iSpam Notification Control -lähdekoodi ja asetukset

# Sendmail\_openclose

var lidEvent = msg.payload.open;

var lidStatus = context.get('lidStatus') || false;

var sendNotification = false;

var sendWarning = context.get('sendWarning') || false;

var mailNotificationSent = context.get('mailNotificationSent') || false;

var currentTime = Date.now();

var limitTime = context.get('limitTime') || 0;

//a five minute limit time for lid forgotten open

const forgotLimit = 10000;

// a 10 second limit for accidental closings of lid when inserting or extracting mail

const lowerLimit = 10000000;

if (lidEvent == 1 && lidStatus === false){

lidStatus = true;

limitTime = currentTime + forgotLimit;

context.set('limitTime', limitTime);

context.set('lidStatus', lidStatus);

sendNotification = true;

}

if (lidEvent == 1 && lidStatus === true){

limitTime = context.get('limitTime')

if (limitTime - currentTime < 0){

sendWarning = true;

limitTime = currentTime + forgotLimit;

context.set('limitTime',limitTime);

}

}

if (lidEvent == 0 && lidStatus === true) {

lidStatus = false;

limitTime = 0;

context.set('limitTime', limitTime);

context.set('lidStatus', lidStatus);

}

msg = {

payload: {

"sendNotification":sendNotification,

"sendWarning":sendWarning,

"mkrOpen":msg.payload.open,

"mkrLuminere":msg.payload.lumiere,

"mkrRSSI":msg.payload.rssi

}

}

return msg;

# CmdTrigger

var sendNotification = context.get('sendNotification') || false;

var sendWarning = context.get('sendWarning') || false;

if (sendNotification !== msg.payload.sendNotification){

msg = {payload:{"sendNotification":true}}

return [msg, null];

}

if (sendWarning !== msg.payload.sendWarning){

msg = {payload:{"sendWarning":true}}

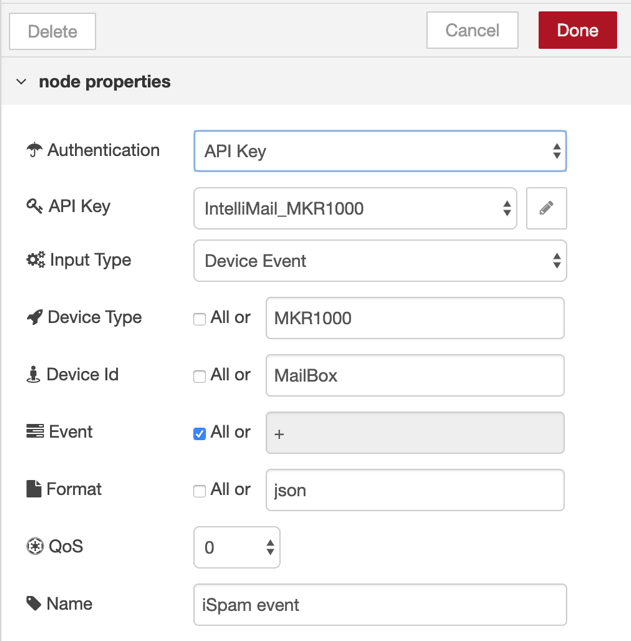
return [null, msg]

} else {

return [null, null]

}

# iSpam-event



# iSpam sendnotification

# 

# ispam sendwarning

