Proyek - IEE3032 - Pemrograman Berbasis Web untuk IoT

SMART HOSPITAL SYSTEM

Martin Emmanuel Chang (212100199)
Moses Anthony Kwik (212100291)

TABLE OF CONTENT

OVERVIEW SYSTEM

Di industri kesehatan, mengelola dan menganalisis data pasien sangat penting. Sebuah proyek Django dapat dirancang untuk mengumpulkan, menyimpan, dan menganalisis data dari berbagai sumber dalam fasilitas kesehatan. Proyek ini dapat difokuskan pada meningkatkan perawatan pasien, mengoptimalkan kondisi ruangan operasi, dan memastikan kepatuhan terhadap regulasi kesehatan.

SUBSISTEM #1 - PATIENT MONITORING SYSTEM

Location: Patient Room
Sensors: IV Level sensor (0 - 1000 mL), Heartbeat sensor (0 - 120 bpm), Fluorescence-based glucose level sensor (100 - 190 mg/dL)
Actuator: Alert, IV flow regulator, Insulin pump

Di dalam patient room, ada sistem Patient Monitoring System yang menggunakan 3 sensor dan 3 aktuator.
Cara kerja aktuator:

- Akuator Alert akan aktif jika IV level sensor lebih rendah dari 10 mL atau Heartbeat sensor lebih rendah dari 70 bpm atau Glucose sensor lebih dari 160 mg/dL, atau semua kombinasi kondisi-kondisi ini.
- IV Flow Regulator akan aktif untuk membuat meningkatkan flow rate cairan IV jika heartbeat sensor dibawah 90 bpm dan level sensor di atas 150 mL.
- Insulin Pump akan aktif untuk meningkat inslutin pump jika glucose level lebih dari 150 mg/dL.

SUBSISTEM #2 - VISITOR HEALTH CHECK SYSTEM

Location: Hospital
Sensors: Camera detection (Boolean), Microbial sensor (50 - 250 j/m^3), Thermal sensor (35 - 40 C)
Actuator: Laser pointer, Door lock, Disinfectant spray

Ada sistem Visitor Health Check System yang menggunakan 3 sensor dan 3 aktuator.

- Laser pointer akan aktif jika Camera mendeteksi visitor yang tidak pakai masker
- Door lock akan aktif jika visitor tidak memakai masker, Microbial sensor mendeteksi bakteri sebanyak 150 J/m^3 di visitor atau Thermal sensor mendeteksi suhu badan yang diatas 38°C, atau semua kombinasi kondisi-kondisi ini.
- Disinfectant spray akan aktif jika Microbial sensor mendeteksi bakteri sebanyak 150 J/m^3 di visitor.

SUBSISTEM #3 - SURGERY ROOM CONDITIONING SYSTEM

Location: Surgery Room
Sensors: Room pressure sensor (0 - 15 Pa), Temperature sensor (20 - 30 C), Humidity sensor (0 - 80 %)
Actuator: Electric damper, HVAC system, Alarm

Ada sistem Operating Room Conditioning System yang menggunakan 3 sensor dan 3 aktuator.

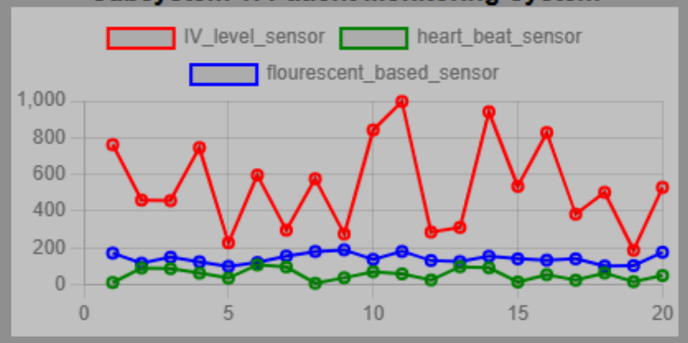
- HVAC Sistem akan aktif menyalahkan cooling system jika sensor temperatur diatas 25 °C atau sensor humiditas diatas 65%, atau dua-duanya.
- Alarm akan aktif jika sensor temperatur diatas 27 °C atau sensor humiditas diatas 70% atau sensor room pressure di bawah 2.0 Pa atau semua kombinasi kondisi-kondisi ini.
- Electric damper akan aktif untuk meningkatkan tekanan ruangan jika sensor room pressure dibawah 2.5 Pa.

HALAMAN UTAMA

Smart Hospital System

INFO

Subsystem 1: Patient Monitoring System

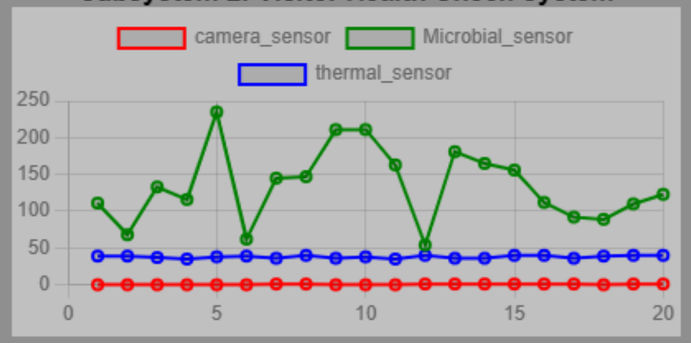


alert_actuator | IV_flow_regulator_actuator | insulin_pump_actuator

ML Output: 1 | ML Output: 1 | ML Output: 1

LOG

Subsystem 2: Visitor Health Check System

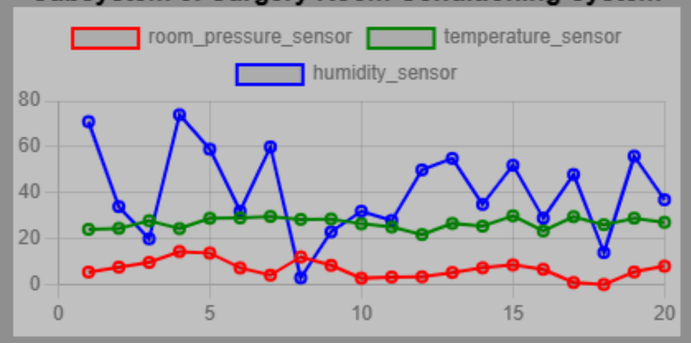


laser_pointer_actuator | door_lock_actuator | disinfectant_spray_actuator

ML Output: 1 | ML Output: 1 | ML Output: 1

LOG

Subsystem 3: Surgery Room Conditioning System



electric_damper_actuator | HVAC_system_actuator | alarm_actuator

ML Output: 1 | ML Output: 1 | ML Output: 1

LOG

212100291 Moses Anthony Kwik
212100199 Martin Emmanuel Chang

HALAMAN INFO

BACK

System Explanation

In the healthcare industry, managing and analyzing patient data is crucial. A Django project can be designed to collect, store, and analyze data from various sources within a healthcare facility. This project can be focused on improving patient care, optimizing hospital operations, and ensuring compliance with health regulations.

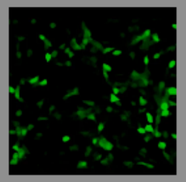
Patient Monitoring System

Location: Patient's room

SENSORS

IV Level Sensor: Measures the infusion fluid level in the patient's infuse bag. Threshold is 0 - 1000mL.

Heartbeat Sensor: Measures the patient's heartbeat rate. Threshold is 0 - 120 bpm.



Flourescent Based Glucose Sensor: Measures the concentration of glucose in the patient's blood stream using fluorophore. Threshold is 100 - 190 mg/dL.

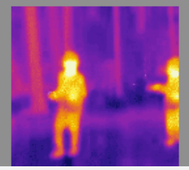
Visitor Health Check System

Location: All around the hospital

SENSORS

Camera Detection Sensor: Uses facial and object recognition system to detect whether a visitor is wearing a medical face mask or not around the hospital. Threshold is 0 (mask detected) or 1 (no mask detected).

Microbial Sensor: Using UV-C light to detect presence of germs or microorganisms on a visitor before they enter the hospital. Threshold is 50 - 250 J/m^3.

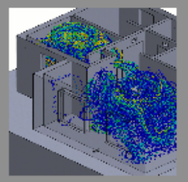


Thermal Sensor: Measures the visitor's body temperature before entering the hospital. Threshold is 35 - 40 °C.

Surgery Room Conditioning System

Location: Surgery room

SENSORS



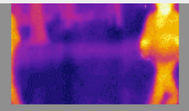
Room Pressure Sensor: Measures the surgery room's air pressure. Maintaining good air pressure in a surgery room is crucial for infection control, airborne pathogen control and ventilation. Threshold is 0 - 15 Pa.

Temperature Sensor: Measures the surgery room's temperature. Maintaining good room temperature in a surgery room is crucial for patient comfort, staff comfort and equipment performance. Threshold is 20 - 30 °C.

Humidity Sensor: Measures the surgery room's humidity. Maintaining good room

HALAMAN INFO

100 - 190 mg/dL.



temperature before entering the hospital. Threshold is 35 - 40 °C.

Humidity Sensor: Measures the room's h
Maintaining good room humidity in a surgery room is crucial for static electricity control, equipment performance, patient and staff comfort, and wound tissue healing. Threshold is 0 - 80 %.

BACK

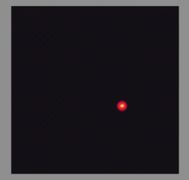
ACTUATORS

Alert Actuator: The alert actuator will activate if the IV level sensor value is below 10 mL or the heartbeat sensor value is below 70 bpm or the flourescent based glucose sensor value is above 160 mg/dL, or any combination of these conditions.

IV Flow Actuator: The IV flow actuator will activate to increase the IV fluid flow rate if the heartbeat sensor value is below 90 bpm and the level sensor value is above 150 mL.

Insulin Pump Actuator: The insulin pump actuator will activate to increase insulin delivery if the glucose level is above 150 mg/dL.

ACTUATORS

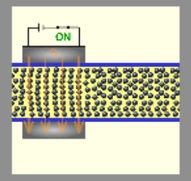


Laser Pointer Actuator: Laser pointer actuator is activated when the camera detection sensor returns value 1 or there is a visitor that doesn't uses a medical face mask.

Door Lock Actuator: The door lock actuator will activate if the visitor is not wearing a medical face mask, the microbial sensor detects bacteria of 150 J/m^3 on the person, or the thermal sensor detects a body temperature above 38°C, or any combination of these conditions.

Disinfectant Spray Actuator: The disinfectant spray actuator will activate if the microbial sensor detects bacteria of 150 J/m^3 on the visitor.

ACTUATORS

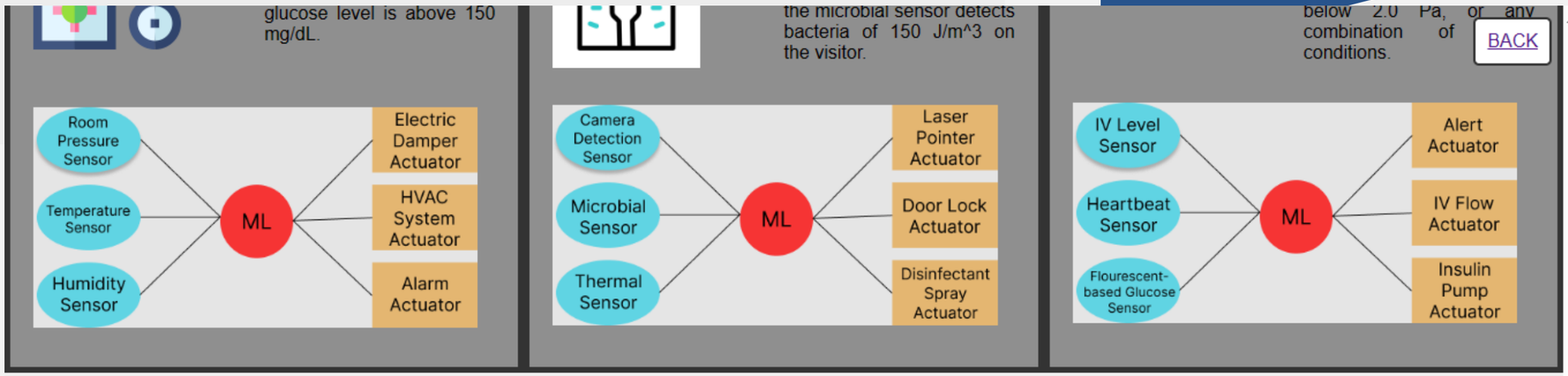


Electric Damper Actuator: The electric damper actuator will activate to increase room pressure if the room pressure sensor reads below 2.5 Pa.

HVAC System Actuator: The HVAC system actuator will activate to turn on the cooling system if the temperature sensor reads above 25 °C or if the humidity sensor reads above 65%, or both.

Alarm Actuator: The alarm actautor will activate if the temperature sensor reads above 27 °C, or if the humidity sensor reads above 70%, or if the room pressure sensor reads below 2.0 Pa, or any combination of these conditions.

HALAMAN INFO



Machine Learning Technique: Logistic Regression with CSV Files

Machine Learning Technique: Logistic Regression
Logistic Regression is a statistical model used for binary classification problems. It models the probability that an instance belongs to a particular class using a logistic function.

Training Using CSV Files
CSV (Comma-Separated Values) files are used to store tabular data, with each row representing a data instance. Training involves using a labeled dataset to teach the model, and the CSV files store the features and labels for training instances.

Integration of Logistic Regression and CSV Files
Logistic regression is applied to the data stored in CSV files during the training process. The model adjusts its parameters to minimize the difference between predictions and actual labels, learning patterns from the training data.

Outcome
After training, the logistic regression model can predict the probability of an instance belonging to a class. It can be applied to new data, calculating probabilities and assigning classes based on a chosen threshold.

HALAMAN LOG1

Patient Monitoring System Log

BACK

Timestamp	IV Level Sensor	Heartbeat Sensor	Fluorescent Based Sensor	Alert	IV Flow Regulator	Insulin Pump
2023-12-09 22:24:14	143.39	58.0	152.0	0.0	1.0	1.0
2023-12-09 22:24:07	368.27	74.0	166.0	0.0	1.0	0.0
2023-12-09 22:23:58	735.03	87.0	184.0	1.0	1.0	1.0
2023-12-09 22:23:51	558.43	35.0	136.0	0.0	1.0	1.0
2023-12-09 22:23:42	191.88	68.0	137.0	0.0	1.0	1.0
2023-12-09 22:23:33	319.58	31.0	189.0	1.0	1.0	1.0
2023-12-09 22:23:23	119.24	104.0	143.0	1.0	1.0	1.0
2023-12-09 22:23:14	199.91	31.0	134.0	1.0	1.0	1.0
2023-12-09 22:23:06	703.44	56.0	142.0	1.0	1.0	1.0
2023-12-09 22:22:59	657.09	57.0	114.0	0.0	1.0	1.0
2023-12-09 22:22:53	643.69	113.0	167.0	0.0	1.0	1.0
2023-12-09 22:22:46	509.84	84.0	144.0	0.0	1.0	1.0
2023-12-09 22:22:40	174.39	113.0	119.0	0.0	1.0	1.0
2023-12-09 22:22:34	892.57	49.0	186.0	0.0	1.0	1.0
2023-12-09 22:22:23	527.94	74.0	144.0	1.0	0.0	1.0
2023-12-09 22:22:21	63.14	1.0	125.0	0.0	1.0	1.0
2023-12-09 22:22:20	25.46	110.0	153.0	0.0	0.0	1.0
2023-12-09 22:22:18	12.78	89.0	120.0	0.0	0.0	0.0
2023-12-09 22:22:16	155.57	48.0	151.0	0.0	1.0	0.0
2023-12-09 22:22:14	667.51	12.0	149.0	0.0		1.0
				0.0		

HALAMAN LOG2

Health Regulation System Log

BACK

Timestamp	Camera_sensor	Microbial_sensor	Thermal_sensor	Laser_pointer	Door_lock	Disinfectant_spray
2023-12-09 22:25:36	1.00	148.0	38.0	1.0	1.0	1.0
2023-12-09 22:25:26	0.00	233.0	36.0	0.0	0.0	1.0
2023-12-09 22:25:15	0.00	130.0	35.0	0.0	0.0	1.0
2023-12-09 22:24:56	1.00	222.0	38.0	0.0	0.0	0.0
2023-12-09 22:24:50	0.00	151.0	35.0	1.0	0.0	0.0
2023-12-09 22:24:42	0.00	125.0	39.0	0.0	1.0	0.0
2023-12-09 22:24:39	0.00	60.0	37.0	1.0	0.0	0.0
2023-12-09 22:24:31	1.00	233.0	37.0	1.0	0.0	1.0
2023-12-09 22:24:22	0.00	55.0	38.0	0.0	0.0	0.0
2023-12-09 22:24:14	0.00	159.0	37.0	1.0	0.0	0.0
2023-12-09 22:24:07	1.00	243.0	39.0	1.0	0.0	0.0
2023-12-09 22:23:59	1.00	170.0	38.0	1.0	1.0	1.0
2023-12-09 22:23:52	1.00	229.0	35.0	1.0	1.0	1.0
2023-12-09 22:23:43	1.00	220.0	39.0	0.0	1.0	1.0
2023-12-09 22:23:33	1.00	169.0	36.0	0.0	1.0	1.0
2023-12-09 22:23:23	0.00	157.0	35.0	1.0	0.0	1.0
2023-12-09 22:23:15	0.00	75.0	35.0	1.0	1.0	1.0
2023-12-09 22:23:06	1.00	186.0	39.0	0.0	0.0	1.0
2023-12-09 22:23:00	0.00	62.0	35.0	1.0	0.0	1.0
2023-12-09 22:22:54	0.00	66.0	38.0	0.0	1.0	1.0

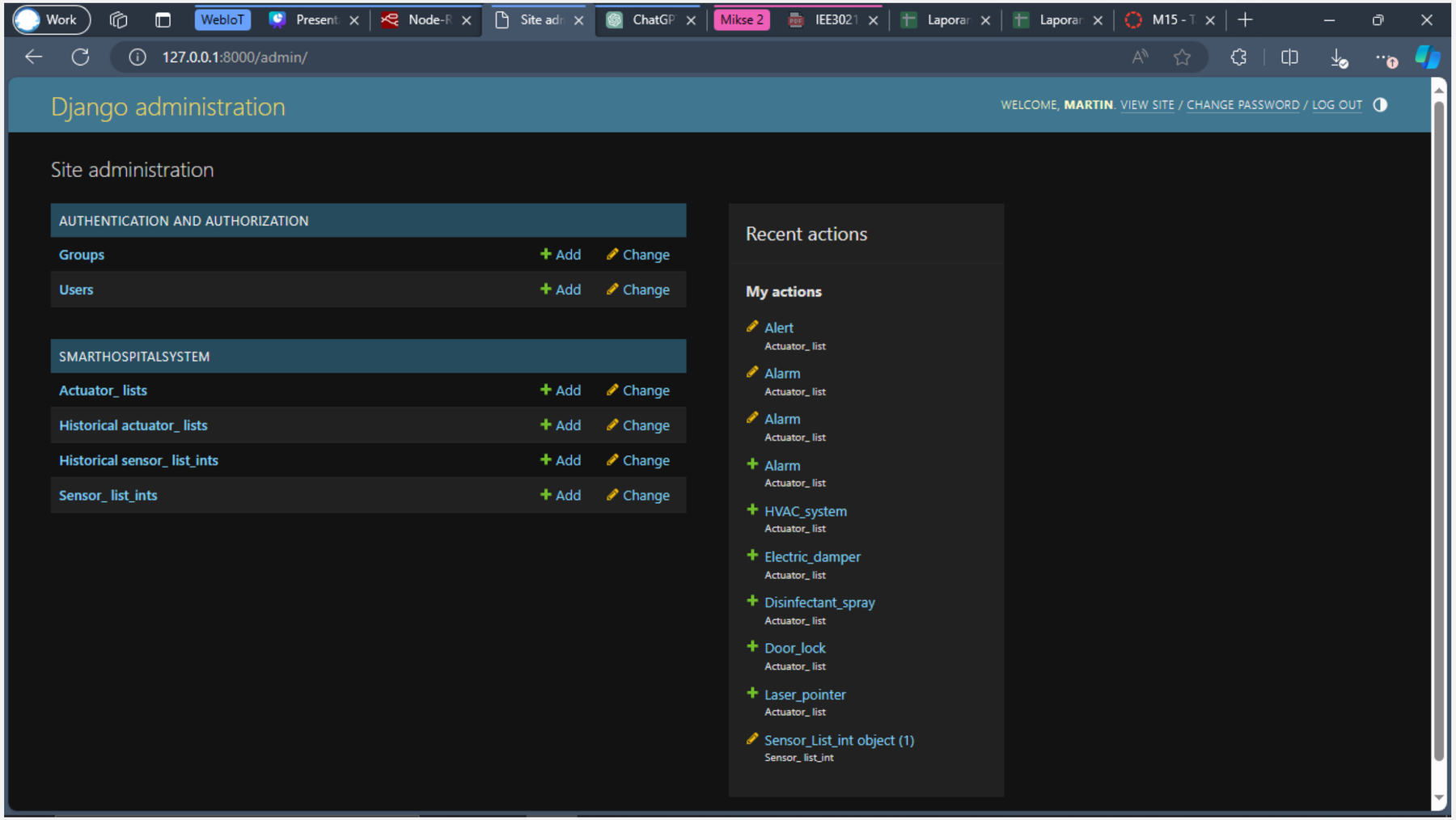
HALAMAN LOG3

Surgery Room Conditioning System Log

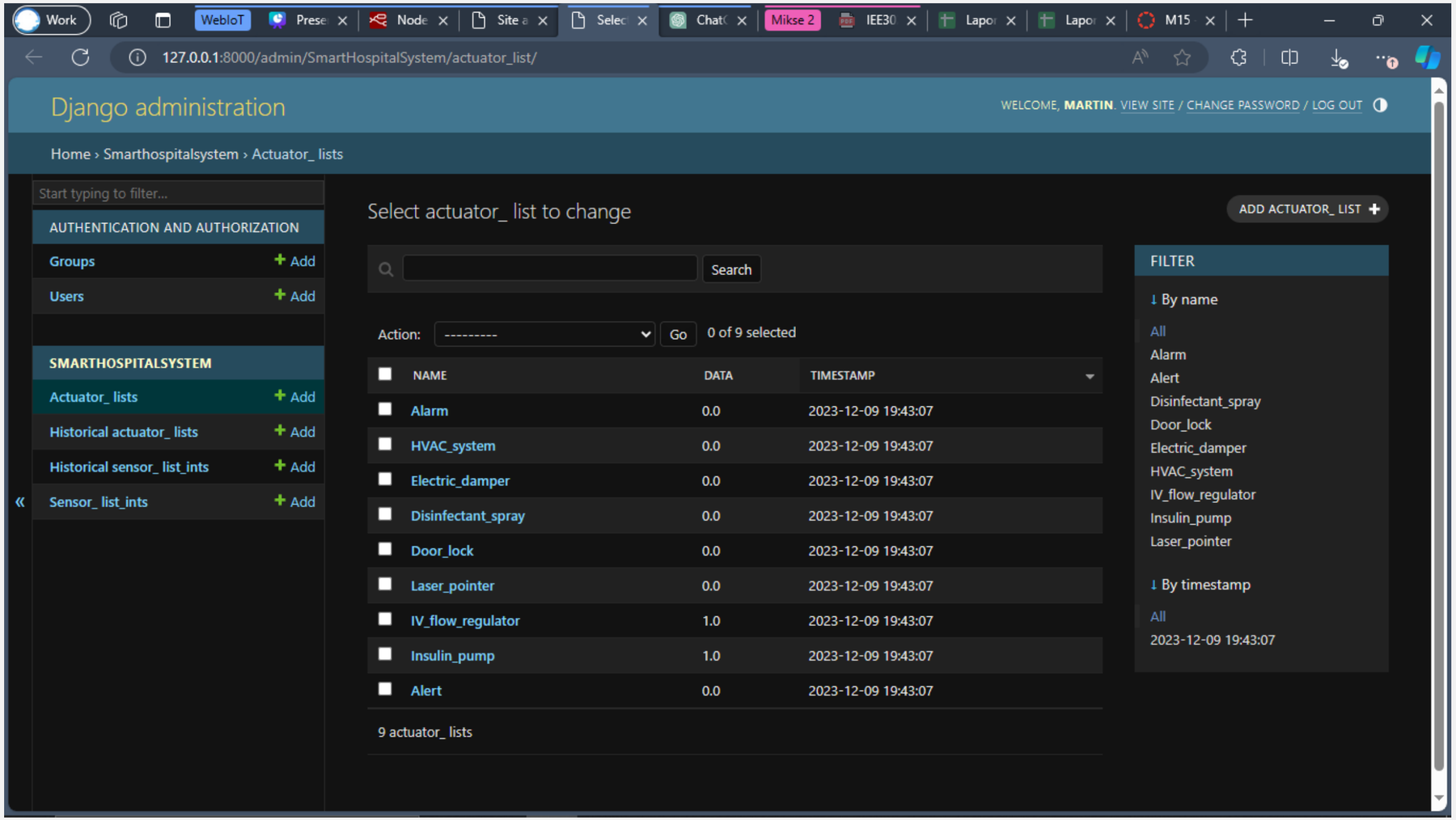
BACK

Timestamp	Room_pressure_sensor	Temperature_sensor	Humidity_sensor	Electric_damper	HVAC_system	Alarm
2023-12-09 22:26:31	1.33	23.26	46.0	1.0	1.0	1.0
2023-12-09 22:26:24	13.14	23.19	50.0	0.0	0.0	0.0
2023-12-09 22:26:20	0.94	22.55	77.0	0.0	0.0	0.0
2023-12-09 22:26:18	6.99	25.12	43.0	1.0	1.0	1.0
2023-12-09 22:26:16	13.51	24.91	61.0	0.0	0.0	0.0
2023-12-09 22:26:14	11.38	20.50	38.0	0.0	0.0	0.0
2023-12-09 22:26:12	3.14	27.04	24.0	1.0	1.0	1.0
2023-12-09 22:26:09	9.08	27.00	39.0	1.0	1.0	1.0
2023-12-09 22:26:06	10.16	22.35	29.0	0.0	0.0	0.0
2023-12-09 22:25:58	13.01	23.50	11.0	1.0	1.0	1.0
2023-12-09 22:25:50	10.81	20.04	0.0	1.0	1.0	1.0
2023-12-09 22:25:43	14.99	26.74	31.0	1.0	1.0	1.0
2023-12-09 22:25:36	3.40	27.07	21.0	1.0	1.0	1.0
2023-12-09 22:25:27	0.05	22.02	5.0	1.0	1.0	1.0
2023-12-09 22:25:16	11.09	20.31	14.0	1.0	1.0	1.0
2023-12-09 22:24:57	1.49	22.22	28.0	0.0	0.0	0.0
2023-12-09 22:24:50	14.21	28.50	9.0	0.0	0.0	0.0
2023-12-09 22:24:42	0.20	20.14	6.0	0.0	0.0	0.0
2023-12-09 22:24:39	14.56	25.50	29.0	0.0	0.0	0.0
2023-12-09 22:24:32	1.43	28.92	23.0	1.0	1.0	1.0

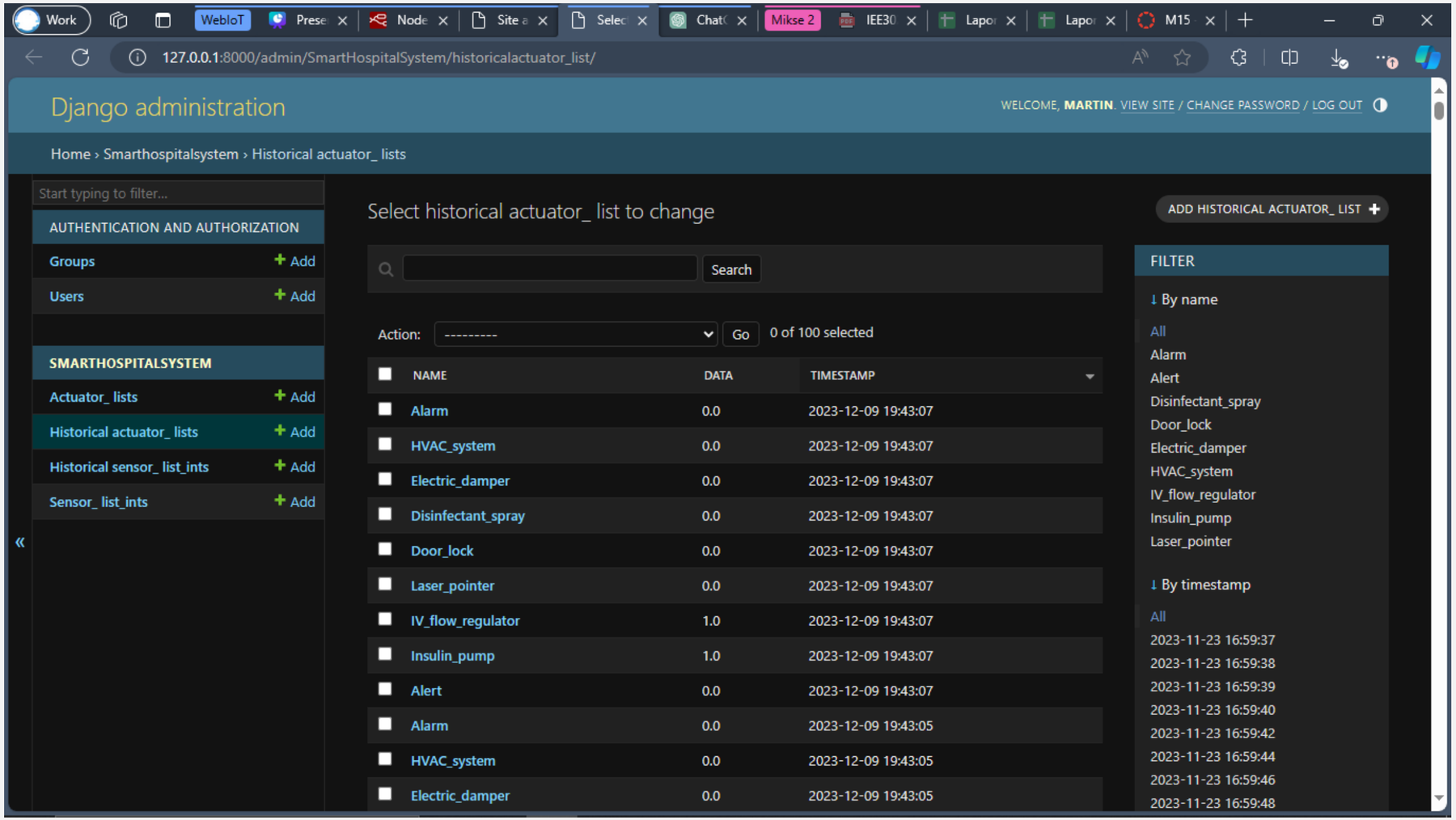
HALAMAN ADMIN DAN DATABASE



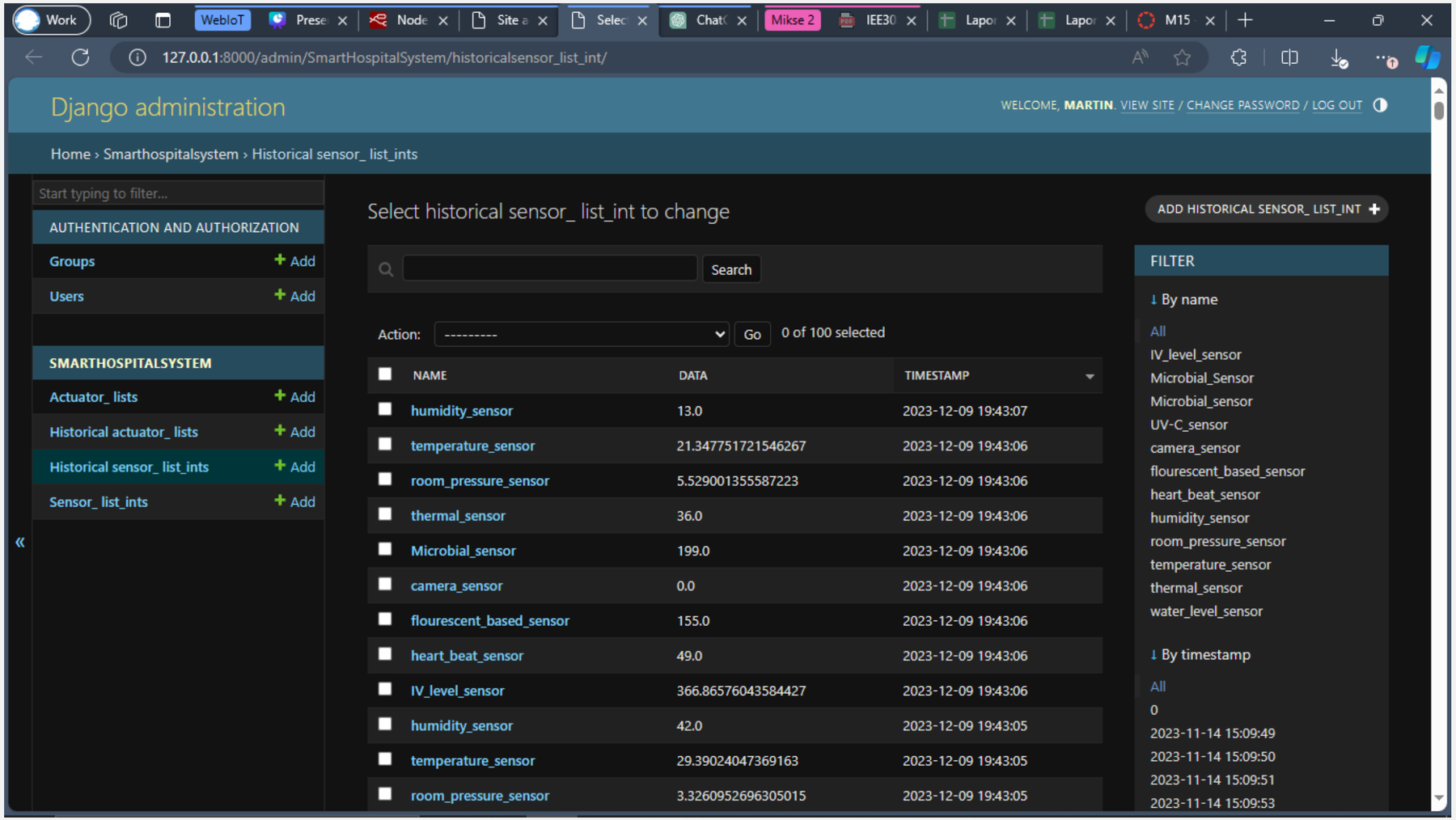
HALAMAN ADMIN DAN DATABASE



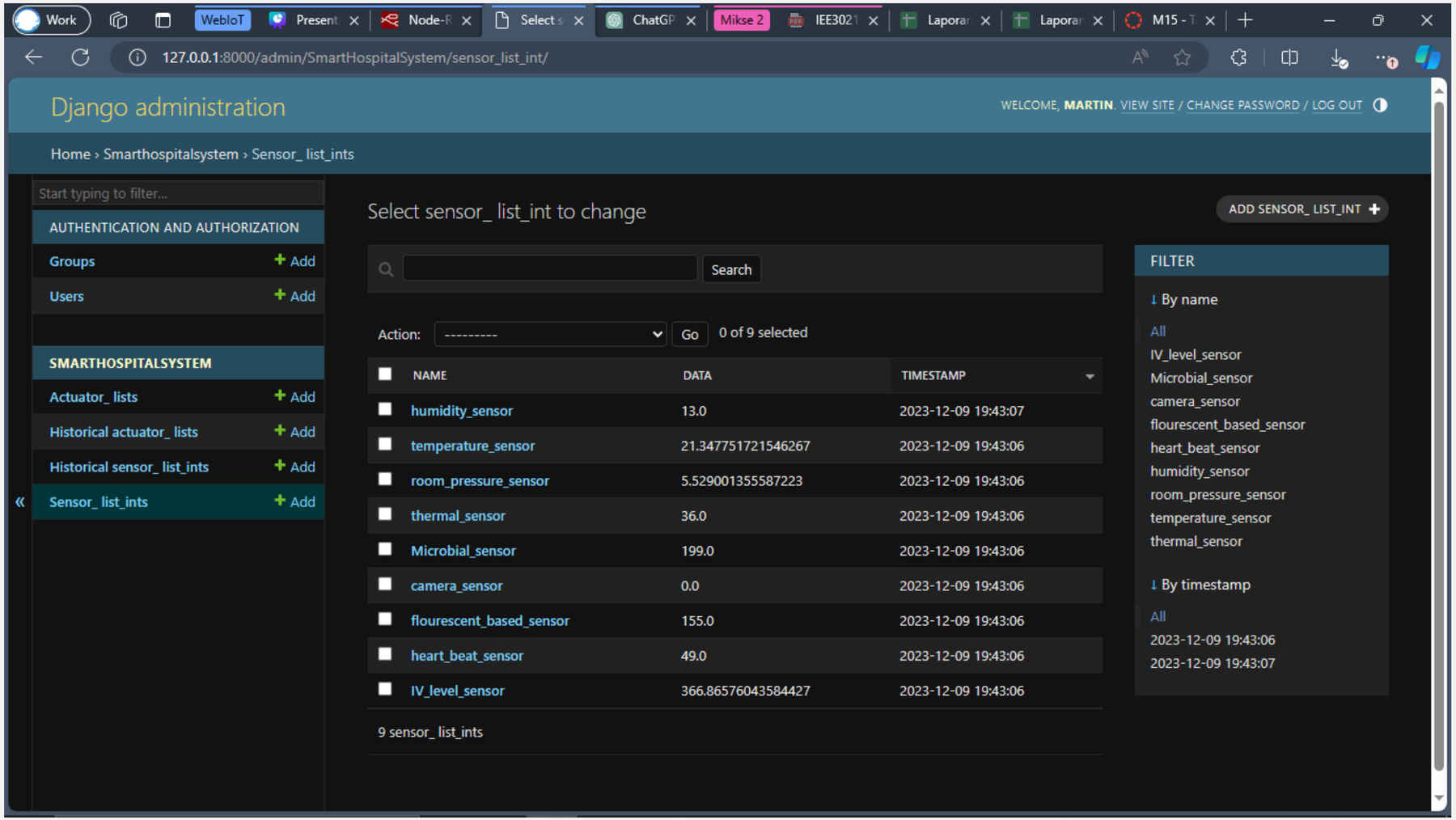
HALAMAN ADMIN DAN DATABASE



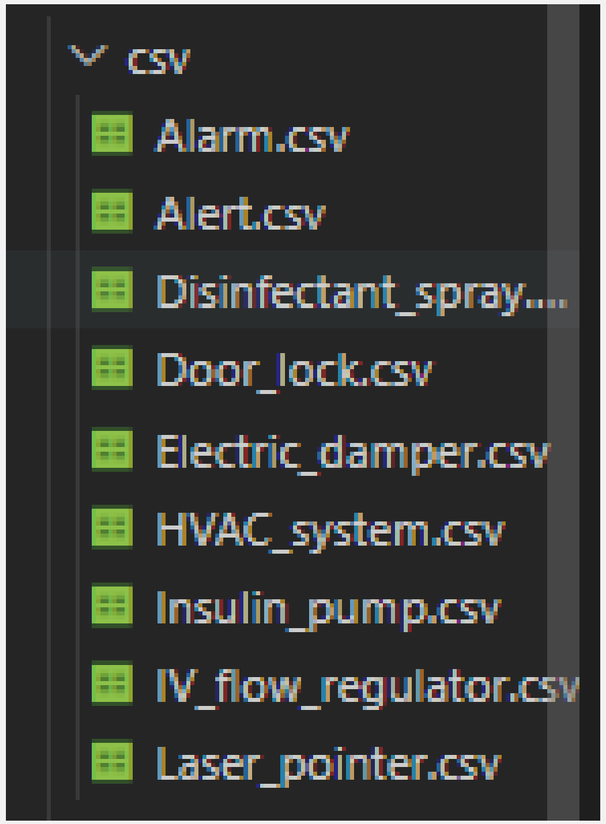
HALAMAN ADMIN DAN DATABASE



HALAMAN ADMIN DAN DATABASE

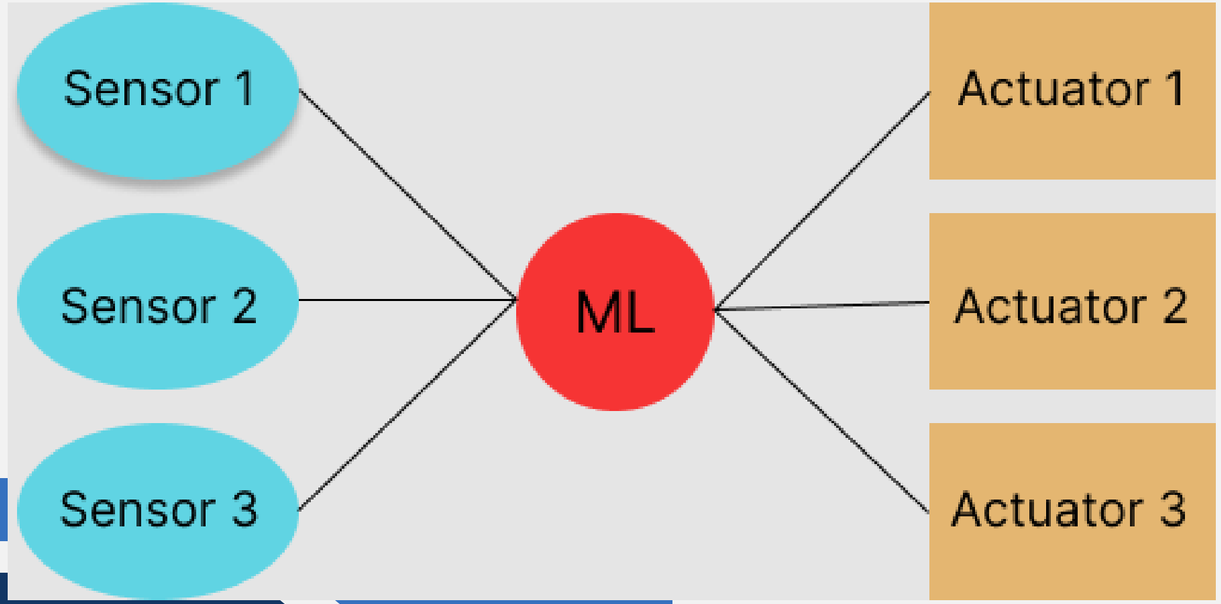


MACHINE LEARNING DATA



Project_UAS > csv > Insulin_pump.csv

```
1   IV_level_sensor,heart_beat_sensor,flourescent_based_sensor,Insulin_pump
2   10,89,100,0
3   49,70,120,0
4   10,95,149,0
5   10,95,150,0
6   10,95,151,0
7   20,89,200,1
8   40,80,160,1
9   50,120,198,1
10
```

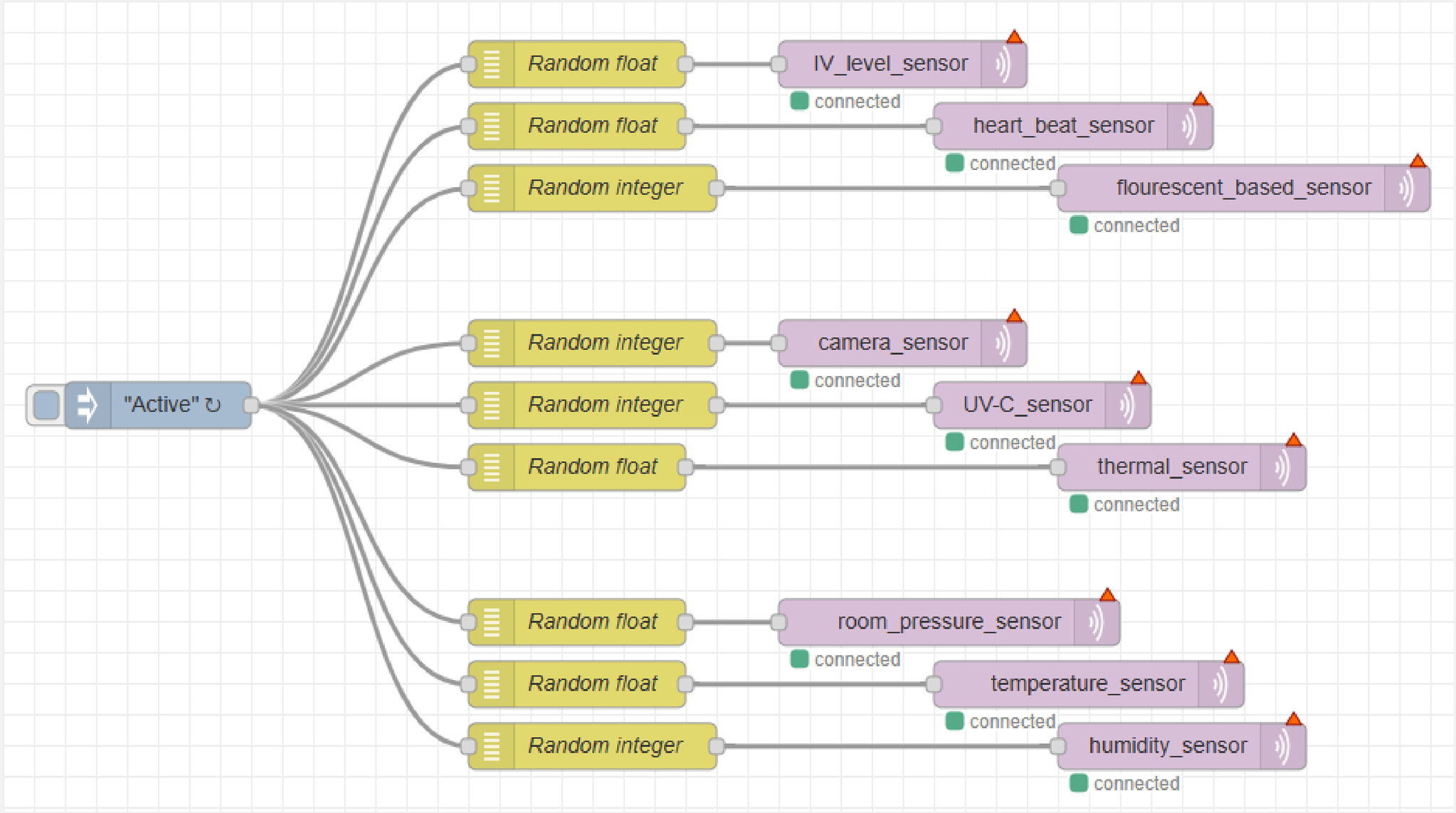


MACHINE LEARNING CODE

```
for actuator in actuators_table1:
    df = pd.read_csv("S:/Uni_stuff/Semester_5/UAS WEB/Project_UAS/csv/" + str(actuator) + ".csv")

    sensor_names_table1 = [sensor.name for sensor in sensors_table1]
    features = df[sensor_names_table1]
    target = df[str(actuator)]

    X_train, X_test, y_train, y_test = train_test_split(features, target, test_size=0.2, random_s

    # Create a logistic regression model
    model = LogisticRegression()

    # Train the model
    model.fit(X_train, y_train)

    # Make predictions on the test set
    predictions = model.predict(X_test)

    # Calculate accuracy (you might want to store or log this value for evaluation)
    accuracy = accuracy_score(y_test, predictions)

    # Use the latest sensor data for prediction
    #new_data = [[latest_sensor_data_dict[sensor] for sensor in features.columns]]


    new_data = pd.DataFrame(
    np.array([[latest_sensor_data_dict[sensor] for sensor in features.columns]]).reshape(1, -1),
    columns=['IV_level_sensor', 'heart_beat_sensor', 'flourescent_based_sensor']
)
    result = model.predict(new_data)
```

Penjelasan:
Dalam kode, nama sensor diambil dari list terus dibuat sebagai features yaitu row atas di csv data. target dipilih sebagai aktuatornya.
Model yang digunakan adalah Logistic Regression dan akan di train menggunakan data dari csv file. Kemudian model ini akan digunakan untuk predict nilai akuator dengan values sensor saat ini.

NODE-RED


"Active" ↻
Random float
Random float
Random integer
Random integer
Random integer
Random float
Random float
Random integer
IV_level_sensor
heart_beat_sensor
flourescent_based_sensor
camera_sensor
UV-C_sensor
thermal_sensor
room_pressure_sensor
temperature_sensor
humidity_sensor
connected

MQTT KODE

```python
def on_connect(client, userdata, flags, rc, ):
    print("Connected with result code " + str(rc))

    # Subscribe to topics for all sensors
    for sensor_name in sensor_names:
        client.subscribe(sensor_name)

def on_message(client, userdata, msg):
    global sensor_data_history_list

    sensor_name = msg.topic
    latest_sensor_data = float(msg.payload.decode())

    # Add the latest sensor data to the history list
    current_time = datetime.now().strftime("%Y-%m-%d %H:%M:%S")
    sensor_data_history_list.append({'sensor_name': sensor_name, 'data': latest_sensor_data, 'timestamp': current_time})

    # Keep only the last 100 sensor readings in the list
    sensor_data_history_list = sensor_data_history_list[-20:]

    # Update the latest sensor data in the dictionary
    latest_sensor_data_dict[sensor_name] = latest_sensor_data
    historical_sensor_data_dict[sensor_name].append({'data': latest_sensor_data, 'timestamp': current_time})
    historical_sensor_data_dict[sensor_name] = historical_sensor_data_dict[sensor_name][-20:]

# Set up the MQTT client
mqtt_client = mqtt.Client()
mqtt_client.on_connect = on_connect
mqtt_client.on_message = on_message
mqtt_broker_address = "127.0.0.1"
mqtt_client.connect(mqtt_broker_address, 1889)
mqtt_client.loop_start()
```

THANK YOU