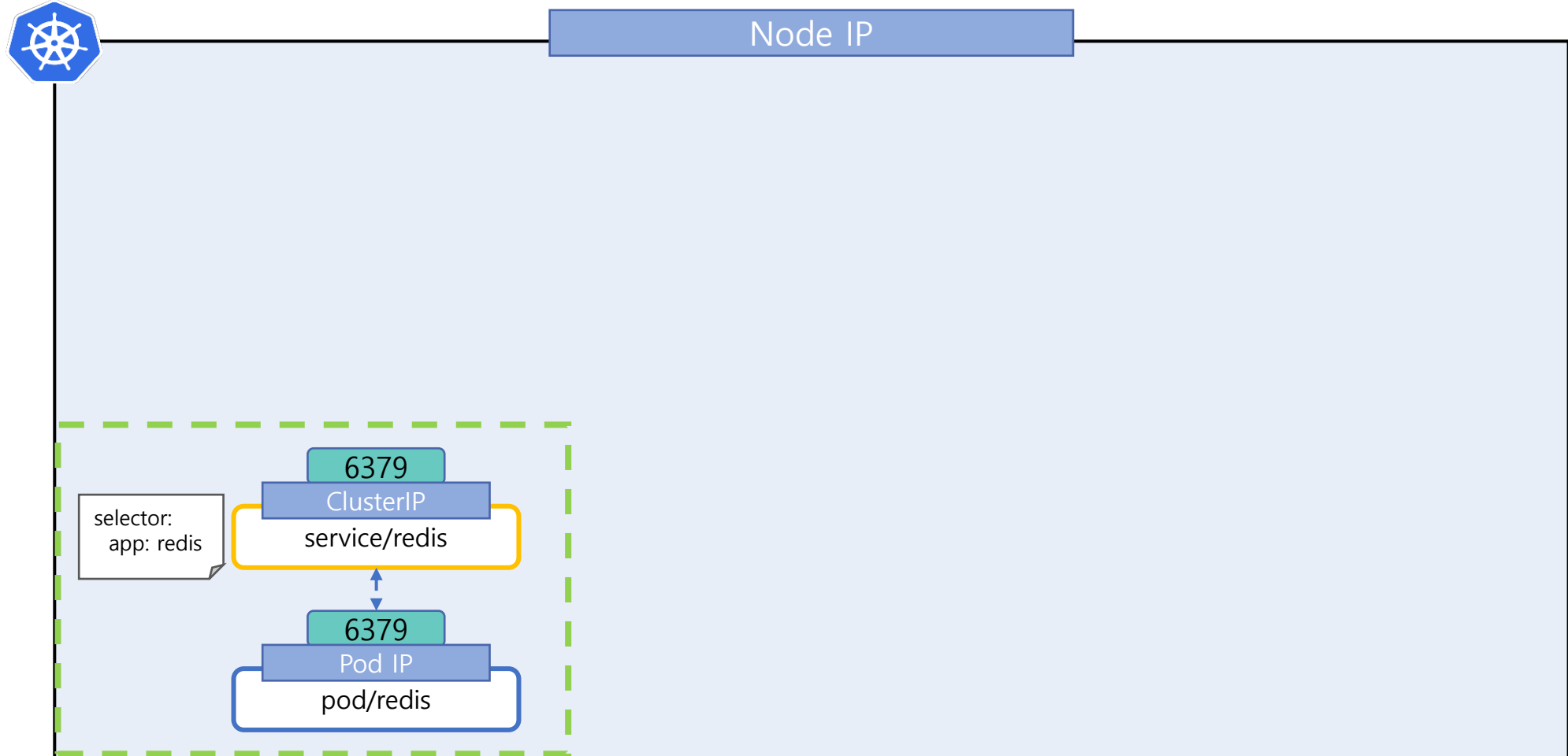
MSA 실습 환경 준비

- **git clone https://github.com/oneJang/k8s-voting-app.git**
 - yaml 실습 파일 다운로드
- **cd k8s-voting-app/k8s-specifications/**
 - 실습 파일 디렉토리 이동

redis Pod 배포 및 ClusterIP 서비스 생성

- 개인의 투표 결과를 queue 형식으로 redis에 데이터 저장
- **cat redis-deployment.yaml** → yaml 파일 내용 확인하기
- **kubectl apply -f redis-deploymet.yaml**
 - redis Pod를 deployment 형식으로 배포
- **cat redis-service.yaml**
- **kubectl apply -f redis-service.yaml**
 - redis service를 ClusterIP 형식으로 서비스 노출
 - 클러스터 내부에서 redis Pod로 접근 가능
- **kubectl get pods**
 - Pod 배포 상태 확인
- **kubectl get service**
 - service 생성 여부 확인

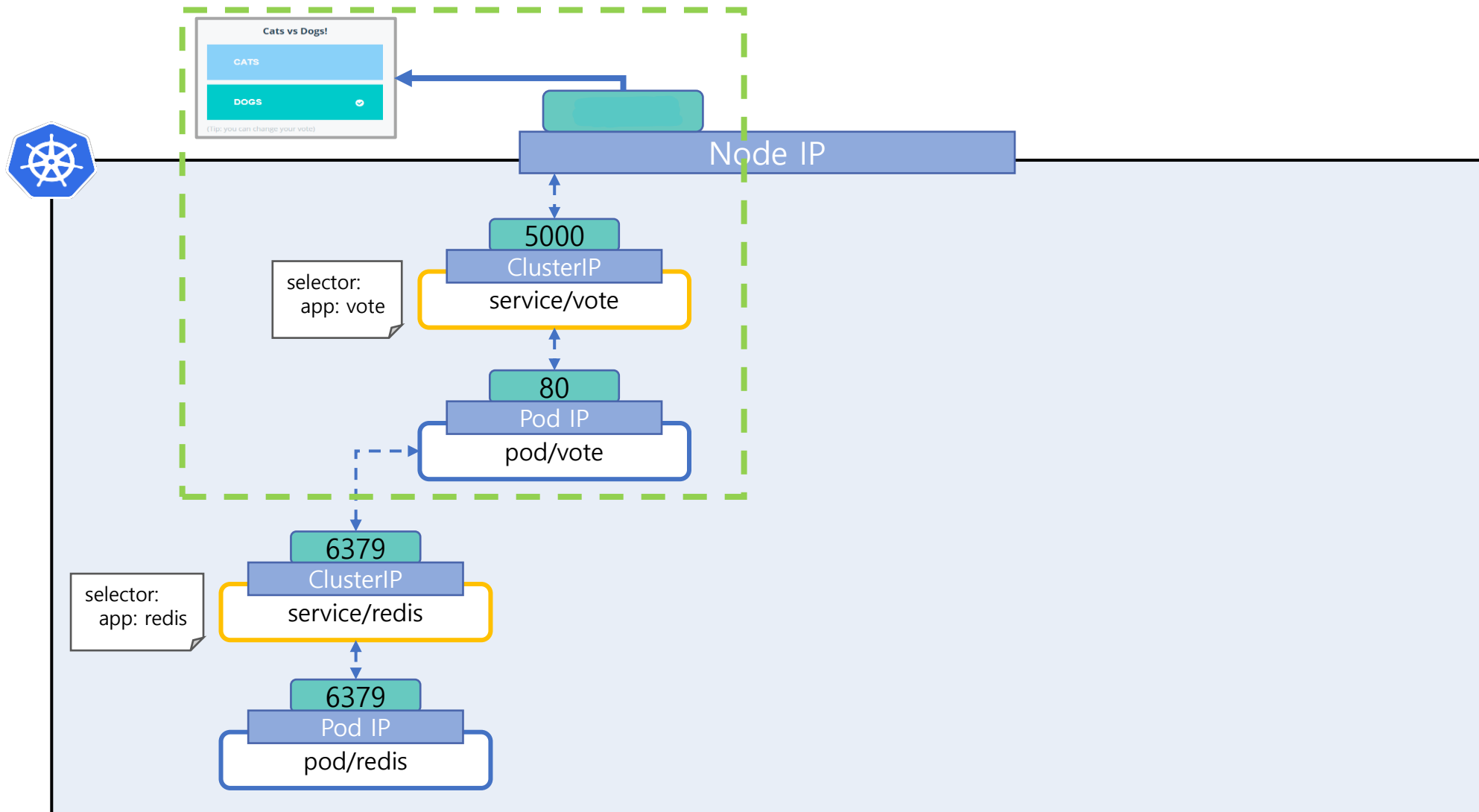
redis Pod 배포 및 ClusterIP 서비스 생성



vote Pod 배포 및 NodePort 서비스 생성

- Python Flask를 이용하여 만든 투표 대시보드에서 투표 수집
- **cat vote-deployment.yaml**
- **kubectl apply -f vote-deploymet.yaml**
 - vote Pod를 deployment 형식으로 배포
- **cat vote-service.yaml**
- **kubectl apply -f vote-service.yaml**
 - vote service를 NodePort 형식으로 서비스 노출
 - 외부에서 특정 port #로 접근 가능
- **kubectl get pods**
 - Pod 배포 상태 확인
- **kubectl get service**
 - service 생성 여부 확인 및 NodePort(31000) 정보 확인 (ex. 5000:31000/TCP)

vote Pod 배포 및 NodePort 서비스 생성



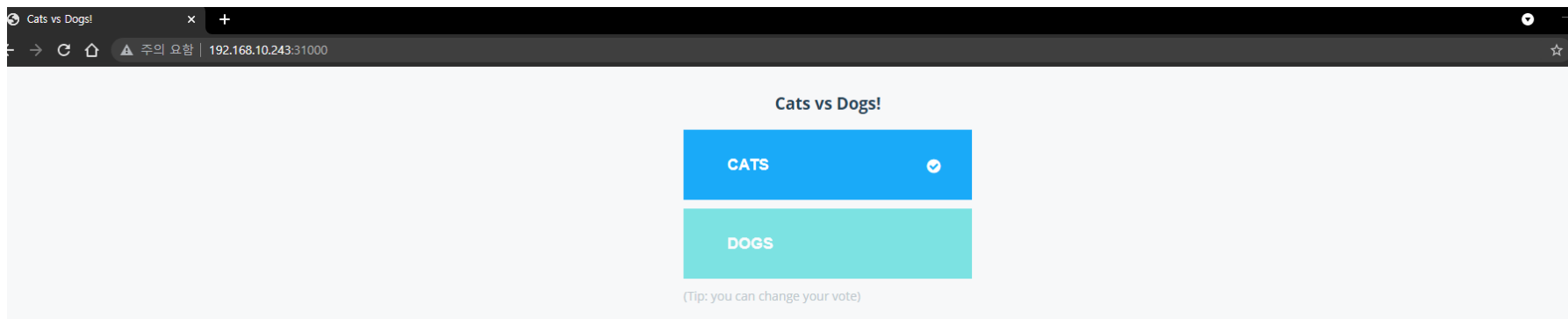
투표 대시보드 접속

- **ifconfig**

- 현재 노드의 ip 주소 확인 (ex. 192.168.x.x)

- 브라우저(크롬) URL 주소 창에 <NODE_IP>: <NodePort> 입력 후 접속 확인

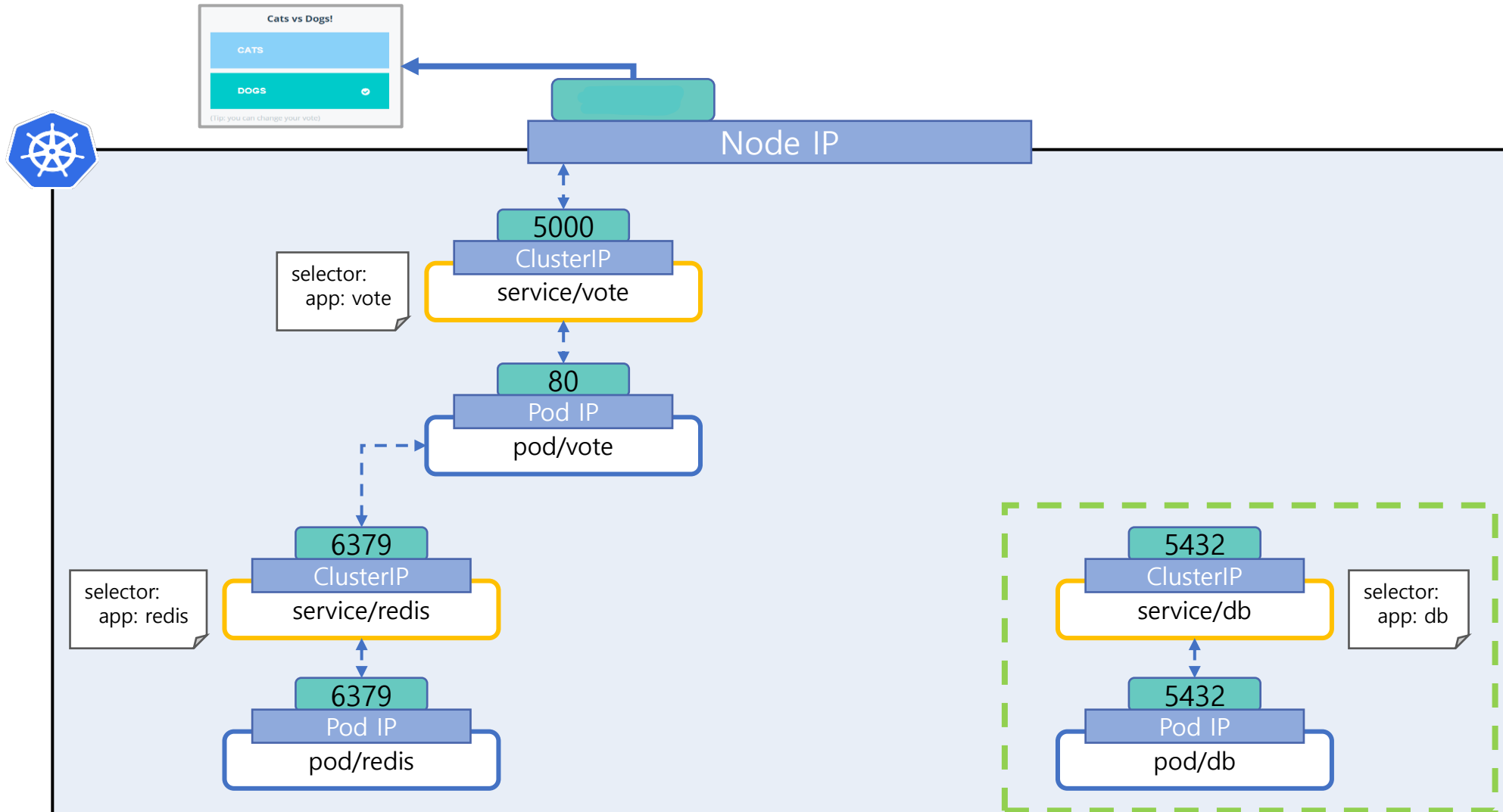
- cats 또는 dogs 중 한 개 클릭 !



db Pod 배포 및 ClusterIP 서비스 생성

- 투표자와 투표내용을 db(PostgreSQL)에 저장
- **cat db-deployment.yaml**
- **kubectl apply -f db-deployment.yaml**
- **cat db-service.yaml**
- **kubectl apply -f db-service.yaml**
- **kubectl get pods**
- **kubectl get service**

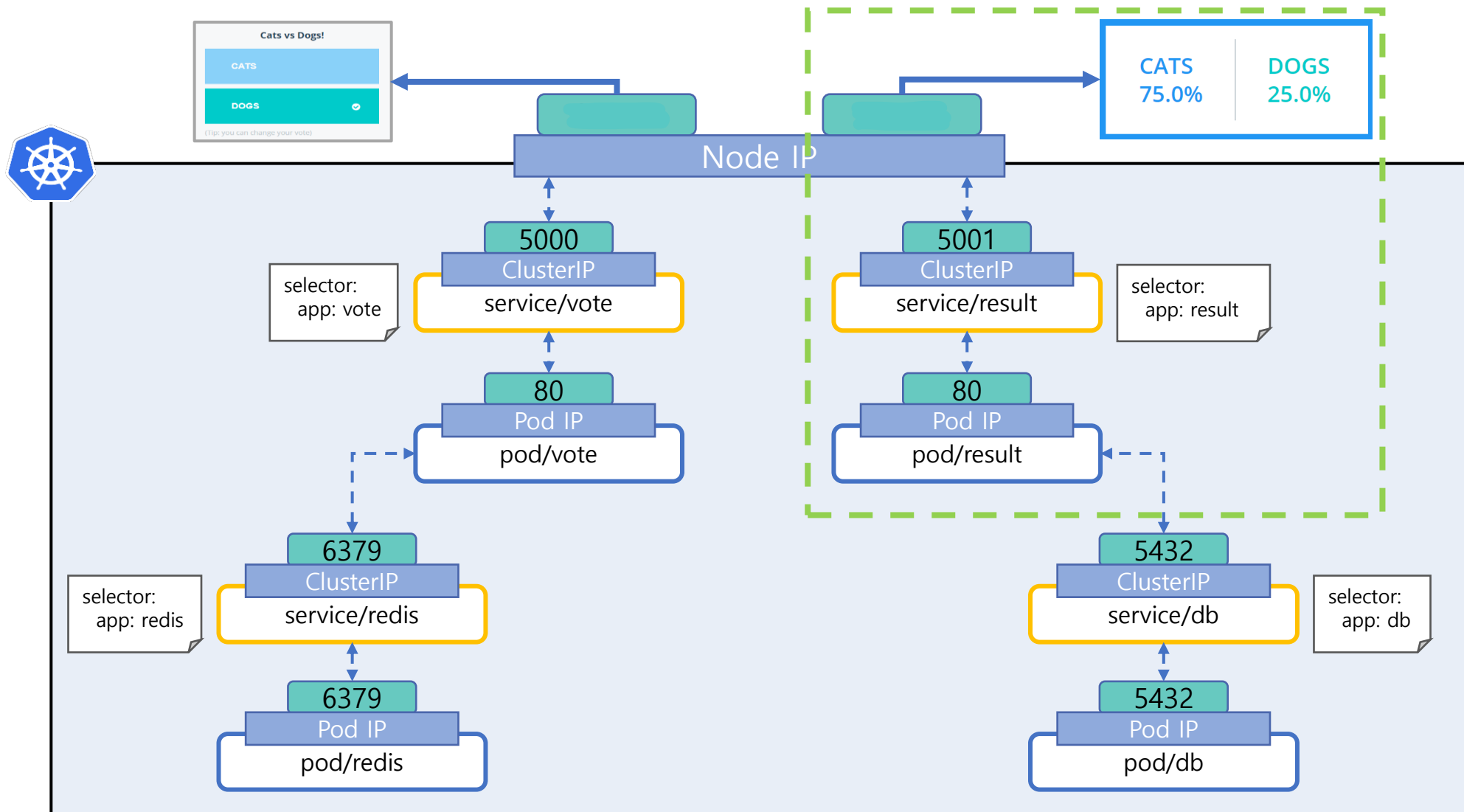
db Pod 배포 및 ClusterIP 서비스 생성



result Pod 배포 및 NodePort 서비스 생성

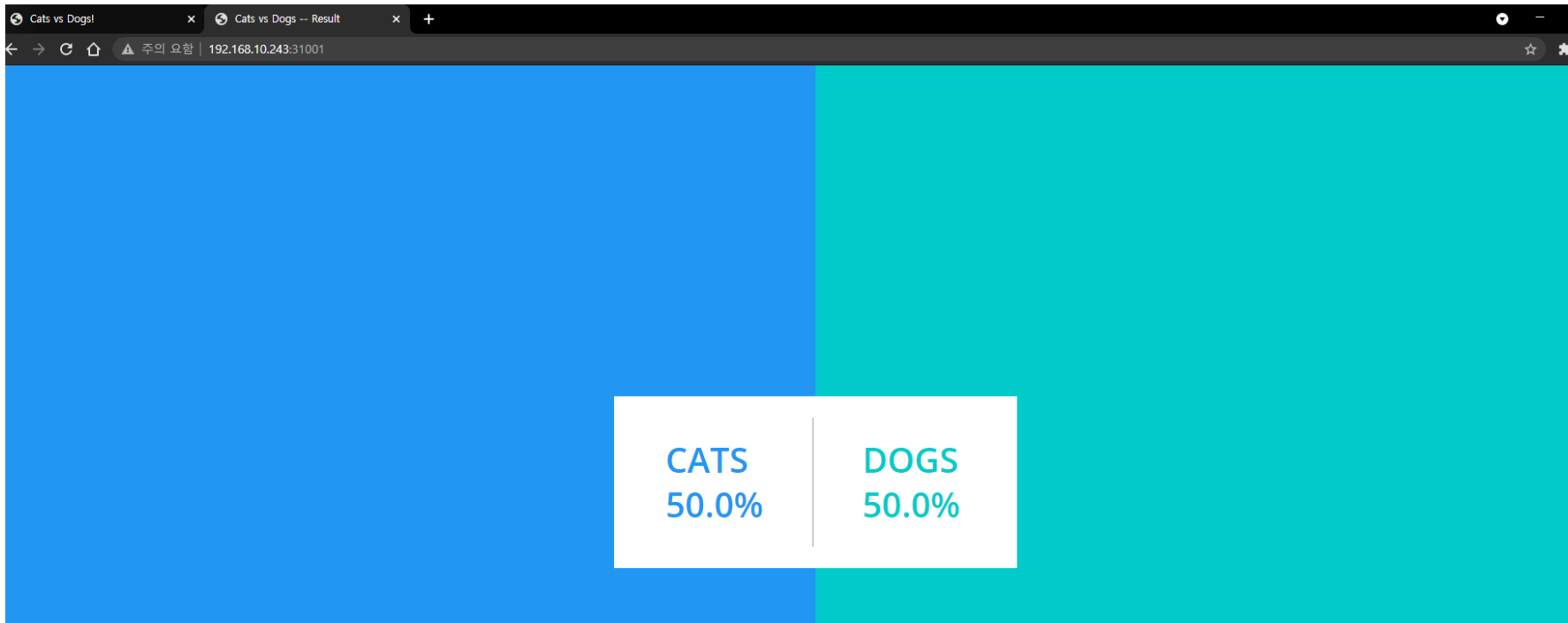
- 투표 결과를 집계하고 자바스크립트를 이용하여 만든 투표 결과 대시보드 제공
- **cat result-deployment.yaml**
- **kubectl apply -f result-deployment.yaml**
- **cat result-service.yaml**
- **kubectl apply -f result-service.yaml**
- **kubectl get pods**
- **kubectl get service**
 - NodePort(31001) 정보 확인 (ex. 5001:31001/TCP)

result Pod 배포 및 NodePort 서비스 생성



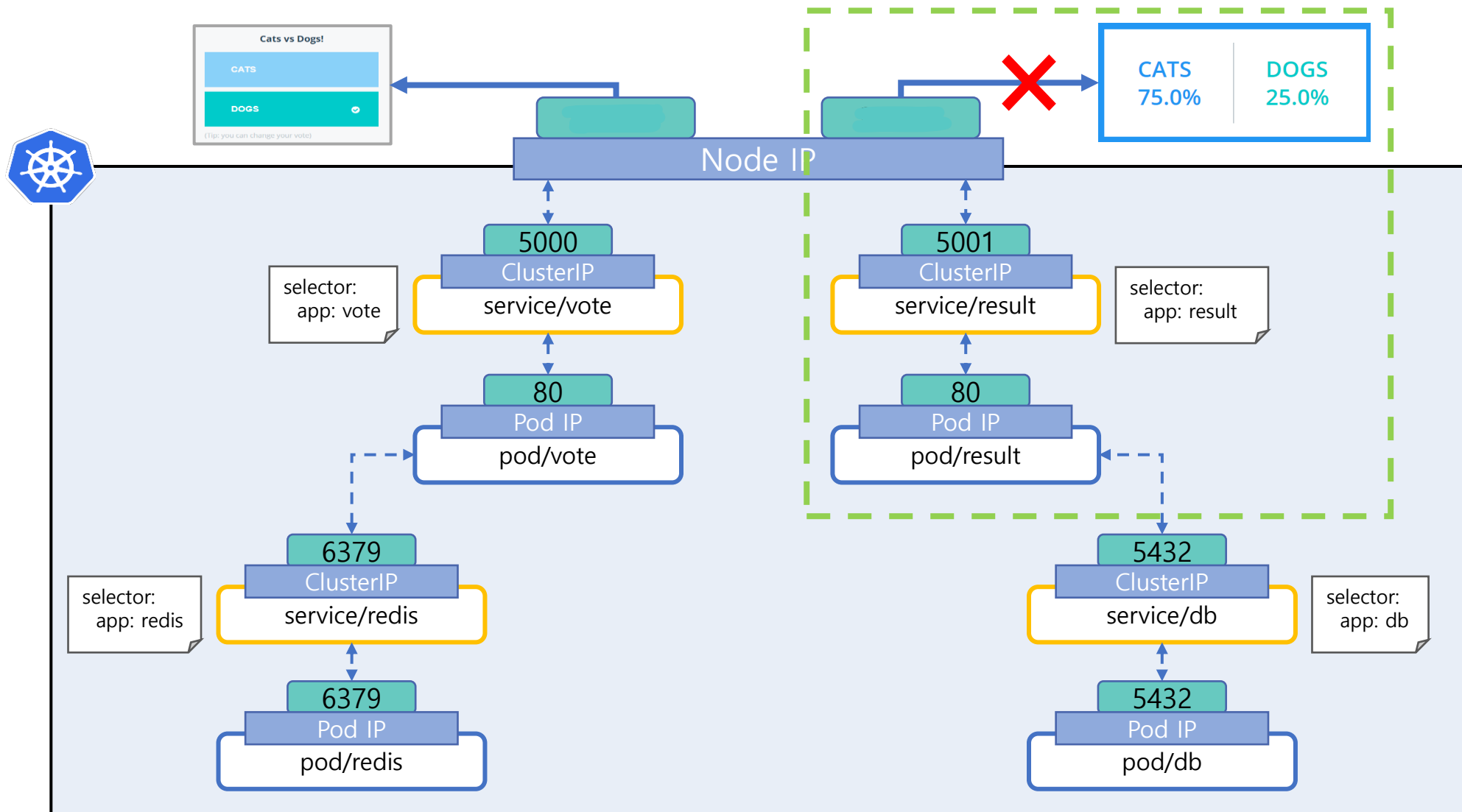
투표 결과 대시보드 접속

- 브라우저(크롬) URL 주소 창에 <NODE_IP>: <NodePort> 입력 후 결과 확인



- cats 또는 dogs 중 한 개를 선택했지만 투표 결과는 50% / 50%로 표시됨
 - redis → PostgreSQL 로 값을 넘기는 컴포넌트 필요!
 - 전체 구조 다시 확인 !

투표 결과 대시보드 접속

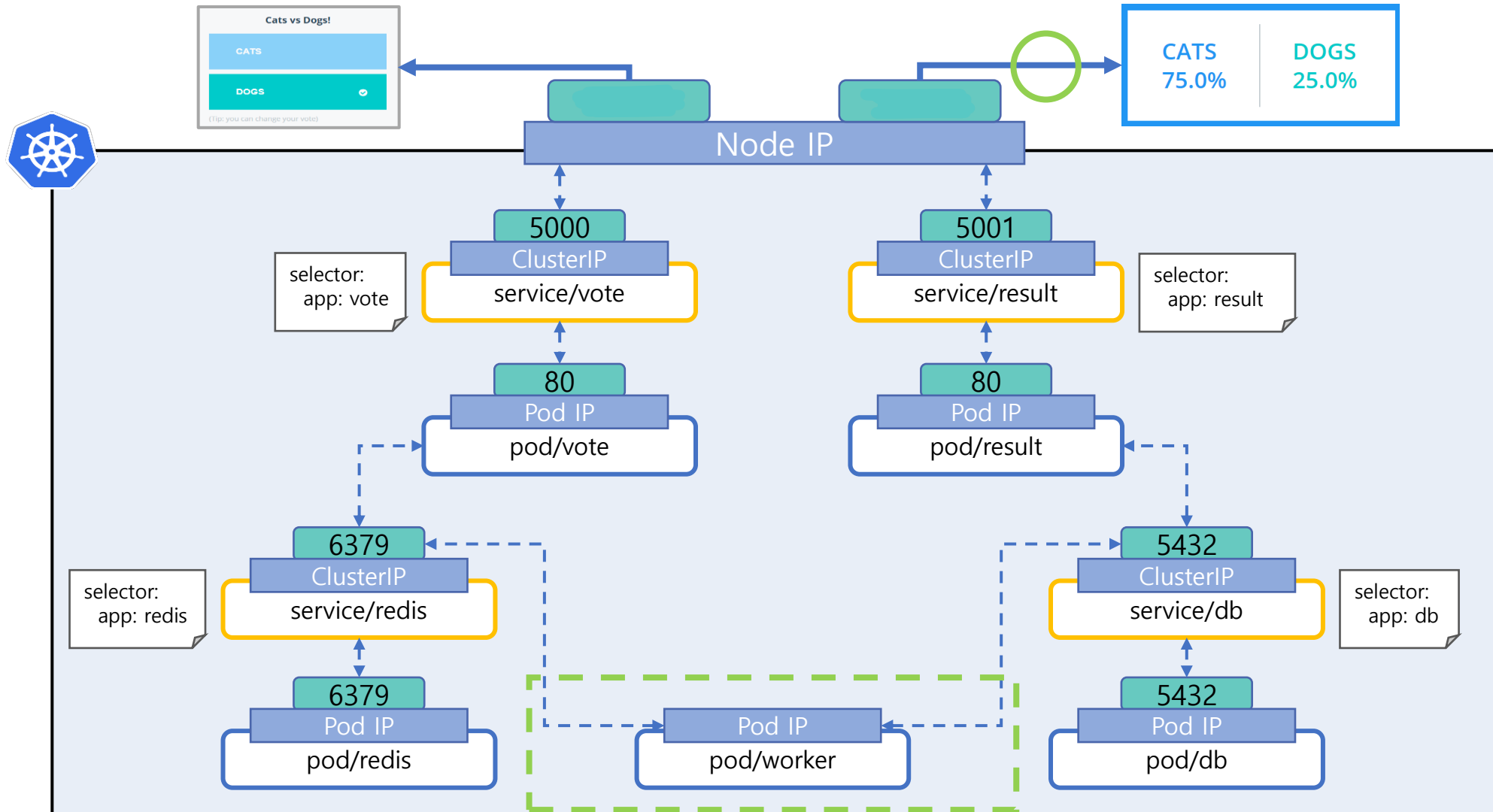


worker Pod 배포

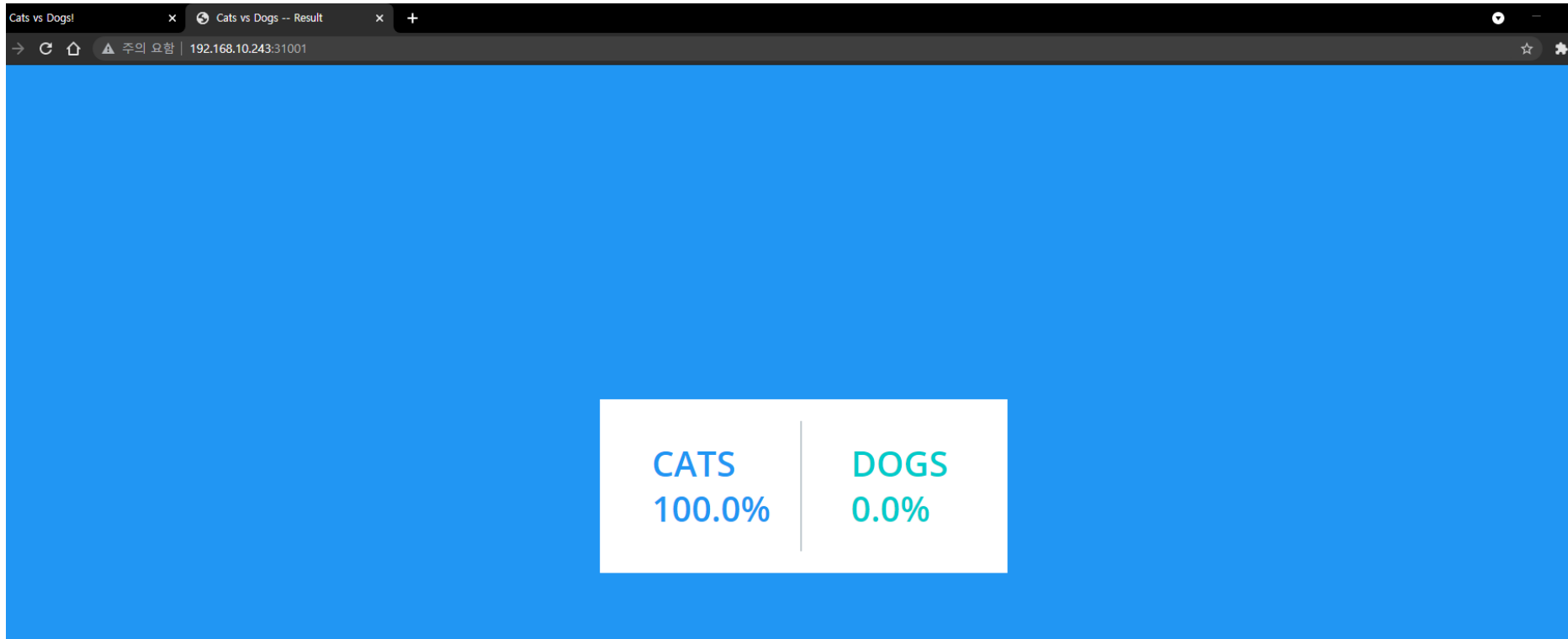
- redis에서 queue 형식으로 저장된 데이터를 받아와 {voter_id/vote} 형식의 고유 voter id로 데이터를 구분하여(.NET) db(PostgreSQL)에 저장
- **cat worker-deployment.yaml**
- **kubectl apply -f worker-deploymet.yaml**
- **kubectl get pods**
- **kubectl logs <WORKER_POD_ID>**

```
Connected to db
Found redis at 10.111.24.164
Connecting to redis
Processing vote for 'a' by '78f88fada70ae47d'
```


worker Pod 배포



투표 결과 대시보드 재접속



- cats로 선택했을시 투표 결과는 100% / 0%으로 표시
 - 투표 내용 정상 반영 확인