

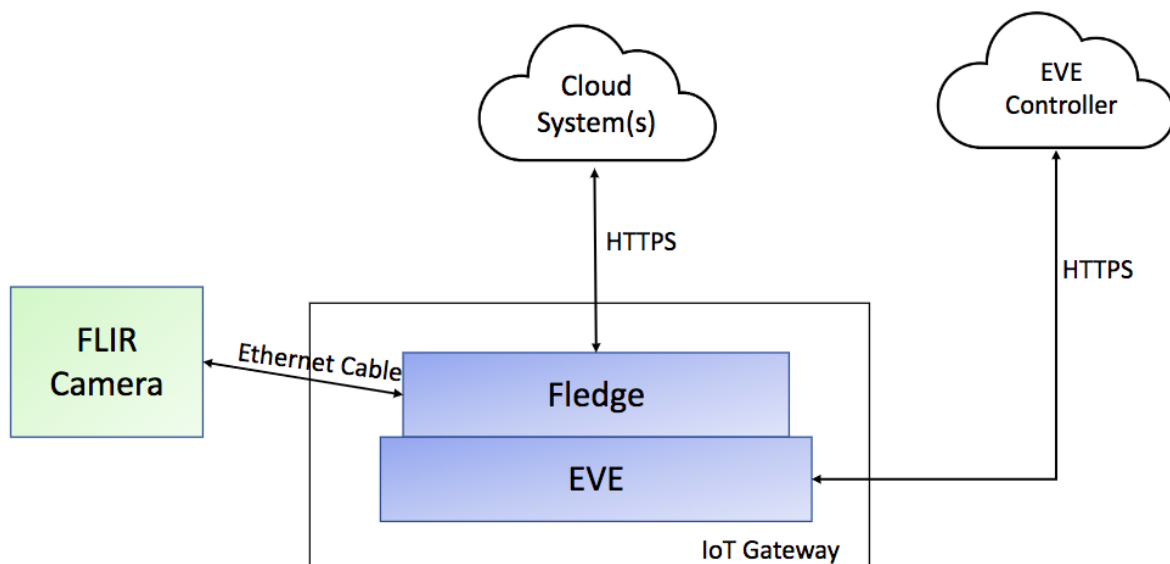
# IIoT Predictive Maintenance Installation Guide

You can just run the test from <https://github.com/lf-edge/eden.git>

The test is available through /tests/flir

<https://github.com/lf-edge/eden/tree/master/tests/flir>

## The components



Set up an infrastructure:

The setup is done on a normal PC or in the cloud. Anywhere where it's convenient for a DevOps person. Then a remote setup/deployment is done.

If it's a standalone device you will prepare an image to burn on SD card.

## Stage 0 Prepare FLIR camera (remote).

The camera should be available via some IP

## Stage 1 Prepare Infrastructure.

Clone Eden Project:

```
git clone https://github.com/lf-edge/eden.git
```

Prepare Eve to be deployed on GCP. You may prepare to deploy it in the cloud or on Edge device (e.g. Raspberry). Then you need to burn the image on SD card

```
make clean
```

```
make build
```

```
./eden config add --devmodel GCP
./eden config set default --key adam.eve-ip --value <The public IP of the machine>
./eden setup
```

Then use the google service key to deploy to GCP.

## Stage 2 Deploy infrastructure (e.g. on GCP)

```
#../google.json is a google service key file
# /home/hello_fleandr/eden/dist/images/eve/live.img is what is generated by setup
./eden utils gcp image -k ../google.json upload /home/hello_fleandr/eden/dist/images/eve/live.img
./eden utils gcp vm -k ../google.json run
```

```
BWD=$(./eden utils gcp vm get-ip -k ../google.json)
```

```
#make sure you allowed gcp firewall
./eden utils gcp firewall -k ../google.json --source-range $BWD --name some$BUILD_NUMBER
```

Start Eden and onboard eve whenever you deployed it.

```
./eden start
./eden eve onboard
```

## Stage 3 Deploy Fledge container

#now run prebuilt Flir container

```
./eden pod deploy --publish=8027:80 --publish=8028:8081 docker://itmoeve/fledgeeveflirdemo --cpus=2 --memory=2GB
```

## Stage 4 Setup Fledge container



Configuration of Fledgev2.docx