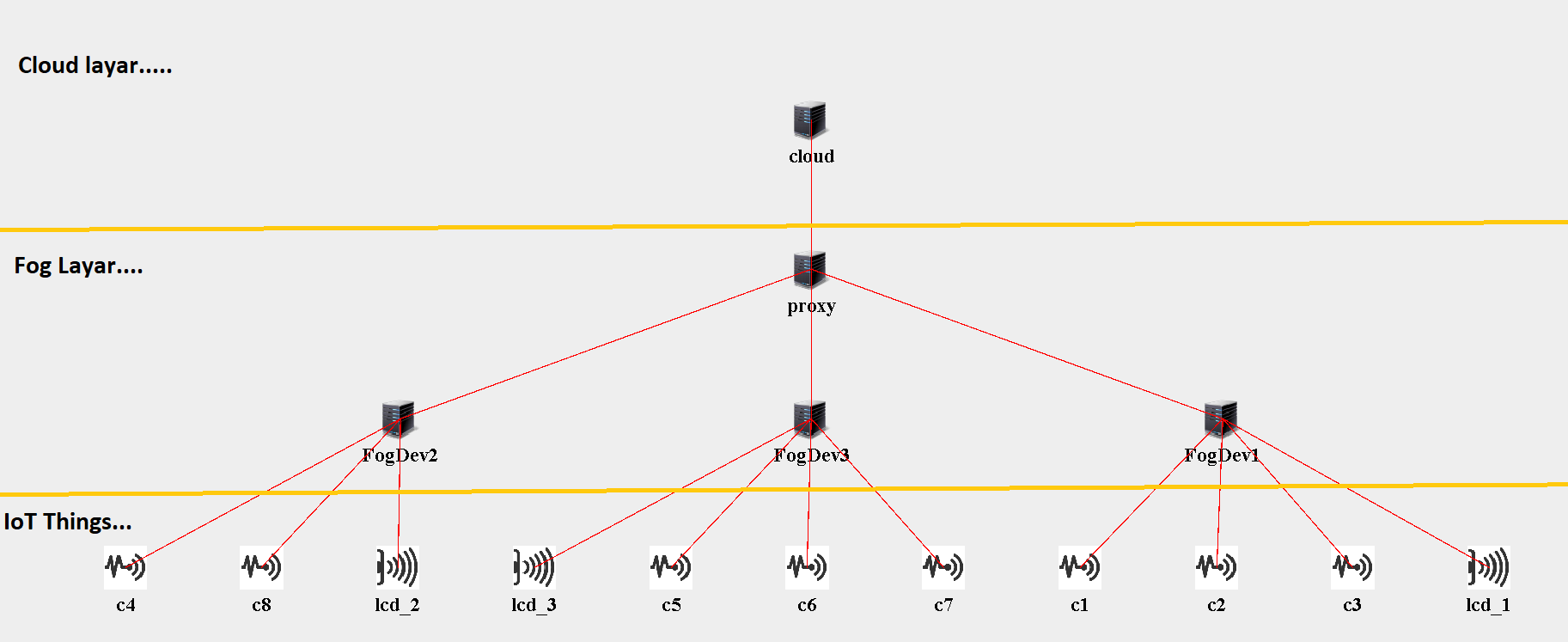
SMART VEHICLE PARKING SYSTEM

# Explanation with topology:

Camara and LCD is fixed at each area, there are 3 areas, connected to cloud via proxy. Camara will capture the photo of slots, whether the slot is empty or not is found by system and status will be update on LCD display, again the free or full status will be stored on cloud data center for future use.

Topology of the system:



# AppModules, AppEdges and Tuples are shown in diagram below.

DISPLAY

CAMARA

# Job Scheduling Algorithm:

Facts:

* Things(sensors) generate jobs, fog nodes also generate jobs, jobs are of two kinds periodic or event based.
* Jobs are scheduled at induvial Fog Nodes for beater QoS that are near to sensor (Things) that create job.
* We use FCFS algorithm to schedule jobs at each fog node and will observe the QoS (in terms of delay and Network usage)

Steps:

1. Add all jobs (arrives from senser/created by fog node) arrived at **i-th** interval to **JOBS\_LIST** based on arrival time.
2. Pick the first **JOB** from **JOBS\_LIST** for processing and remove it from to JOBS\_LIST.
3. Process the selected **JOB**, keep account of QoS parameter **Qi** for that **JOB**.
4. Repeat step 1,2 and 3 for i = 0 to N intervals that will complete the simulation time.
5. Print the QoS parameters (average delay, network usage)

# Problem formulation for Job scheduling to improve QoS for SMART VEHICLE PARKING SYSTEM - using iFogSim simulator

Total IoT-Fog system (Simulation) time **T** is divided into **N** equal duration intervals denoted as **It (I1, I2, I3…. In).**

We assume fixed number of fog devises and denote the set of fog devises by **F.**

The workloads are in the form of jobs, all new jobs created at the interval **It** are denoted as **Nt**.

Set all active jobs being denoted as **At**

If job **j ∈ Nt** cannot be allocated to a fog devise then it is added to a wait queue **Wt**

All created jobs that are not active and are not in the wait queue are considered to be completed

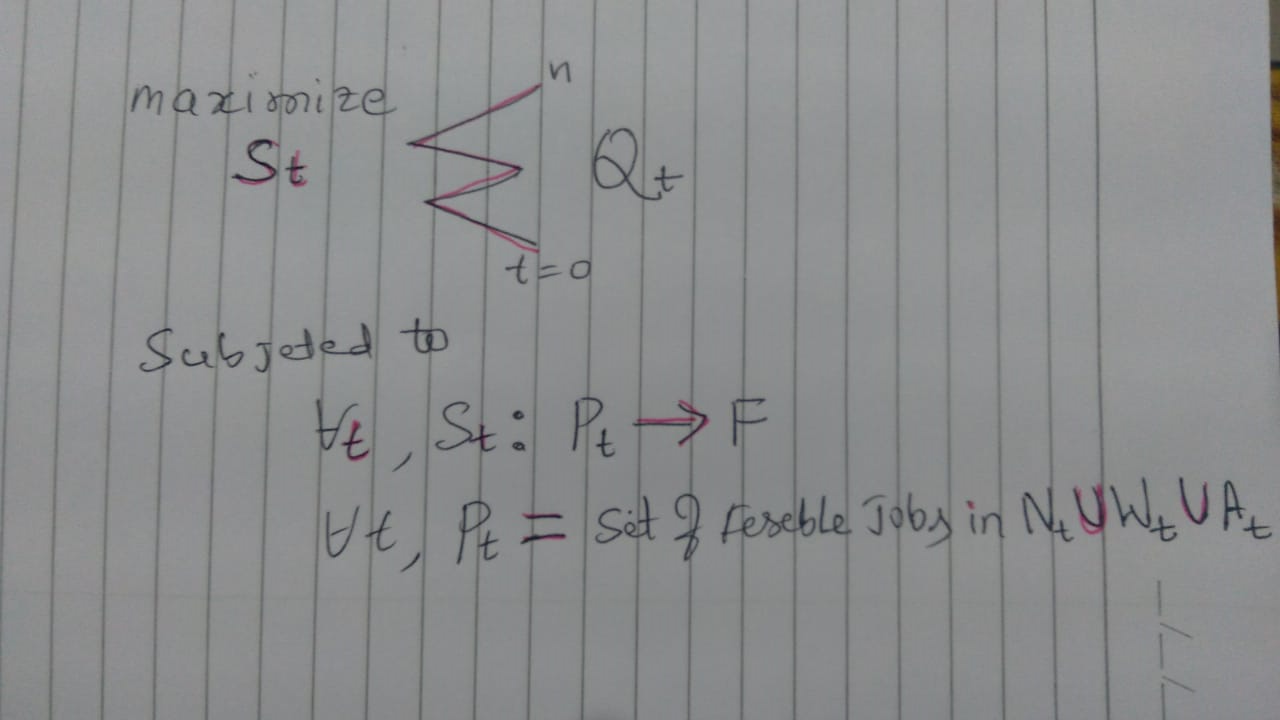
We consider the problem of maximizing the **QoS** objective score accumulated across all intervals **It** **(I1, I2, I3…. In).** in a bounded time, experiment.

We denote the QoS score for interval (**It**) by **Qt** and consider a total **N** interval in an experiment.

We denote the utilization metrics of all fog devises in interval **It−1** as **Ut**. Now using **Ut**, we need to predict a scheduling decision **St**.

All tasks for jobs in **Nt ∪ Wt ∪ At** are called feasible tasks.

Thus, the problem can be formulated as:



Results:

