MARIoT: An Authoring Framework for Creating IoT Applications with Mobile Augmented Reality

Meral Kuyucu, Gökhan İnce

User Experience Laboratory
Department of Computer Engineering
Istanbul Technical University (Turkey)

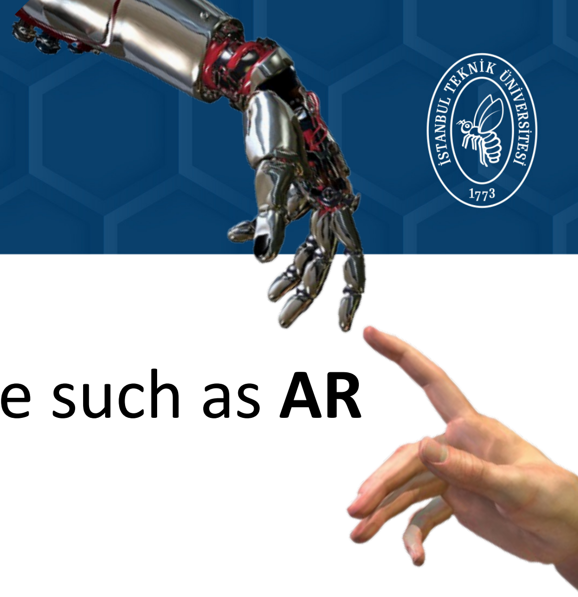
Contact email: **korkmazmer@itu.edu.tr**

ACHI 2021

Meral Kuyucu

Meral Kuyucu received her BSE and MS degrees from the Department of Computer Engineering at Istanbul Technical University (ITU) in 2017 and 2021 respectively. She received the ITU-Turkcell academic grant throughout her graduate studies. She will continue to pursue a PhD degree at ITU where she is currently working as a Research and Teaching Assistant at the User Experience Lab. She will continue to conduct research on AR-enabled IoT systems.

Introduction



- With the launch of the internet, components of the digital age such as **AR** and **IoT** took off.

Technology fusion:
Cooperative
Complementary
[Tidd, 2013]

→

Harness advantages
Mitigate weaknesses

We propose the fusion of **AR** and **IoT**

IoT Exists in 3D Space
+
Need Intuitive Interaction

→

3D Surface of Interaction

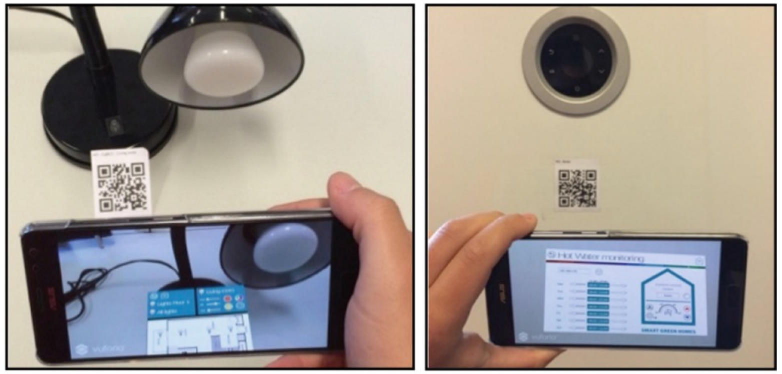
Proposed System

- **MARIoT: Mobile Augmented Reality for the Internet of Things**
  - IoT Network
  - Communication Server
  - AR Authoring Tool and Application Generator

- **Research Questions**
  - Will **tech-savvy but not necessarily code-savy** users find the suggested framework helpful when creating customized AR enabled smart environments?
  - Will users find an AR interface for interacting with IoT devices **intuitive?**

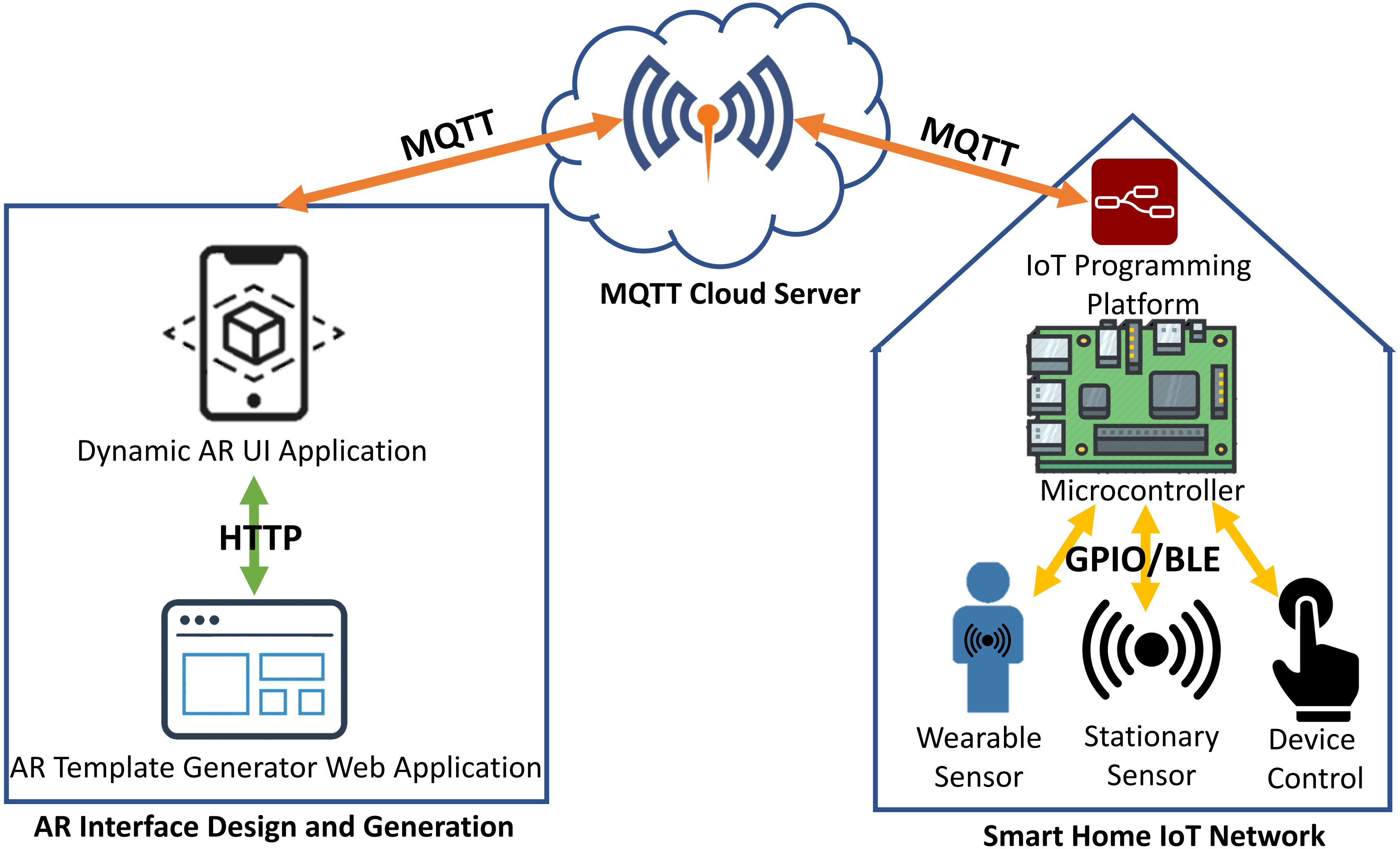
Literature Review

- Many studies fuse AR and IoT at the **implementation level**.
- Most studies do not conduct usability tests [Marques et al., 2019].
- Studies that do, **do not aim for Low Code/No Code (LCNC) solutions**.

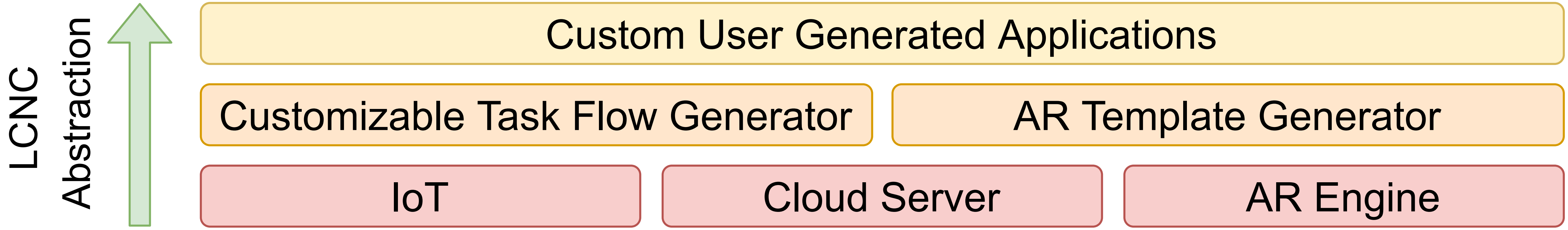


Although many studies in literature investigate End-User Development (EUD) for IoT and AR **separately**, to our knowledge, there was no research conducted on authoring tools for the **integration of AR and IoT.**

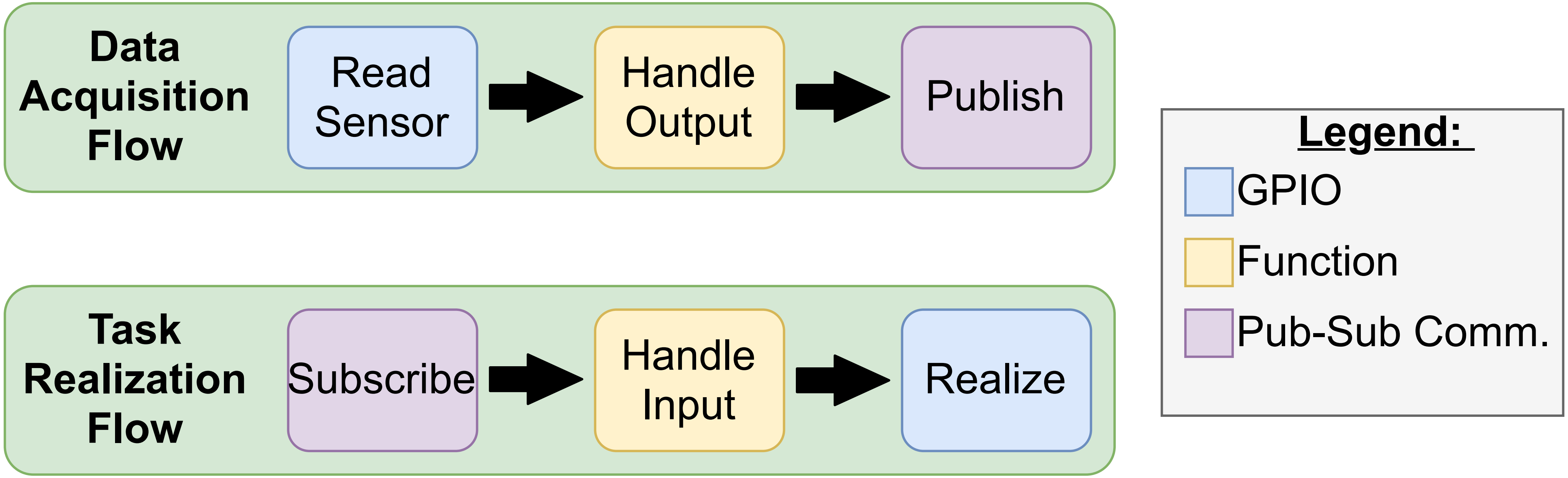
System Components of MARIoT



Architecture of MARIoT


LCNC
Abstraction
Custom User Generated Applications
Customizable Task Flow Generator
AR Template Generator
IoT
Cloud Server
AR Engine

Customizable Task Flow Generator for IoT



Data Acquisition Flow

Read Sensor → Handle Output → Publish

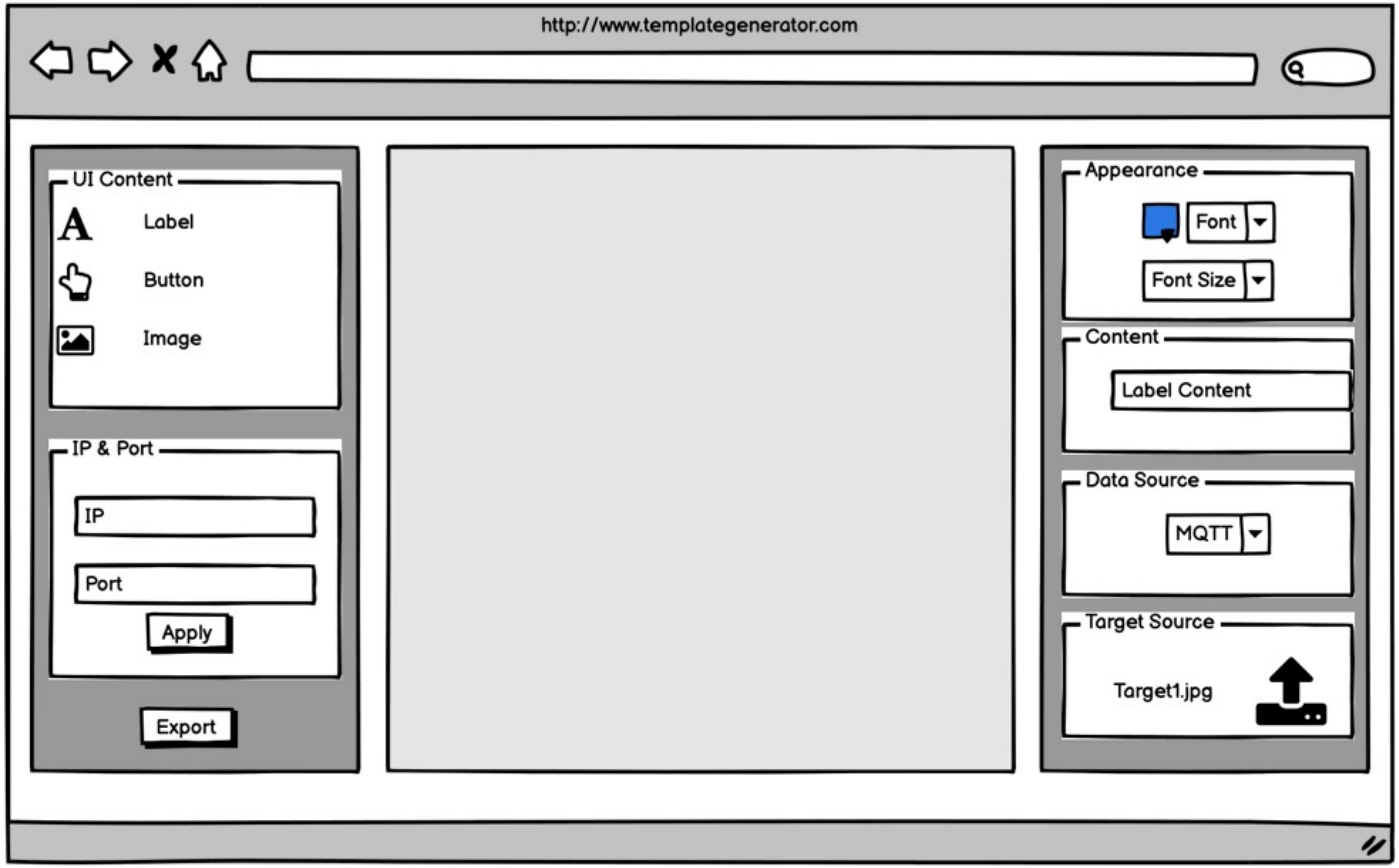
Task Realization Flow

Subscribe → Handle Input → Realize

Legend:

- 🟦 GPIO
- 🟨 Function
- 🟪 Pub-Sub Comm.

Template Generator for AR Applications



Pub-Sub Messaging Communication Protocol

- **Backbone of System = Communication of Elements**

- **MQTT:** Lightweight, pub-sub network protocol that transports messages among devices.
    - **Publisher:** Broadcasts messages with topics
    - **Subscriber:** Listens for messages with specific topics

Software

Dynamic AR UI Generator

vuforia

AR Template Generator

HTML

CSS

jQuery user interface

JS

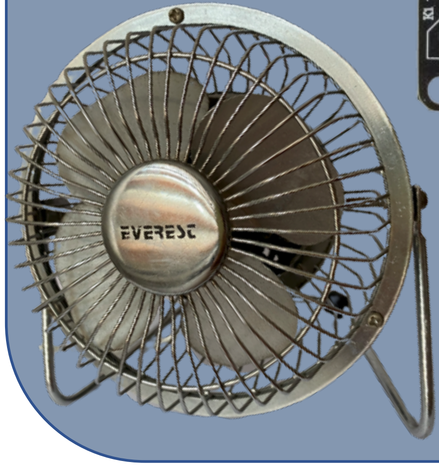
Customizable Task Flow Generator

Node-RED

Pub-Sub Broker

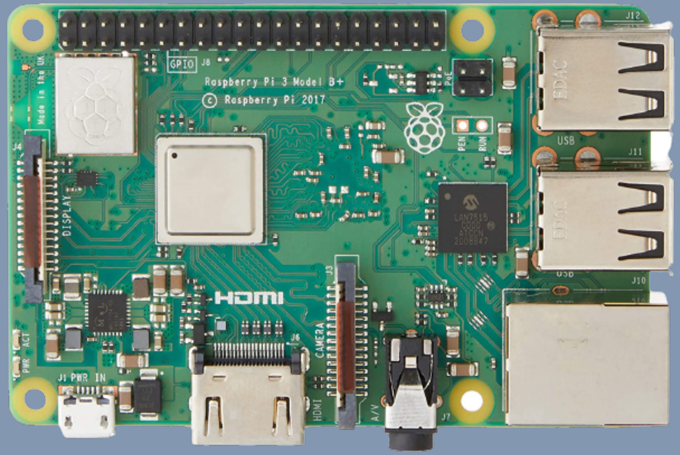
Hardware

Actuators:



Relay Switch

Microcontroller:



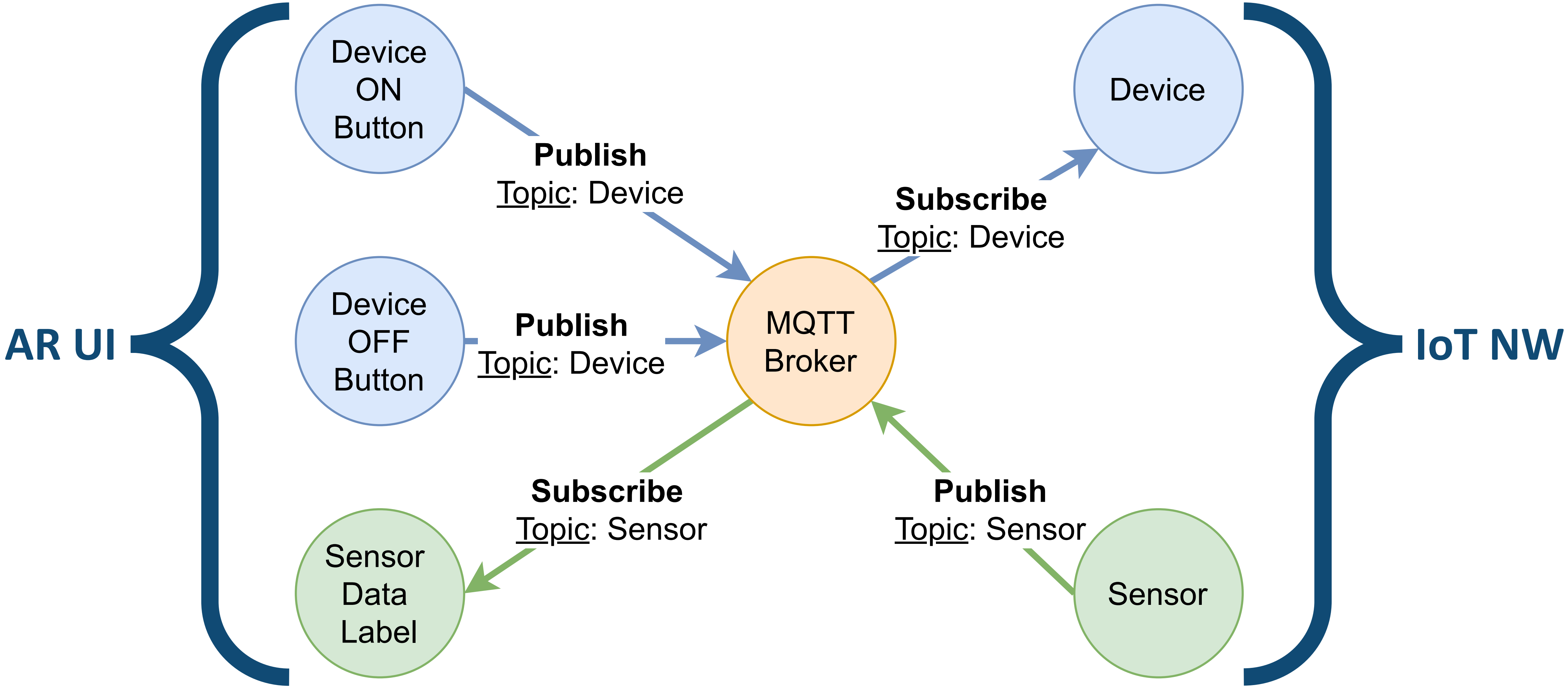
Raspberry Pi 3B+

Sensors:

Microphone

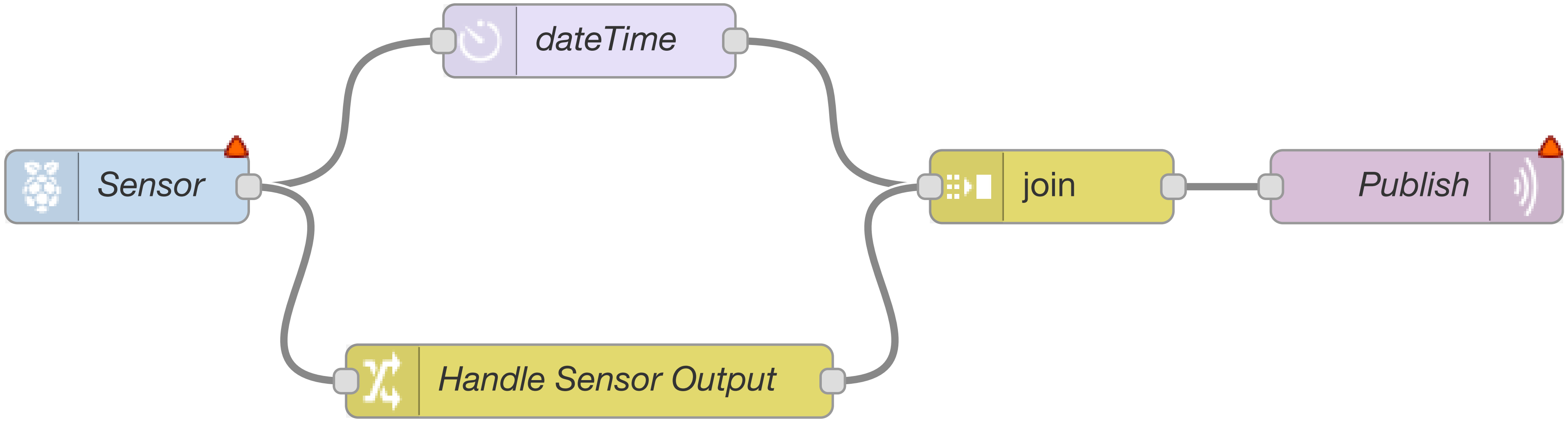
Photoresistor

Pub-Sub Communication Architecture

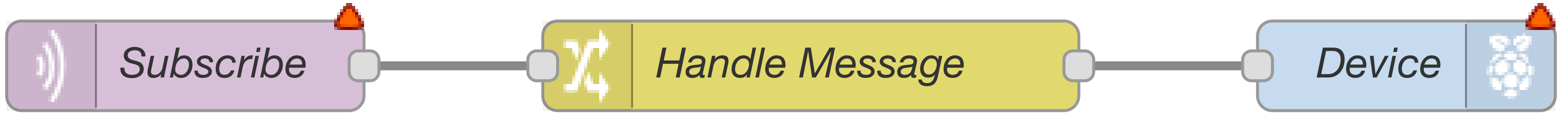


Customizable Task Flow Generator

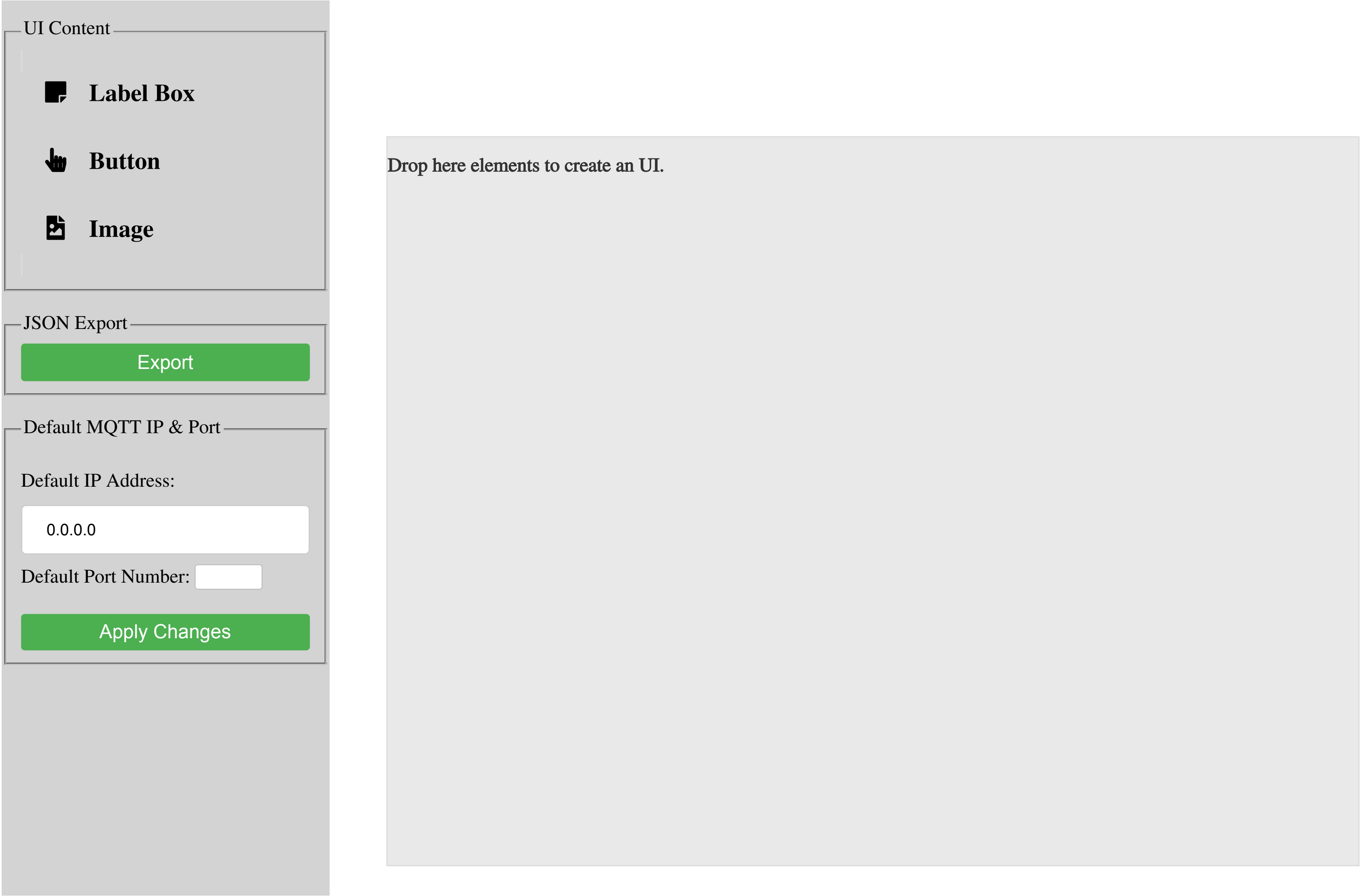
Data Acquisition Flow



Task Realization Flow



AR Template Generator

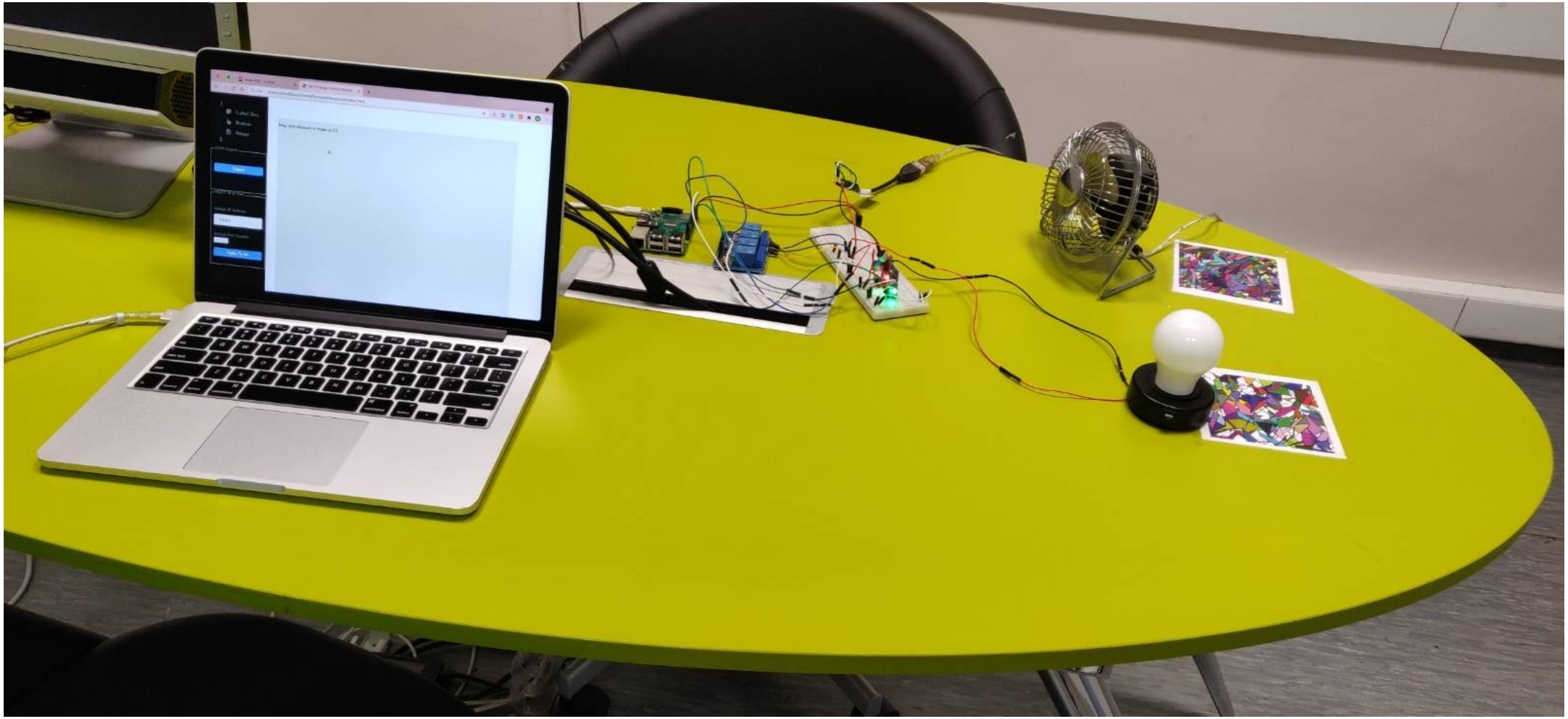


Default IP Address: 0.0.0.0

Default Port Number:

Drop here elements to create an UI.

Experimental setup



Pilot Studies & Participant Demographics

- **Two** pilot studies were conducted with **expert users.**

- Determine **maximum time** to complete a task.

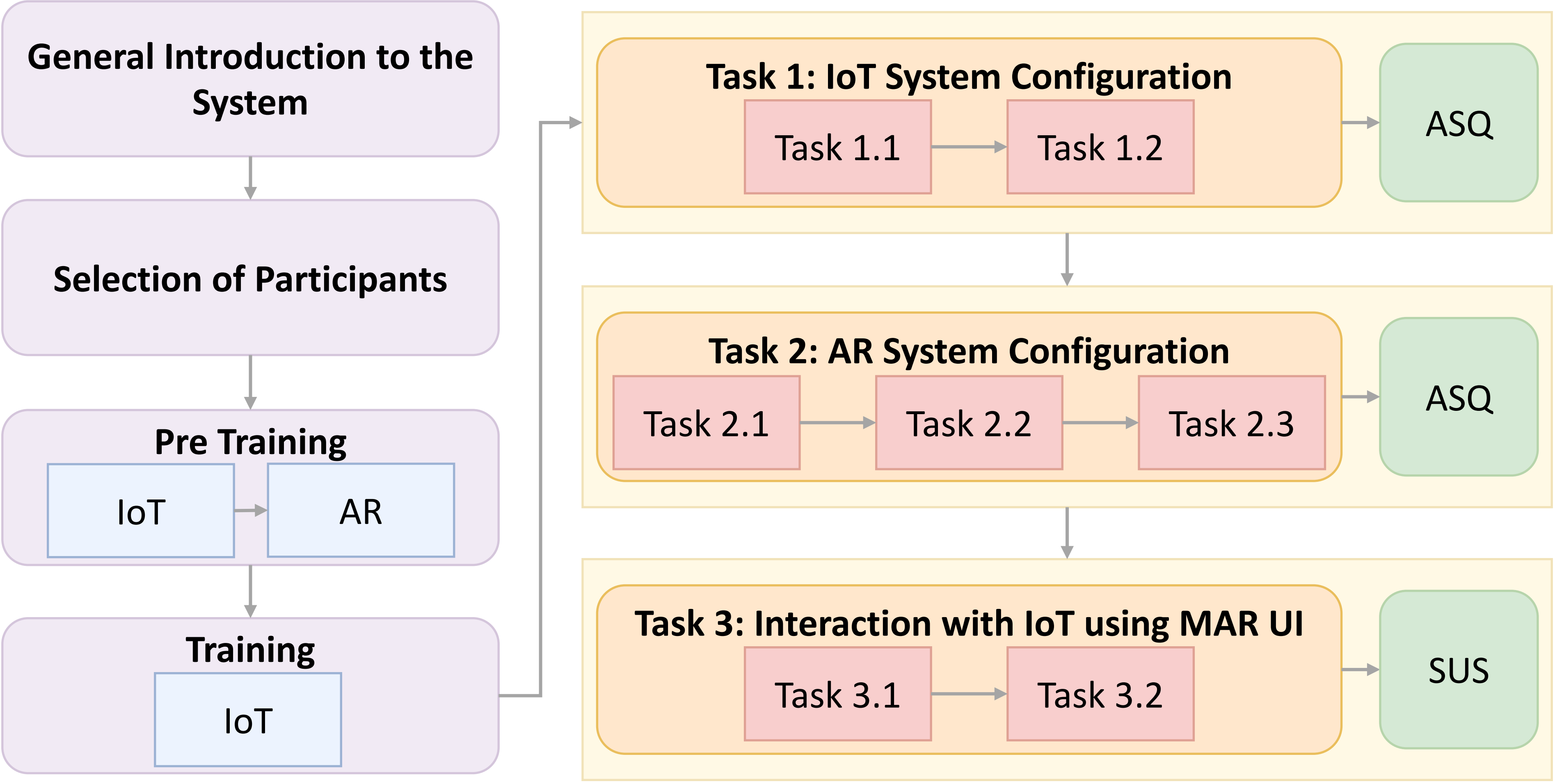
- **Changes** in template generator interface to **ensure consistency.**

- **Participant Demographics:**
  - Aged 18 – 26
  - Computer, Genetics, Textile Engineering
  - Some experience with AR, IoT and Smart Home Automation Systems

Metrics

Usability Dimension	**Evaluation Metrics**	**Units**	**Investigation Techniques**
Effectiveness	Completion Rate	Percentage (%)	Direct Observation
	Number of Errors	Number	
Efficiency	Task Completion Time	Seconds	
Satisfaction	After Scenario Questionnaire	1-5 Likert Scale	Questionnaire
	System Usability Scale		

Experiment Schema



General Introduction to the System

Selection of Participants

Pre Training
IoT → AR

Training
IoT

Task 1: IoT System Configuration
Task 1.1 → Task 1.2
ASQ

Task 2: AR System Configuration
Task 2.1 → Task 2.2 → Task 2.3
ASQ

Task 3: Interaction with IoT using MAR UI
Task 3.1 → Task 3.2
SUS

Legend

🟪	Pre-Test Stage
🟦	Training Step
🟧	Test Task
🟥	Subtask
🟩	Evaluation

Task Breakdown

- <u>Task 1:</u> IoT System Configuration
  - ***Task 1.1:*** Data Acquisition Flow
  - ***Task 1.2:*** Task Realization Flow
- <u>Task 2:</u> AR System Configuration
  - ***Task 2.1:*** Default IP Port Configuration
  - ***Task 2.2:*** Insertion of Labels
  - ***Task 2.3:*** Insertion of Buttons
- <u>Task 3:</u> Interaction with IoT Using MAR UI
  - ***Task 3.1:*** Visualizing Sensor Output
  - ***Task 3.2:*** Controlling Devices

Pre-Training and Training

IoT Pre-Training:

- Data Acquisition Video
- Task Realization Video

AR Pre-Training

- Description of how UI elements should interact with IoT elements.

IoT Training:

- Import and configure a **Data Acquisition flow.**
- Import and configure a **Task Realization flow.**

AR Training

- No training provided for AR.
- Aim to measure how **intuitive** participants found the system.

Overview of Test Tasks

Task 1) Importing and customizing two generic flows:

- **Task 1.1)** Data Acquisition flow
- **Task 1.2)** Task Realization flow

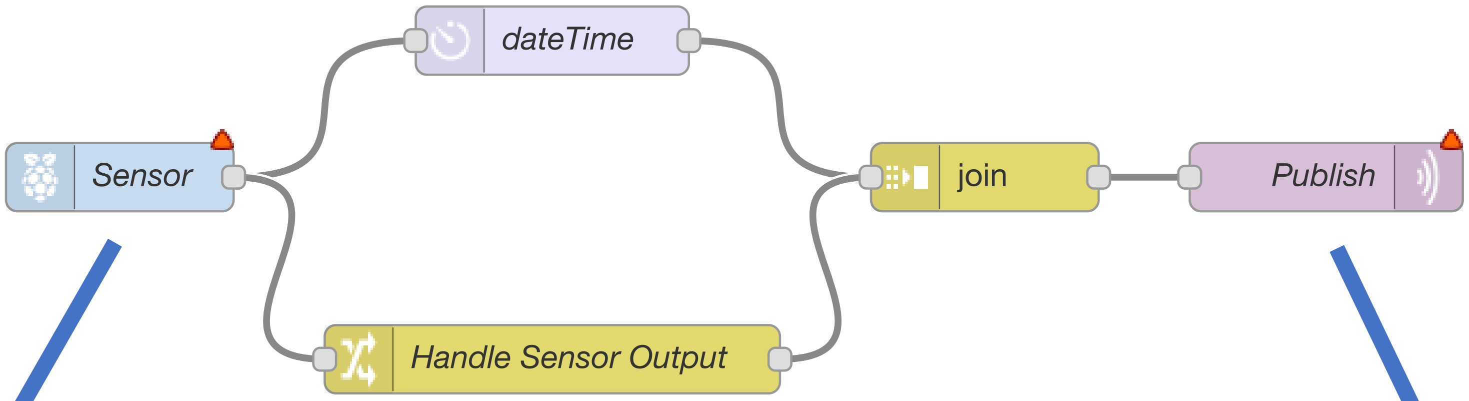
Task 2) Using the template generator to create a template of a mobile AR interface:

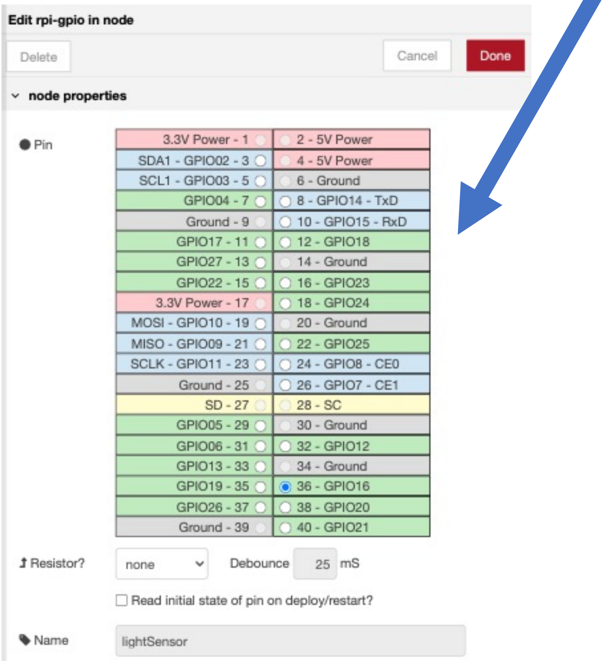
- **Task 2.1)** Configuration of server settings
- **Task 2.2)** Insertion of labels to template
- **Task 2.3)** Insertion of buttons to template

Task 3) Using the dynamically and automatically created mobile AR interface:

- **Task 3.1)** Observing sensor output
- **Task 3.2)** Controlling a device

Task 1.1





Edit rpi-gpio in node

Delete | Cancel | Done

⌄ **node properties**

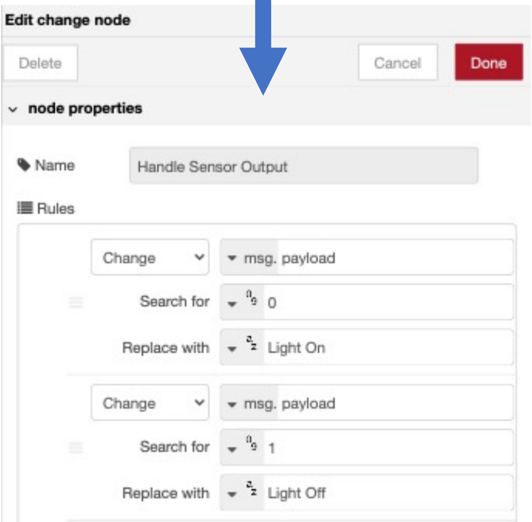
● Pin

3.3V Power - 1	2 - 5V Power
SDA1 - GPIO02 - 3 ○	○ 4 - 5V Power
SCL1 - GPIO03 - 5 ○	6 - Ground
GPIO04 - 7 ○	○ 8 - GPIO14 - TxD
Ground - 9	○ 10 - GPIO15 - RxD
GPIO17 - 11 ○	○ 12 - GPIO18
GPIO27 - 13 ○	14 - Ground
GPIO22 - 15 ○	○ 16 - GPIO23
3.3V Power - 17	○ 18 - GPIO24
MOSI - GPIO10 - 19 ○	20 - Ground
MISO - GPIO09 - 21 ○	○ 22 - GPIO25
SCLK - GPIO11 - 23 ○	○ 24 - GPIO8 - CE0
Ground - 25	○ 26 - GPIO7 - CE1
SD - 27	28 - SC
GPIO05 - 29 ○	30 - Ground
GPIO06 - 31 ○	○ 32 - GPIO12
GPIO13 - 33 ○	34 - Ground
GPIO19 - 35 ○	● 36 - GPIO16
GPIO26 - 37 ○	○ 38 - GPIO20
Ground - 39	○ 40 - GPIO21

Resistor? none   Debounce 25 mS

□ Read initial state of pin on deploy/restart?

Name: lightSensor



Edit change node

Delete | Cancel | Done

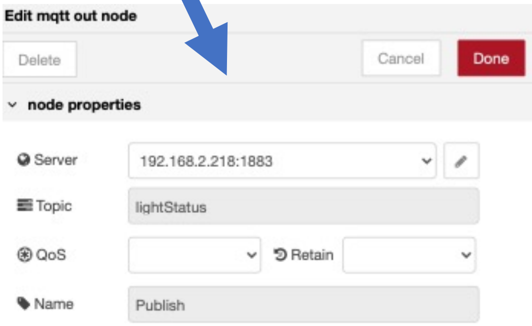
⌄ **node properties**

Name: Handle Sensor Output

Rules

Change msg. payload
Search for 0
Replace with Light On

Change msg. payload
Search for 1
Replace with Light Off



Edit mqtt out node

Delete | Cancel | Done

⌄ **node properties**

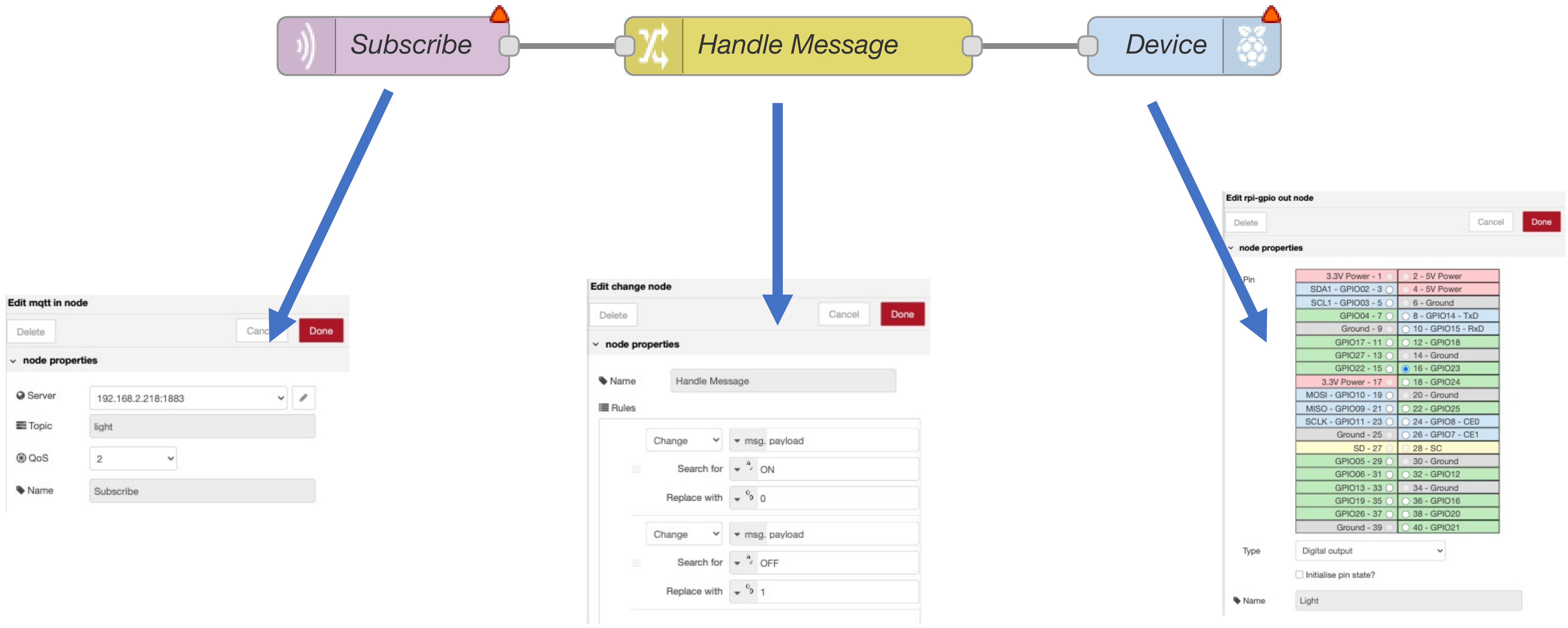
Server: 192.168.2.218:1883

Topic: lightStatus

QoS: Retain:

Name: Publish

Task 1.2



Edit mqtt in node

Delete | Cancel | Done

node properties

- Server: 192.168.2.218:1883
- Topic: light
- QoS: 2
- Name: Subscribe

Edit change node

Delete | Cancel | Done

node properties

- Name: Handle Message
- Rules:
  - Change msg. payload
    - Search for: ON
    - Replace with: 0
  - Change msg. payload
    - Search for: OFF
    - Replace with: 1

Edit rpi-gpio out node

Delete | Cancel | Done

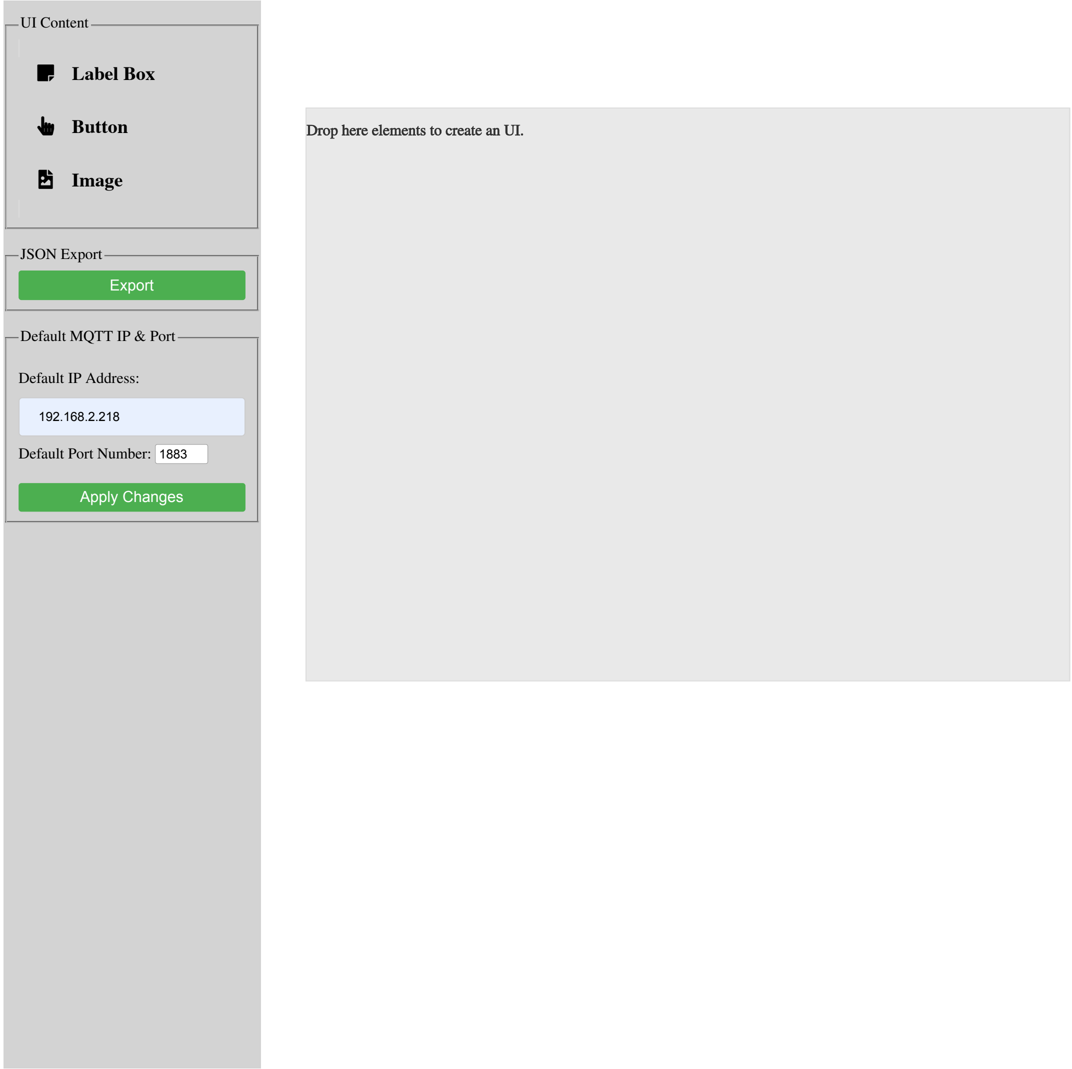
node properties

Pin

3.3V Power - 1 ○	○ 2 - 5V Power
SDA1 - GPIO02 - 3 ○	○ 4 - 5V Power
SCL1 - GPIO03 - 5 ○	○ 6 - Ground
GPIO04 - 7 ○	○ 8 - GPIO14 - TxD
Ground - 9 ○	○ 10 - GPIO15 - RxD
GPIO17 - 11 ○	○ 12 - GPIO18
GPIO27 - 13 ○	○ 14 - Ground
GPIO22 - 15 ○	● 16 - GPIO23
3.3V Power - 17 ○	○ 18 - GPIO24
MOSI - GPIO10 - 19 ○	○ 20 - Ground
MISO - GPIO09 - 21 ○	○ 22 - GPIO25
SCLK - GPIO11 - 23 ○	○ 24 - GPIO8 - CE0
Ground - 25 ○	○ 26 - GPIO7 - CE1
SD - 27 ○	○ 28 - SC
GPIO05 - 29 ○	○ 30 - Ground
GPIO06 - 31 ○	○ 32 - GPIO12
GPIO13 - 33 ○	○ 34 - Ground
GPIO19 - 35 ○	○ 36 - GPIO16
GPIO26 - 37 ○	○ 38 - GPIO20
Ground - 39 ○	○ 40 - GPIO21

- Type: Digital output
- □ Initialise pin state?
- Name: Light

Task 2.1



UI Content

JSON Export

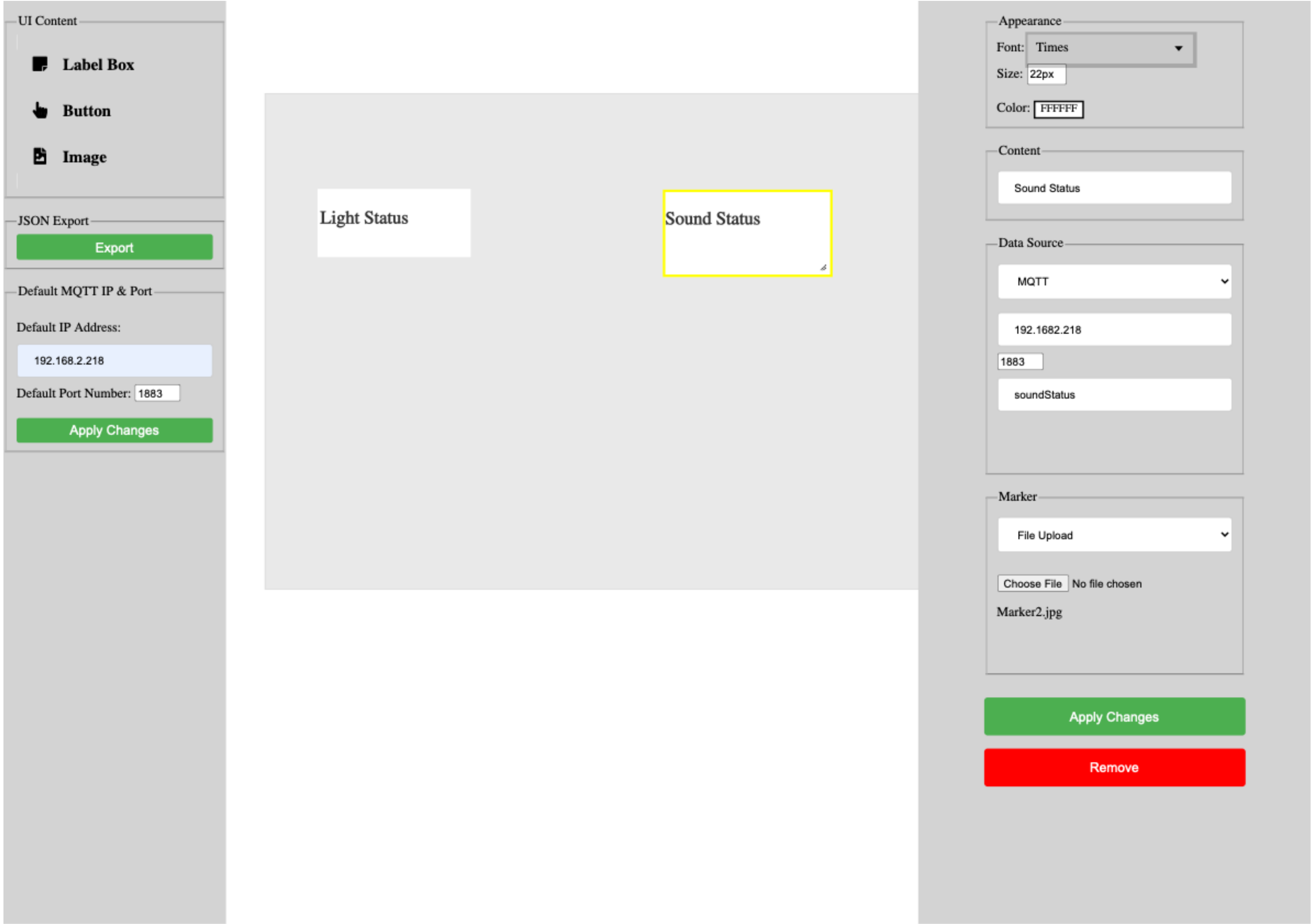
Default MQTT IP & Port

Default IP Address: 192.168.2.218

Default Port Number: 1883

Drop here elements to create an UI.

Task 2.2



Default IP Address: 192.168.2.218

Default Port Number: 1883

Font: Times

Size: 22px

Color: FFFFFF

Sound Status

MQTT

192.1682.218

1883

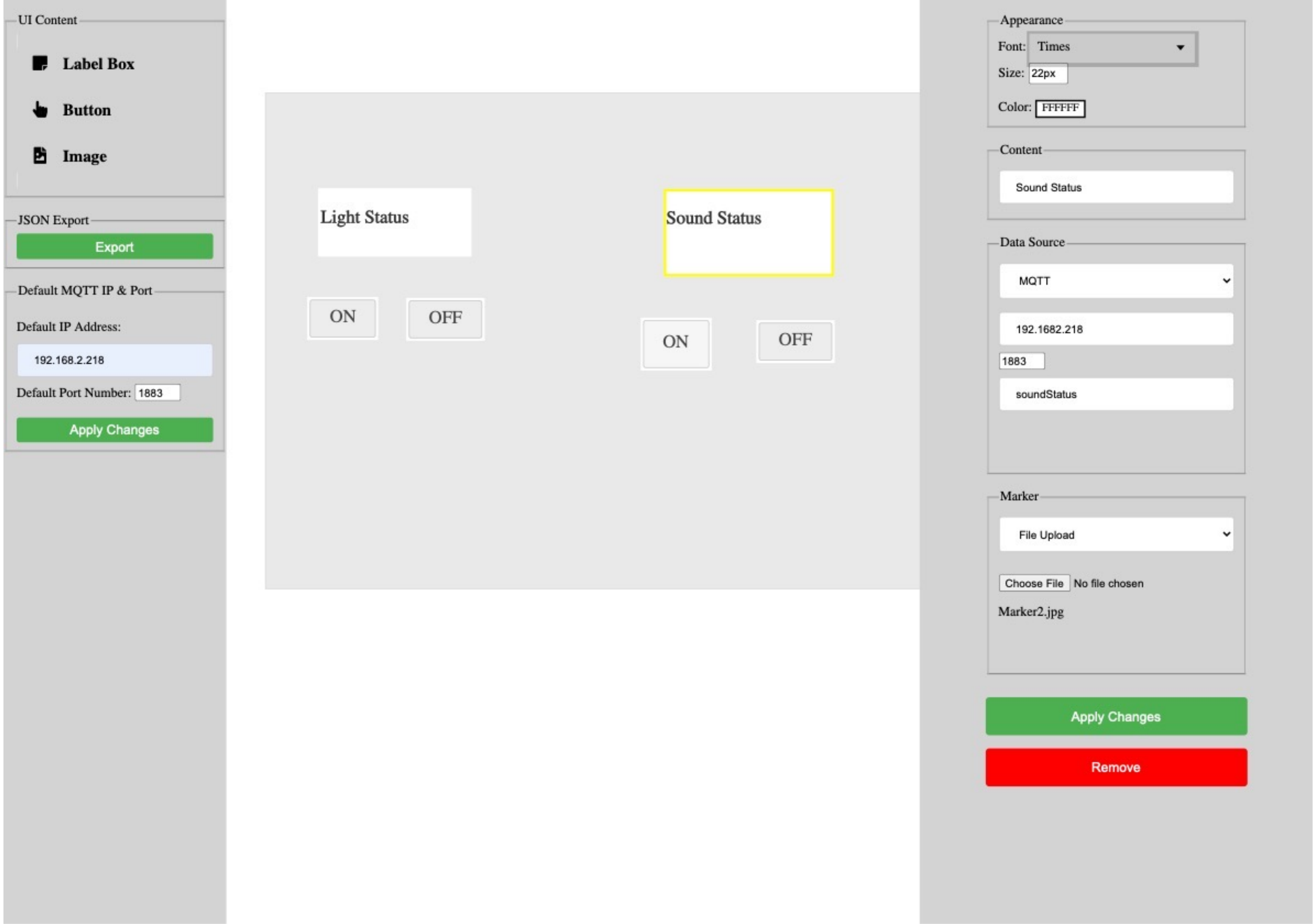
soundStatus

File Upload

No file chosen

Marker2.jpg

Task 2.3



Default MQTT IP & Port

Default IP Address: 192.168.2.218

Default Port Number: 1883

Appearance

Font: Times

Size: 22px

Color: FFFFFF

Content

Sound Status

Data Source

MQTT

192.1682.218

1883

soundStatus

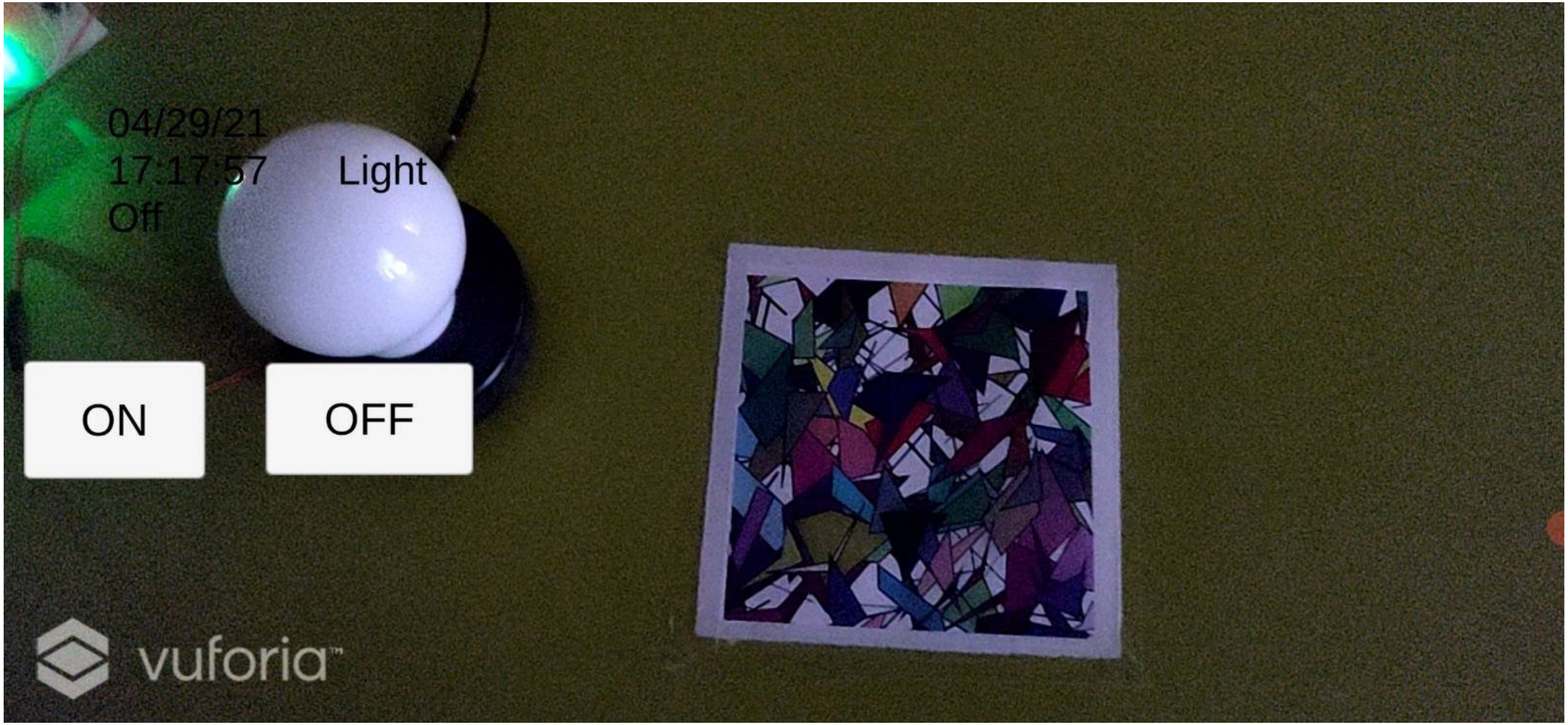
Marker

File Upload

Choose File No file chosen

Marker2.jpg

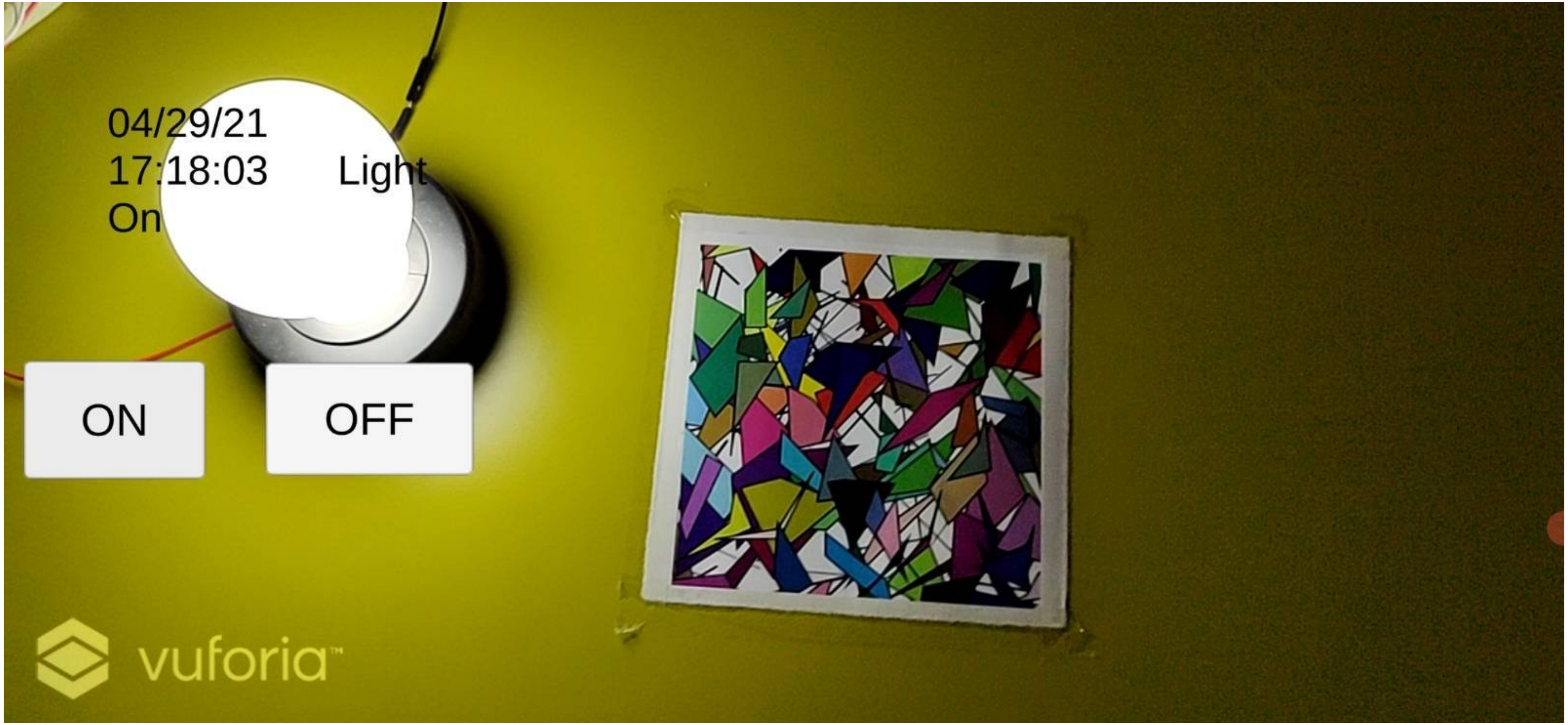
Task 3.1



04/29/21
17:17:57
Off

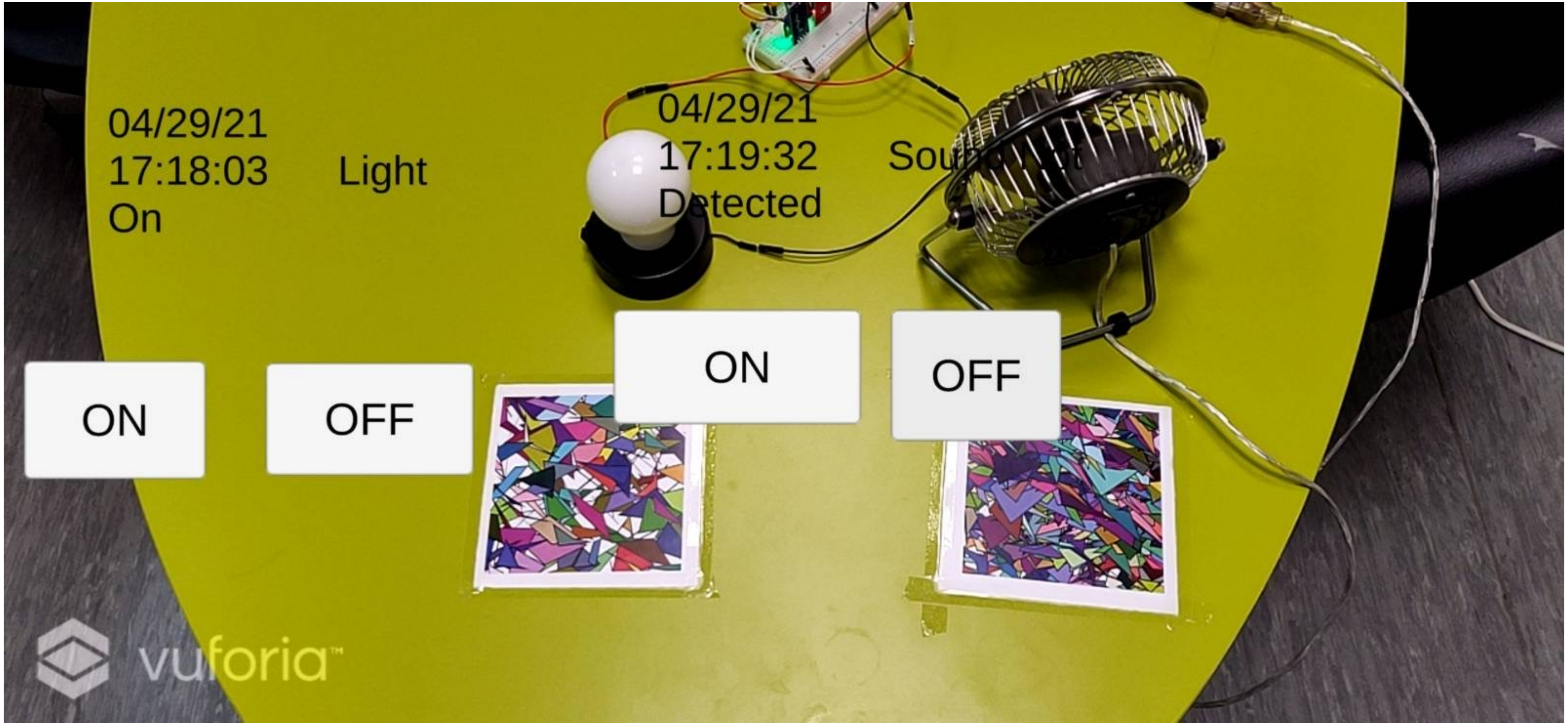
Light

Task 3.2



04/29/21
17:18:03 Light
On

Task 3



04/29/21
17:18:03 Light
On

04/29/21
17:19:32 Sound Not
Detected

Results

Metric	Task 1.1	Task 1.2	Task 2.1	Task 2.2	Task 2.3
Task Completion Rate	89%	89%	89%	67%	100%
Average Number of Errors	0.23	0.45	0.12	0.56	0.34
Average Completion Time	119.56	73.7	28.44	182.89	211.67
Given Maximum Time (sec.)	180	120	45	180	420
Average ASQ	4.82		4.74		

SUS Score	81.9 (A)

Participant Feedback

- Showed **interest** in the system.
- Inquired about using the system in their own **homes**.
- Felt **accomplished** after using the system.
- Found it **intuitive**.
- Suggested different **scenarios for the use of this system.**
- Made **suggestions to improve** the system.

Performance Assessment wrt. Background

	Completion Rate		Completion Time		Number of Errors	
Tasks	**CE**	**NCE**	**CE**	**NCE**	**CE**	**NCE**
Task 1.1	100%	75%	103.2	114.65	0	2
Task 1.2	80%	100%	76.5	56.75	4	0
Task 2.1	100%	75%	29	18	0	1
Task 2.2	80%	50%	150.25	174	2	3
Task 2.3	100%	100%	193.2	234.75	0	3

Conclusion

- An **end-to-end framework integrating AR and IoT** using open source technologies is presented.
- A **LCNC framework** which provides users with **necessary abstractions** so that they can create a **personalized smart home application** is established.
- The **usability of the framework** has been tested using an application generated with this framework.

Future Work

- Different methods of **data visualization.**
- Different methods of **context awareness.**
- Different **devices for AR** (head-mounted gear).
- Seamless **communication with Node-RED** to reduce cognitive load.

Thank you for watching.

MARIoT: An Authoring Framework for Creating IoT Applications with Mobile Augmented Reality

Meral Kuyucu, Gökhan İnce

User Experience Laboratory
Department of Computer Engineering
Istanbul Technical University (Turkey)

Contact email: **korkmazmer@itu.edu.tr**