var keyboardObj = require('./keyboard.js');

var message = require('./message');

var flag = 0;

var MongoClient = require('mongodb').MongoClient;

var url = 'mongodb://localhost/User\_DB';

var net =require('net');

var socket = net.connect({port : 5000, host:'122.36.48.183'});

socket.on('connect', function(){

socket.write('client connect!');

});

exports.returnText = function(req, res) {

var v\_type = req.body.type;

var v\_user\_key = req.body.user\_key;

var v\_content = req.body.content;

console.log("\*\*\* message.returnText =====================================");

console.log("\*\*\* type : " + v\_type);

console.log("\*\*\* user\_key : " + v\_user\_key);

console.log("\*\*\* content : " + v\_content);

if(flag == 0 && v\_content == "가스 밸브 제어"){

keyboardObj.gas\_control(req,res);

flag++;

}

if(v\_content == "창문 제어" && flag == 0){

keyboardObj.window\_control(req,res);

flag++;

}

if(v\_content == "전등 제어" && flag == 0){

keyboardObj.lamp\_control(req,res);;

flag++;

}

if(v\_content == "밸브 열기" && flag == 1){

;

message.GasValve\_Open(req, res);

socket.write('vo');

MongoClient.connect(url, function(error, database){

if(error){

console.log(error);

}else{

db.collection('Users').update({ user\_key : v\_user\_key},{ $set: { gas\_status: 1}});

}

});

}

if(v\_content == "밸브 닫기" && flag == 1){

message.GasValve\_Close(req, res);

socket.write('vc');

MongoClient.connect(url, function(error, database){

if(error){

console.log(error);

}else{

db.collection('Users').update({ user\_key : v\_user\_key},{ $set: { gas\_status: 0}});

}

});

}

if(v\_content == "창문 잠금 켜기" && flag == 1){

message.Window\_Lock(req, res);

socket.write('wo');

MongoClient.connect(url, function(error, database){

if(error){

console.log(error);

}else{

db.collection('Users').update({ user\_key : v\_user\_key},{ $set: { window\_status: 1}});

}

});

}

if(v\_content == "창문 잠금 끄기" && flag == 1){

message.Window\_Unlock(req, res);

socket.write('wc');

MongoClient.connect(url, function(error, database){

if(error){

console.log(error);

}else{

db.collection('Users').update({ user\_key : v\_user\_key},{ $set: { window\_status: 0}});

}

});

}

if(v\_content == "전등 켜기" && flag == 1){

message.Lamp\_On(req, res);

socket.write('lo');

MongoClient.connect(url, function(error, database){

if(error){

console.log(error);

}else{

db.collection('Users').update({ user\_key : v\_user\_key},{ $set: { lamp\_status: 1}});

}

});

}

if(v\_content == "전등 끄기" && flag == 1){

message.Lamp\_Off(req, res);

socket.write('lc');

MongoClient.connect(url, function(error, database){

if(error){

console.log(error);

}else{

db.collection('Users').update({ user\_key : v\_user\_key},{ $set: { lamp\_status: 0}});

}

});

}

if(v\_content == "처음 화면" || v\_content == "ㄱ"){

keyboardObj.returnhome(req, res);

flag = 0;

}

};

exports.GasValve\_Open = function(req, res) {

var v\_type = req.body.type;

var v\_user\_key = req.body.user\_key;

var v\_content = req.body.content;

console.log("\*\*\* message.returnText =====================================");

console.log("\*\*\* type : " + v\_type);

console.log("\*\*\* user\_key : " + v\_user\_key);

console.log("\*\*\* content : " + v\_content);

res.send(

{

message:{

text: '가스 밸브가 열렸습니다. 테스트 성공\nㄱ을 입력하시면 처음 화면으로 되돌아갑니다.'

}

}

);

};

exports.GasValve\_Close = function(req, res) {

var v\_type = req.body.type;

var v\_user\_key = req.body.user\_key;

var v\_content = req.body.content;

console.log("\*\*\* message.returnText =====================================");

console.log("\*\*\* type : " + v\_type);

console.log("\*\*\* user\_key : " + v\_user\_key);

console.log("\*\*\* content : " + v\_content);

res.send(

{

message:{

text: '가스 밸브가 잠겼습니다. 테스트 성공\nㄱ을 입력하시면 처음 화면으로 되돌아갑니다.'

}

}

);

};

exports.Window\_Lock = function(req, res) {

var v\_type = req.body.type;

var v\_user\_key = req.body.user\_key;

var v\_content = req.body.content;

console.log("\*\*\* message.returnText =====================================");

console.log("\*\*\* type : " + v\_type);

console.log("\*\*\* user\_key : " + v\_user\_key);

console.log("\*\*\* content : " + v\_content);

res.send(

{

message:{

text: '창문이 잠겼습니다. 테스트 성공\nㄱ을 입력하시면 처음 화면으로 되돌아갑니다.'

}

}

);

};

exports.Window\_Unlock = function(req, res) {

var v\_type = req.body.type;

var v\_user\_key = req.body.user\_key;

var v\_content = req.body.content;

console.log("\*\*\* message.returnText =====================================");

console.log("\*\*\* type : " + v\_type);

console.log("\*\*\* user\_key : " + v\_user\_key);

console.log("\*\*\* content : " + v\_content);

res.send(

{

message:{

text: '창문 잠금이 해제되었습니다. 테스트 성공\nㄱ을 입력하시면 처음 화면으로 되돌아갑니다.'

}

}

);

};

exports.Lamp\_On = function(req, res) {

var v\_type = req.body.type;

var v\_user\_key = req.body.user\_key;

var v\_content = req.body.content;

console.log("\*\*\* message.returnText =====================================");

console.log("\*\*\* type : " + v\_type);

console.log("\*\*\* user\_key : " + v\_user\_key);

console.log("\*\*\* content : " + v\_content);

res.send(

{

message:{

text: '전등이 켜졌습니다. 테스트 성공\nㄱ을 입력하시면 처음 화면으로 되돌아갑니다.'

}

}

);

};

exports.Lamp\_Off = function(req, res) {

var v\_type = req.body.type;

var v\_user\_key = req.body.user\_key;

var v\_content = req.body.content;

console.log("\*\*\* message.returnText =====================================");

console.log("\*\*\* type : " + v\_type);

console.log("\*\*\* user\_key : " + v\_user\_key);

console.log("\*\*\* content : " + v\_content);

res.send(

{

message:{

text: '전등이 꺼졌습니다. 테스트 성공\nㄱ을 입력하시면 처음 화면으로 되돌아갑니다.'

}

}

);