



SMART PET FEEDER



LE SSERAFIM



GROUP MEMBERS

1



Pinpimarn Pinpisit
6410545541

2



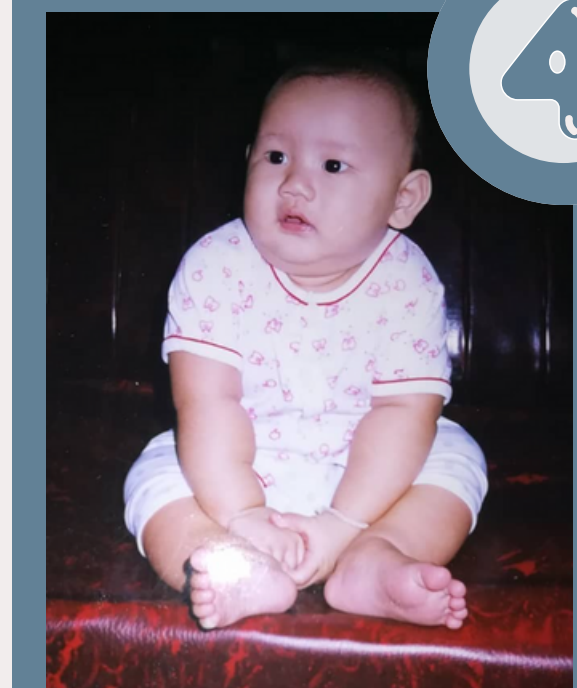
Jiratchaya Phinyodom
6410546131

3



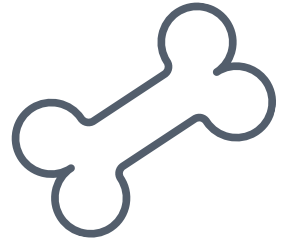
Preawpan Thamapipol
6410546203

4



Thanadol Udomsirinanchai
6410546165

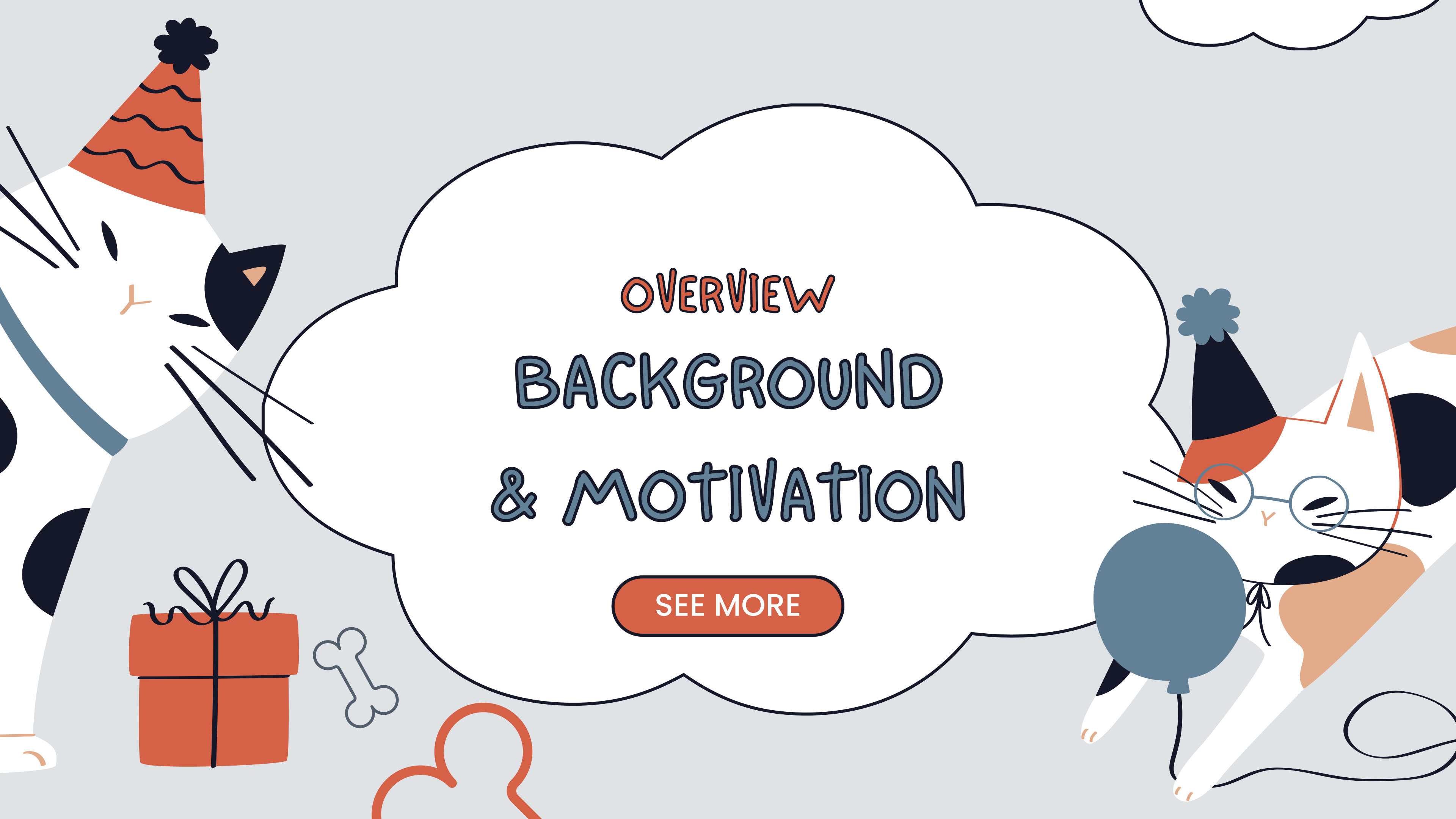




AGENDA

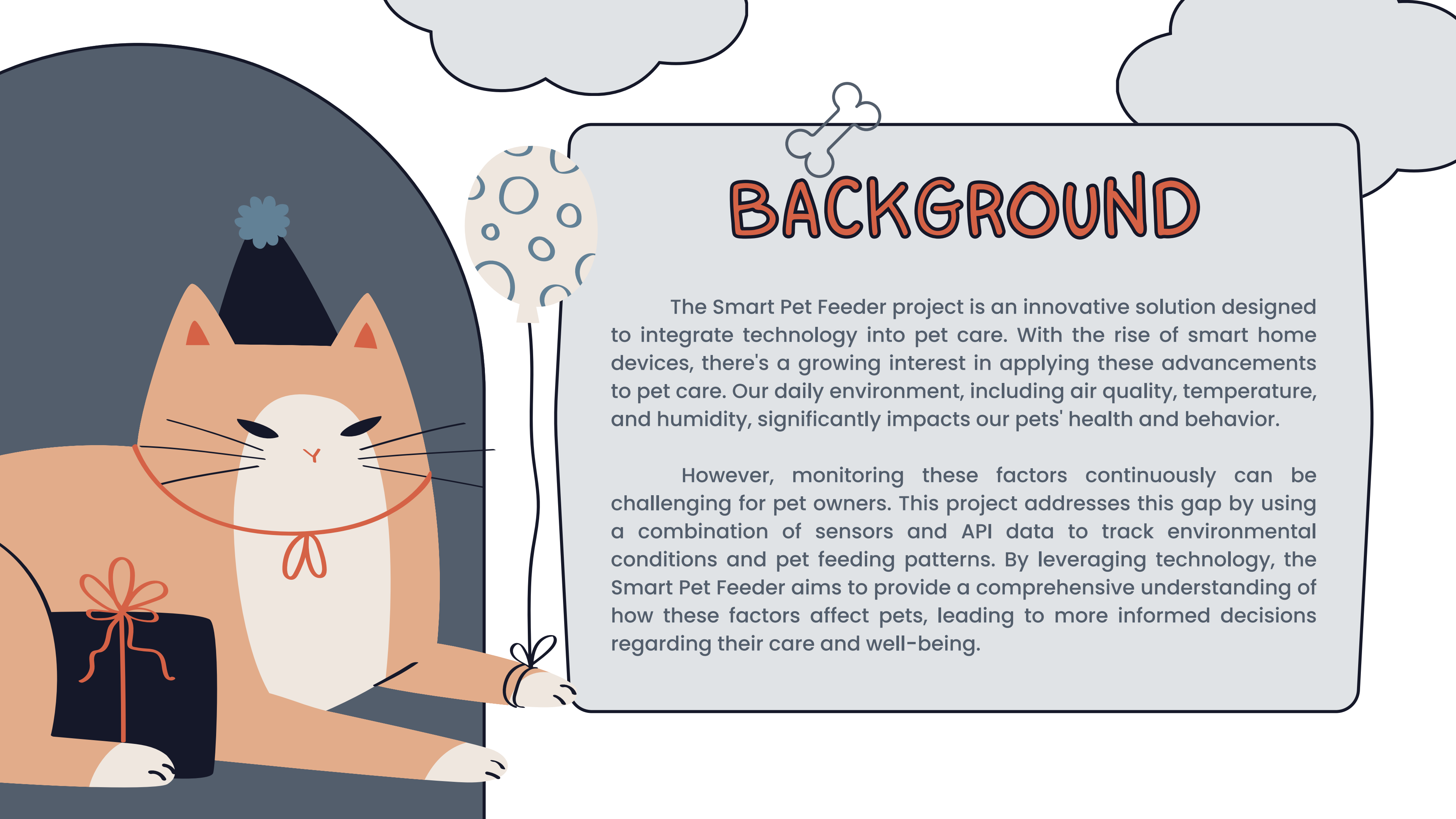
SMART PET FEEDER

- Overview: background, motivation, etc
- Overall architecture
- Data sources (primary and secondary) and collection mechanisms
- Database schema used for data integration
- Data sharing API
- Data visualization
- Demonstration of key features



OVERVIEW BACKGROUND & MOTIVATION

SEE MORE



BACKGROUND

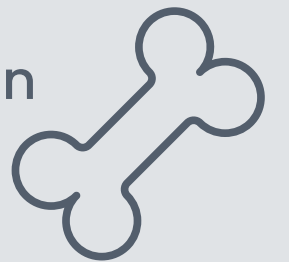
The Smart Pet Feeder project is an innovative solution designed to integrate technology into pet care. With the rise of smart home devices, there's a growing interest in applying these advancements to pet care. Our daily environment, including air quality, temperature, and humidity, significantly impacts our pets' health and behavior.

However, monitoring these factors continuously can be challenging for pet owners. This project addresses this gap by using a combination of sensors and API data to track environmental conditions and pet feeding patterns. By leveraging technology, the Smart Pet Feeder aims to provide a comprehensive understanding of how these factors affect pets, leading to more informed decisions regarding their care and well-being.

MOTIVATION

The motivation behind the Smart Pet Feeder project stems from the desire to enhance pet care using technology. Pets are beloved family members, and their health and happiness are paramount to their owners. However, busy lifestyles can make it difficult to monitor and adjust their environment constantly.

This project is driven by the idea that a better understanding of environmental factors and their impact on pets can lead to improved care. By offering insights into how air quality, temperature, humidity, and food conditions affect pets' eating habits and health, the Smart Pet Feeder empowers pet owners to make more informed decisions. This project aims to bridge the gap between technology and pet care, ultimately improving the quality of life for pets and their owners.



QUESTIONS

Is the pet's environment safe?

How does air quality affect the pet's appetite?

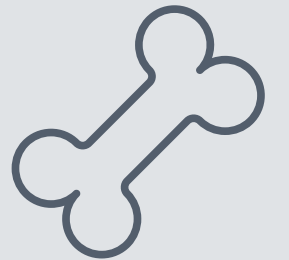
How does environmental data impact food preservation?

Are there any relations between food humidity and a pet's appetite?

Should pet owners adjust the feeding schedule to optimize pet feeding habits?

Are there any relations between environmental conditions and a pet's health?

QUESTIONS

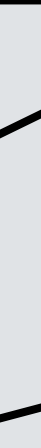
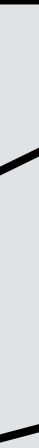
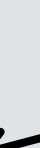
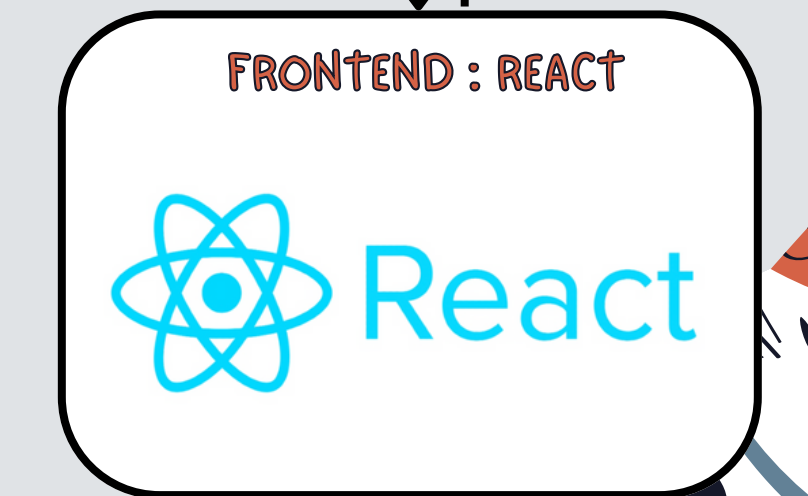
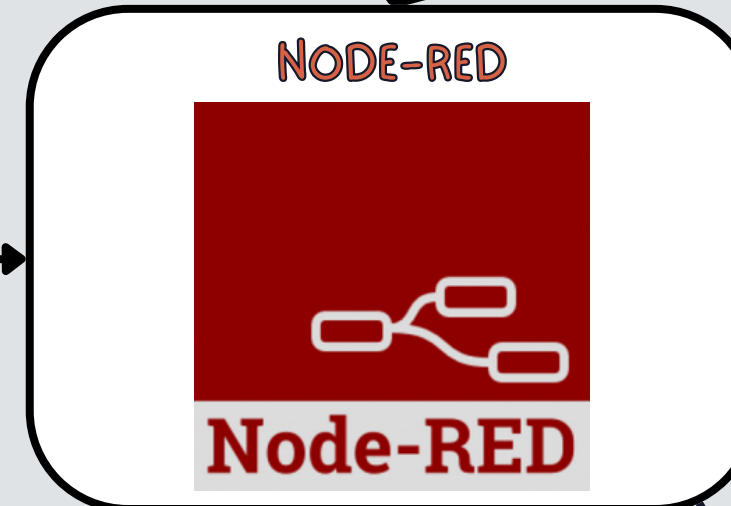
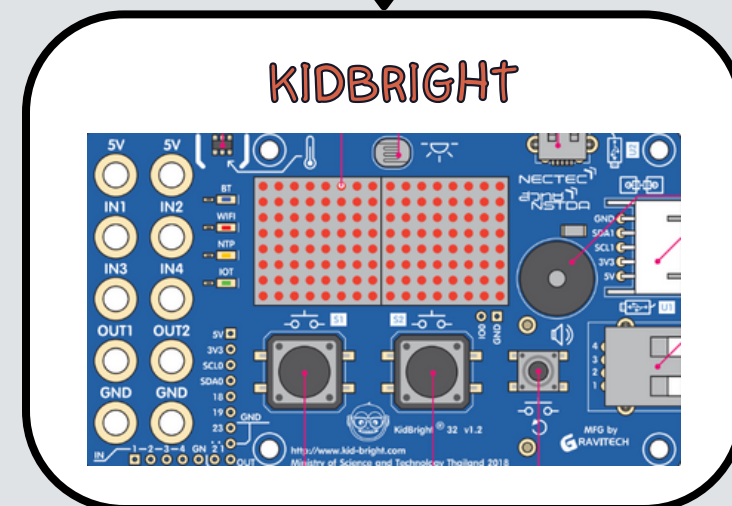
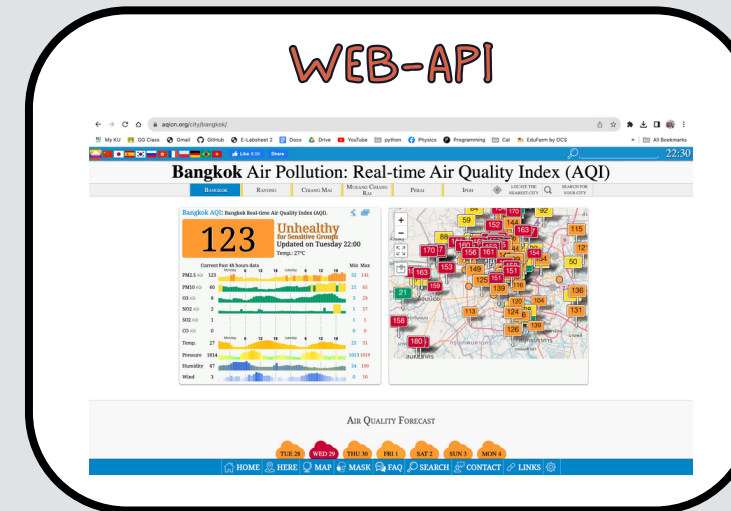
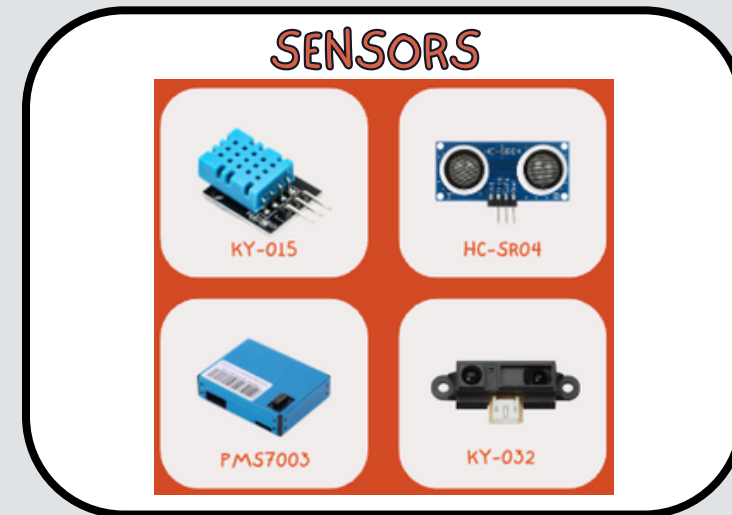




OVERALL ARCHITECHTURE

SEE MORE

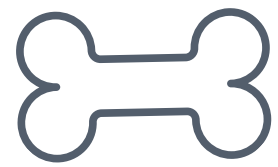
OVERALL ARCHITECHTURE





DATA SOURCES AND COLLECTION MECHNISMS

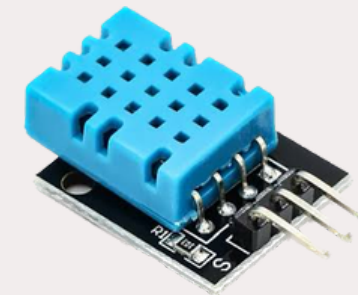
[SEE MORE](#)



DATA SOURCES

PRIMARY DATA

- Temperature and humidity sensor
 - Measure room's current temperature and humidity(1)
 - Measure food humidity(1)
- Ultrasonic (Ultrasonic sensor)
 - Measure remaining pet food
- Air quality sensor(Dust Sensor)
 - Measure room's air quality
- Infrared Obstacle Detector
 - Detect pet around the pet feeder area (to check that pet is eating)



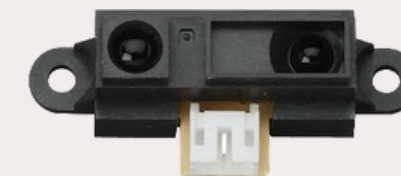
KY-015



HC-SR04



PMS7003



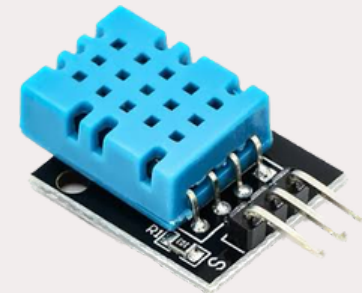
KY-032



COLLECTION MECHANISM

TEMP & HUM SENSOR

Collect data every 30 min



KY-015



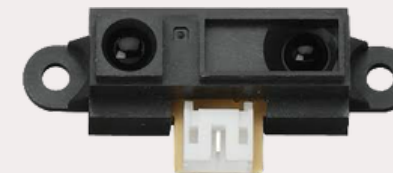
HC-SR04

ULTRASONIC SENSOR

Collect data every 30 min



PMS7003



DX-GP3V

DUST SENSOR

Collect data every 30 min

INFRARED SENSOR

Collect data when pets comes in front of the sensor for more than 15 secs
(prevent of collecting data from pet just walking by)

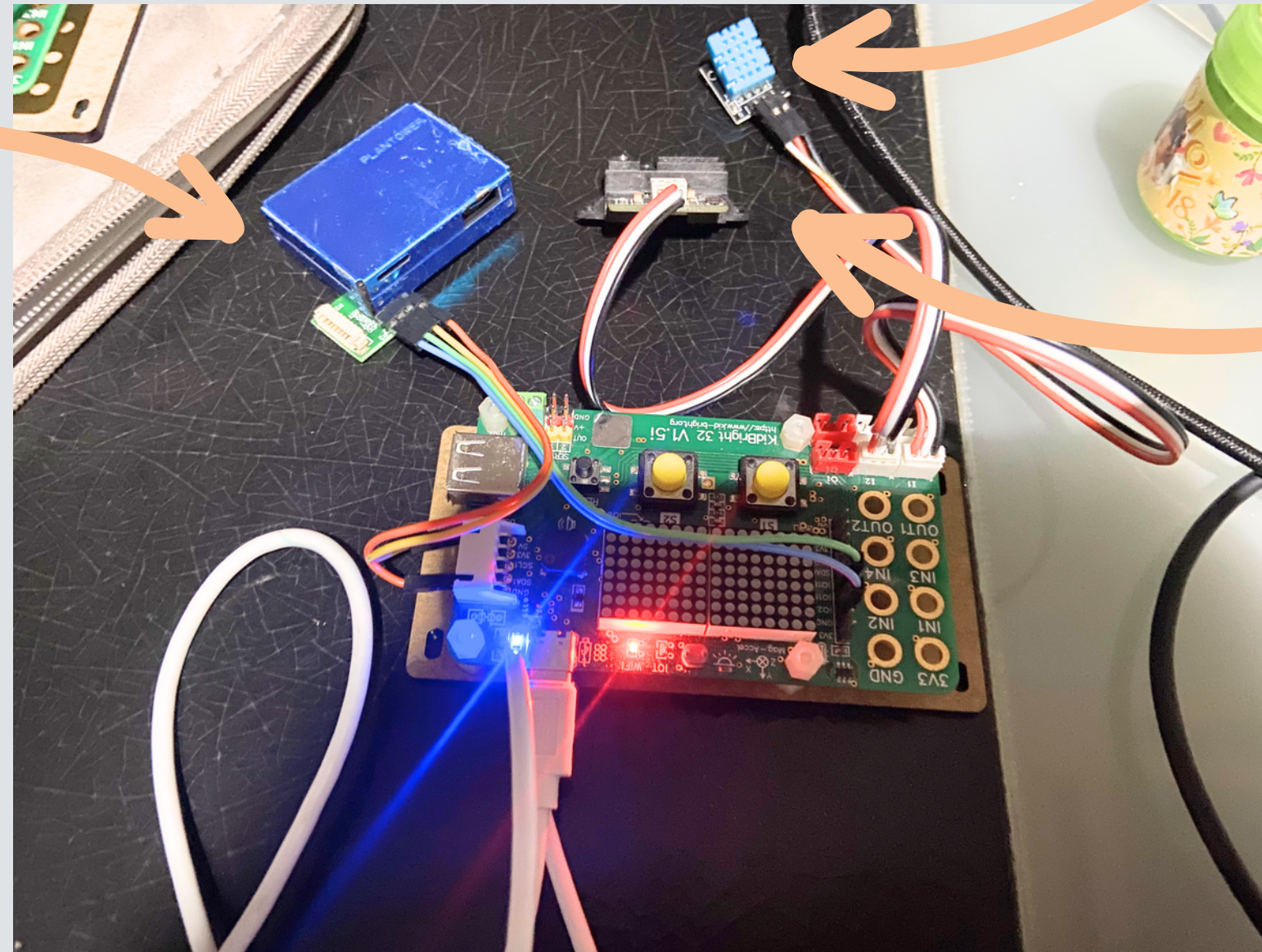


COLLECTION MECHANISM

DUST
SENSOR

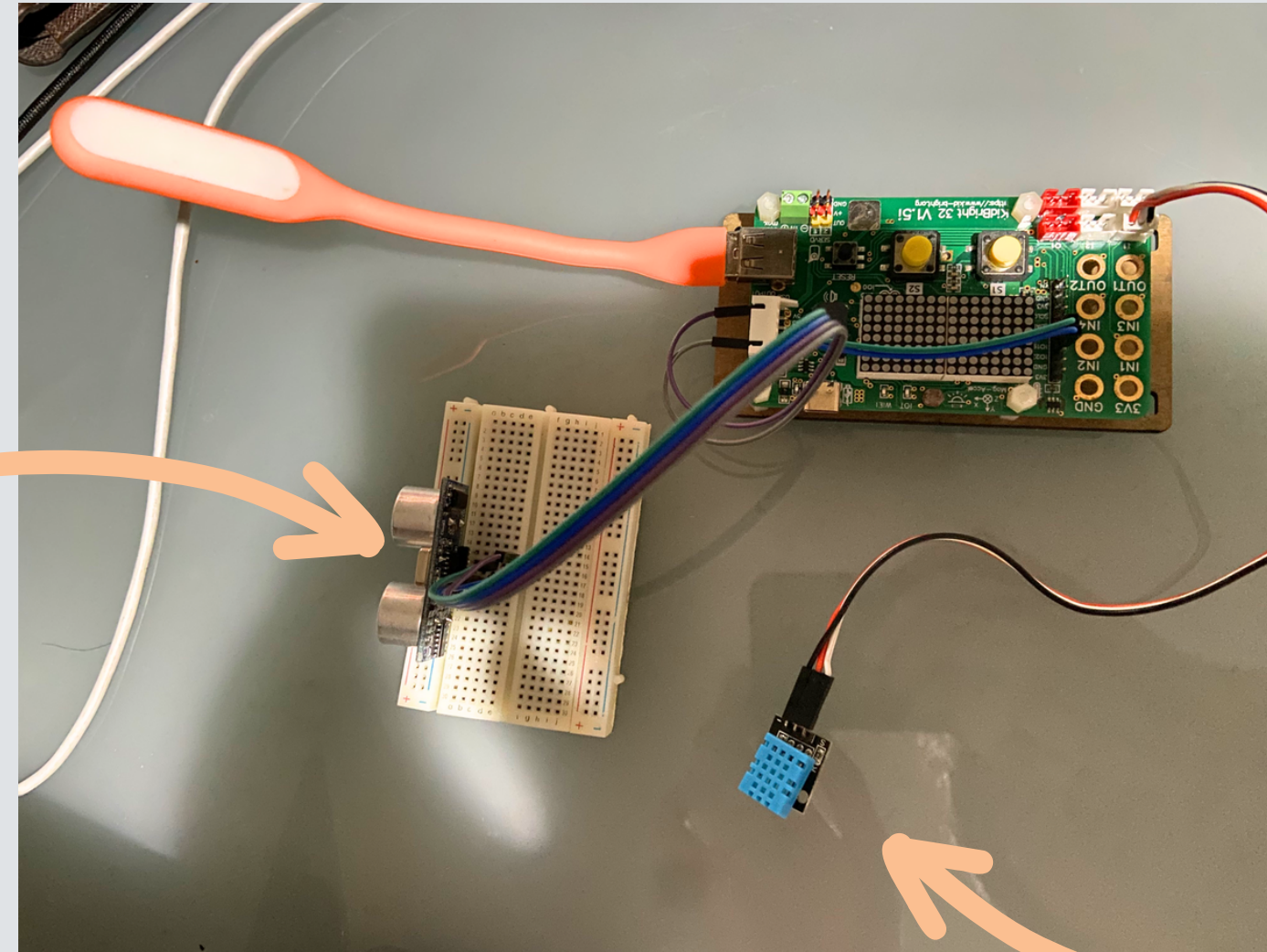
TEMP & HUM
SENSOR

INFRARED
SENSOR

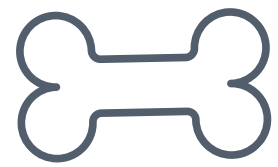


COLLECTION MECHANISM

ULTRASONIC
SENSOR



TEMP & HUM
SENSOR



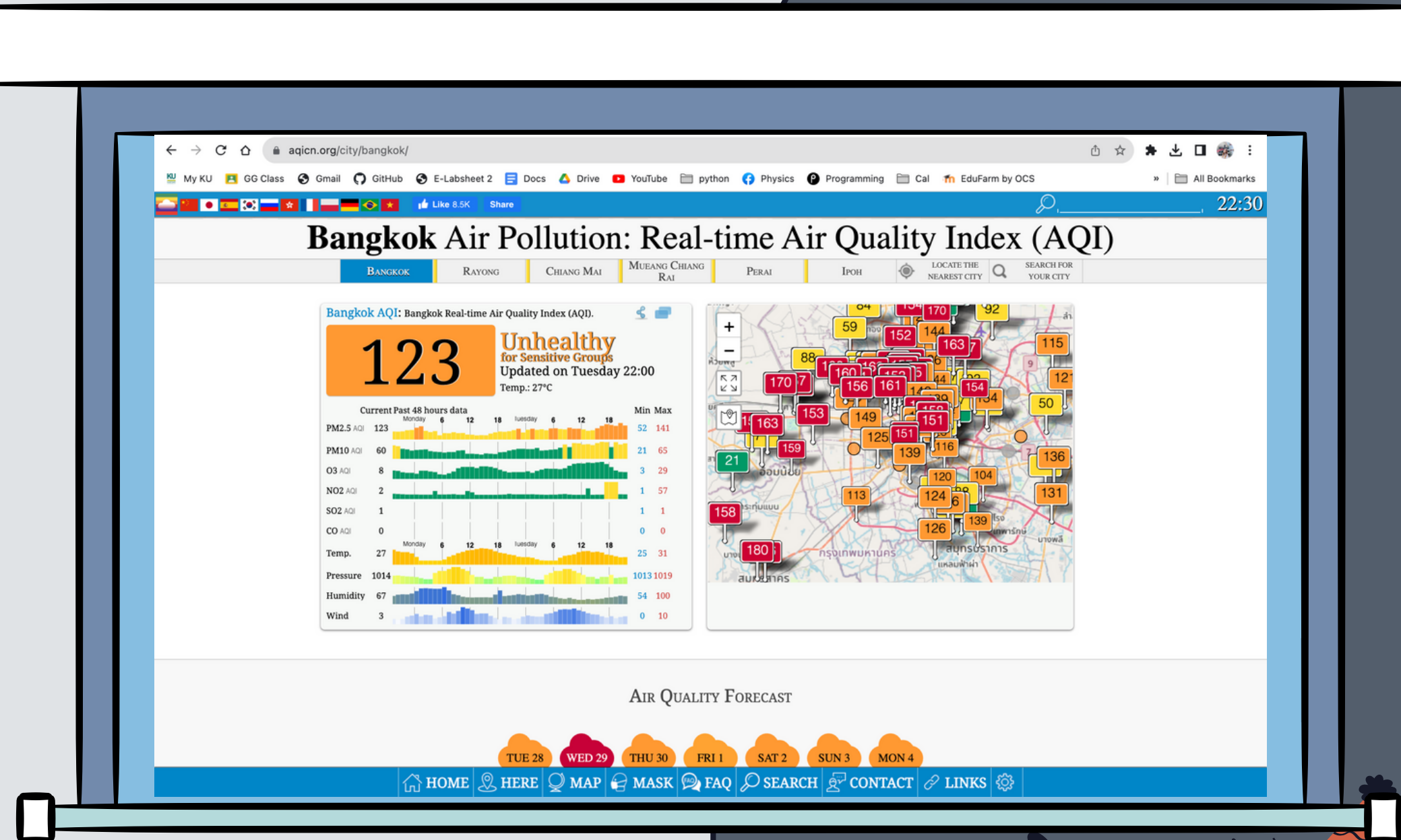
DATA SOURCES

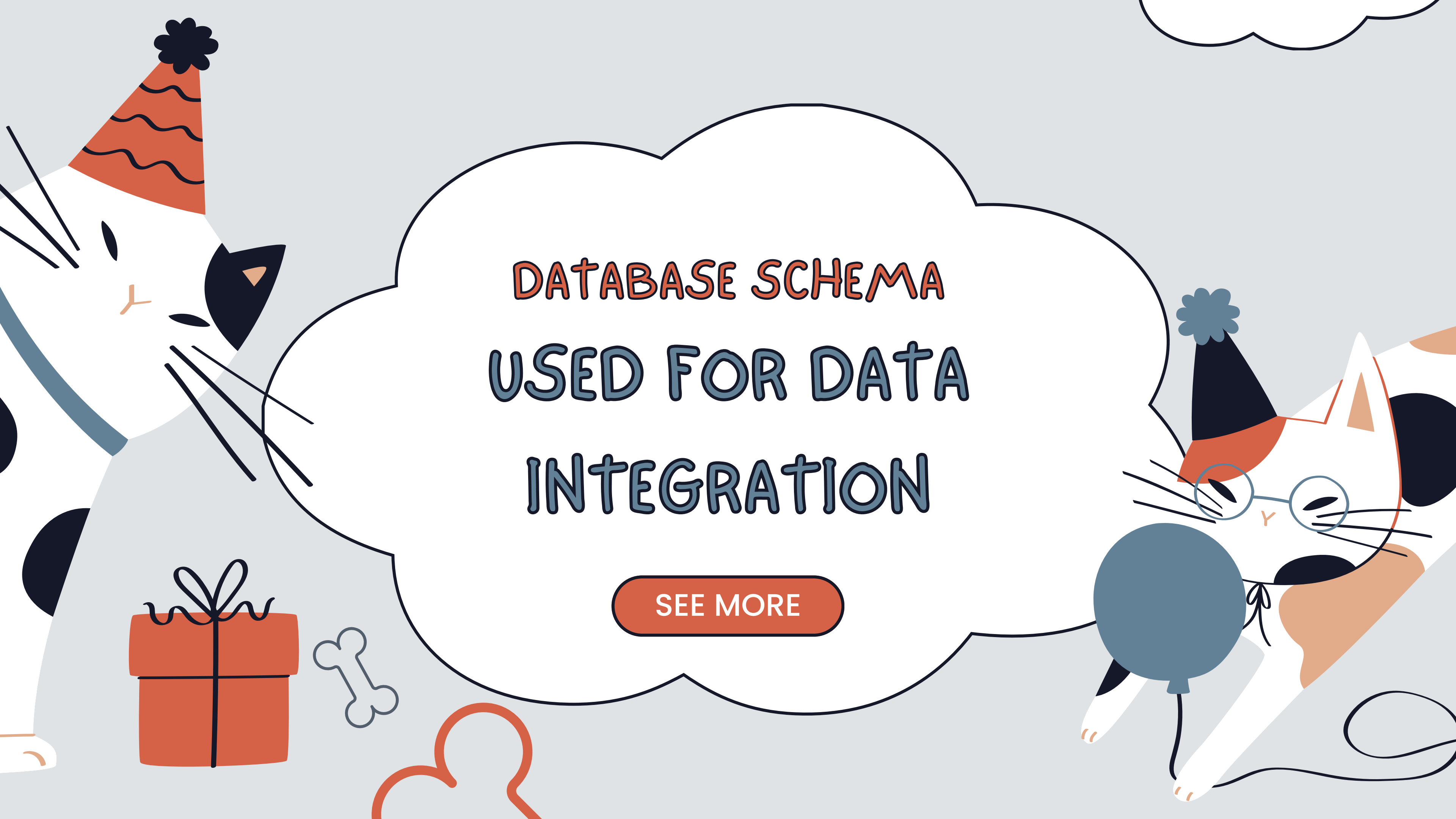
SECONDARY DATA

Local Air Quality and Temperature: Use an air quality and temperature monitoring API to obtain real-time air quality data for your area. This can help users decide whether it's safe to open a window or if an air purifier should be activated when the pet eats.

Data source link

This API fetches humidity, temperature, and PM2.5 data every hour.





DATABASE SCHEMA USED FOR DATA INTEGRATION

[SEE MORE](#)

GROUP 4 DATABASE

SensorRoomData


Collects temperature, humidity, and PM2.5 data within the room.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 	int(11)			No	None		AUTO_INCREMENT
2	ts	timestamp			No	CURRENT_TIMESTAMP		DEFAULT_GENERATED
3	temp	float			No	None		
4	hum	float			No	None		
5	pm	int(11)			No	None		

GROUP 4 DATABASE

sensorTankData


Collects temperature and humidity within the tank.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 	int(11)			No	None		AUTO_INCREMENT
2	ts	timestamp			No	CURRENT_TIMESTAMP		DEFAULT_GENERATED
3	temp	float			No	None		
4	hum	float			No	None		

GROUP 4 DATABASE

envData

Collects temperature, humidity, and PM2.5 data within the environment.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 	int(11)			No	None		AUTO_INCREMENT
2	ts	timestamp			No	CURRENT_TIMESTAMP		DEFAULT_GENERATED
3	pm	int(11)			No	None		
4	hum	float			No	None		
5	temp	float			No	None		

GROUP 4 DATABASE

dataVisual

Collects temperature, humidity, PM2.5 and eating_time data from sensorRoomData database for using to display data visualization.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 🗝️	int(11)			No	None		AUTO_INCREMENT
2	pm25	int(255)			No	None		
3	temp	float			No	None		
4	hum	int(11)			No	None		
5	eating_time	int(11)			No	None		



GROUP 4 DATABASE

mealPlan


Collects days, time, portion, and enable_status from the user that they want to feed the food to their pet.

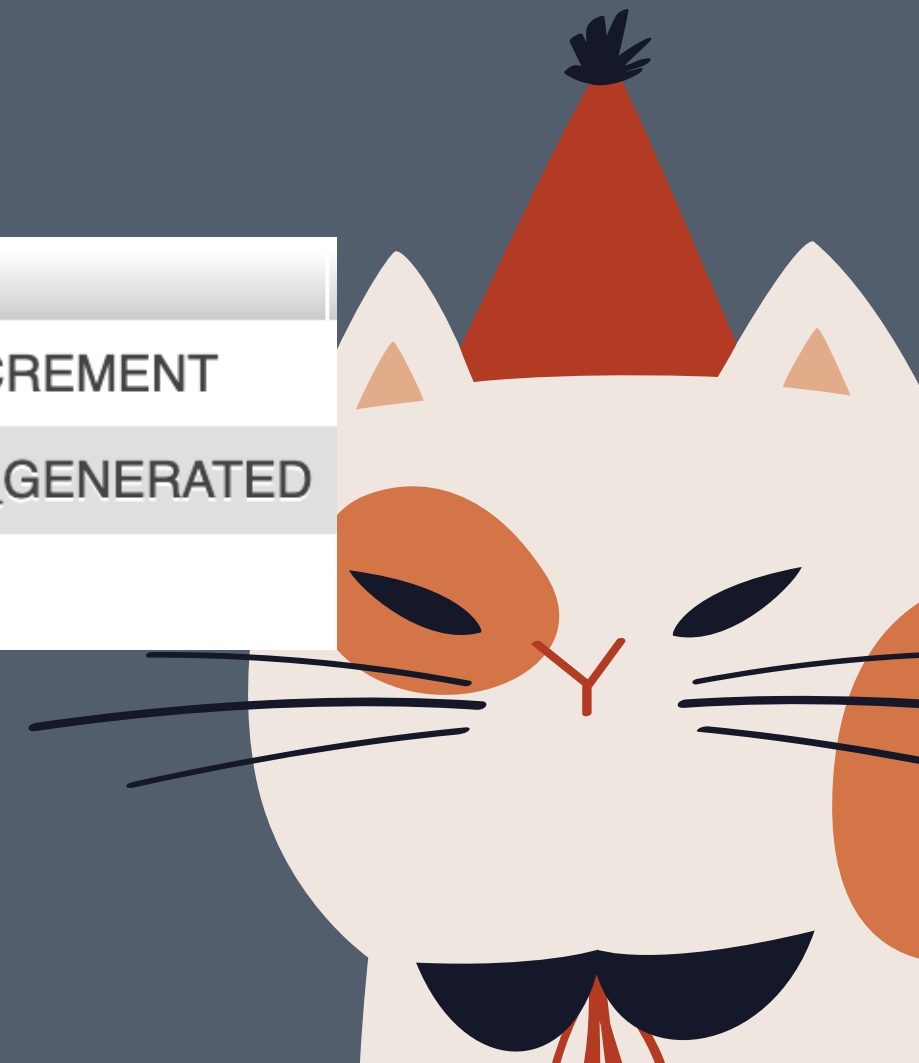
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	schedule_id 🔑	int(11)			No	None		
2	ts	timestamp			No	CURRENT_TIMESTAMP		DEFAULT_GENERATED
3	days	varchar(255)	utf8_general_ci		No	None		
4	time	varchar(255)	utf8_general_ci		No	None		
5	por	int(11)			No	None		
6	enable_status	tinyint(1)			No	None		

GROUP 4 DATABASE

feedingData

Collects history portion given to pet


#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 	int(11)			No	None		AUTO_INCREMENT
2	ts	timestamp			No	CURRENT_TIMESTAMP		DEFAULT_GENERATED
3	por	int(11)			No	None		

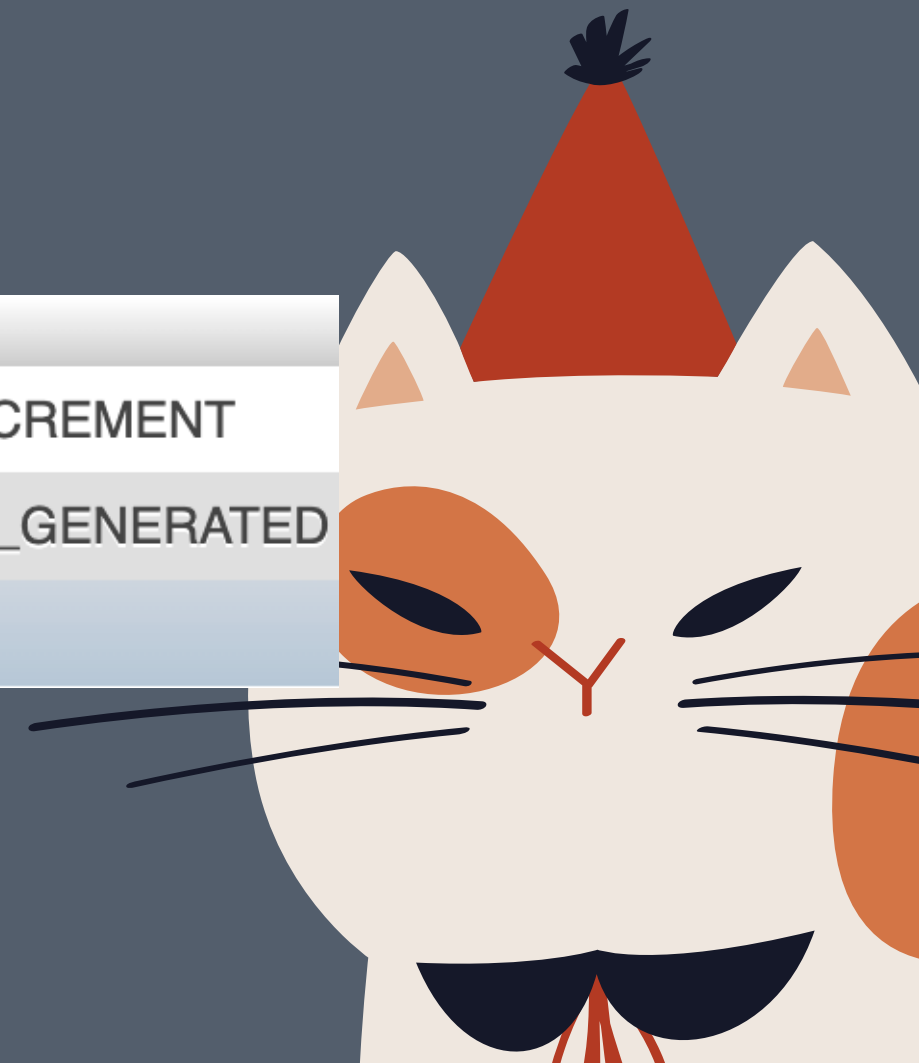


GROUP 4 DATABASE

behaviorData

Collects timestamp and status when pet are coming to eat food. If your pet is standing in front of the sensor, its status will be 1. If your pet walks away or is not present, its status will be 0.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 	int(11)			No	None		AUTO_INCREMENT
2	ts	timestamp			No	CURRENT_TIMESTAMP		DEFAULT_GENERATED
3	status	int(1)			Yes	NULL		

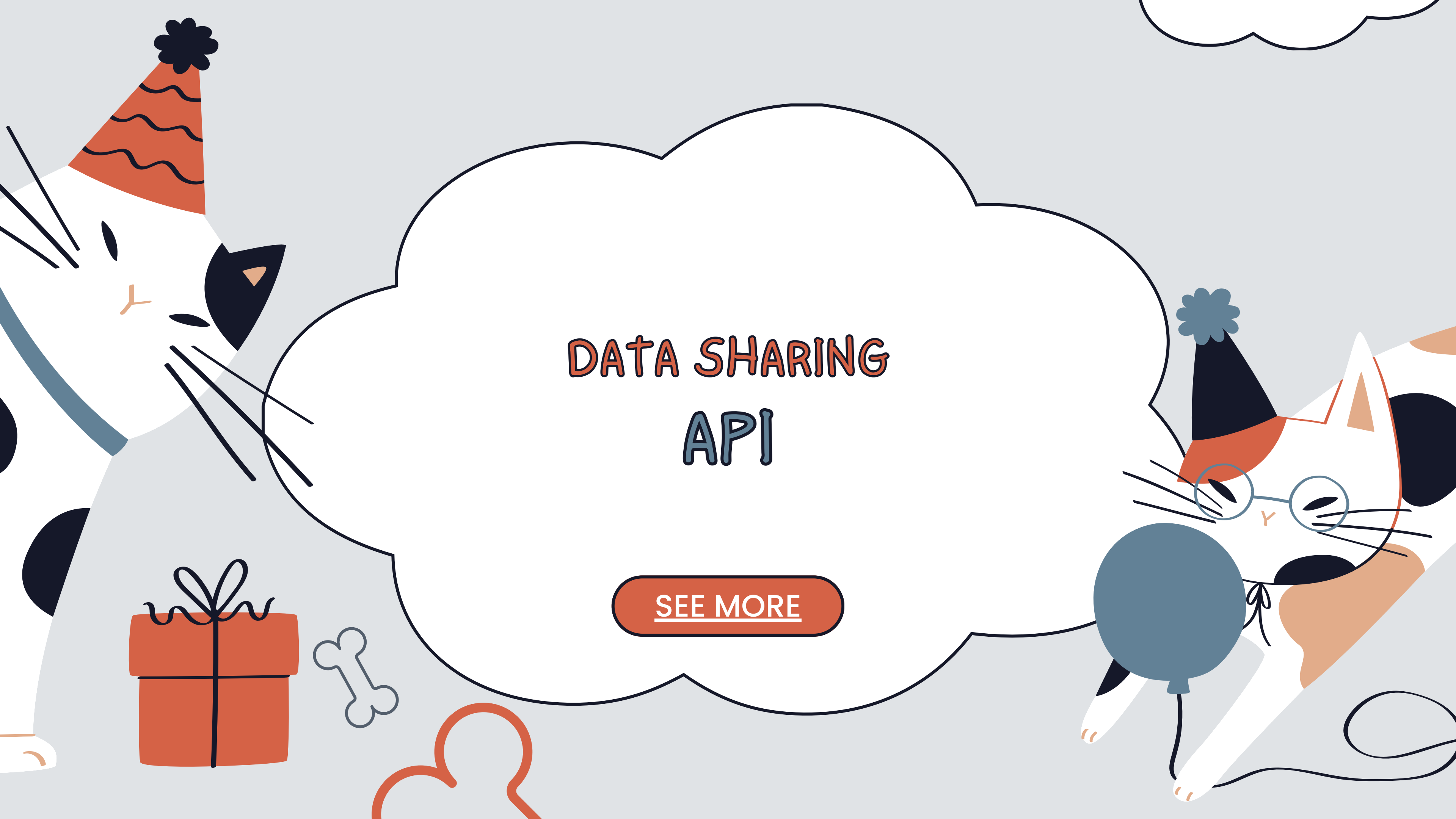


GROUP 4 DATABASE

TankData

Collect data when the machine is feeding. If the feeder is running, it will display a value of 1, but if it is stopped, it will display a value of 0, and the remaining amount of food in the tank will be calculated as a percent.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 🔑	int(11)			No	None		AUTO_INCREMENT
2	remaining_percentage	int(11)			No	None		
3	feeding_status	tinyint(4)			No	None		



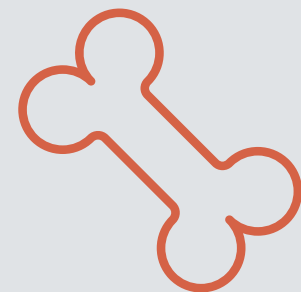
DATA SHARING API

[SEE MORE](#)

GET

/get-behavior-data

Returns a list of behavior data that contains
timestamp and status



Description

Success

Media type

application/json ▼

Controls Accept header.

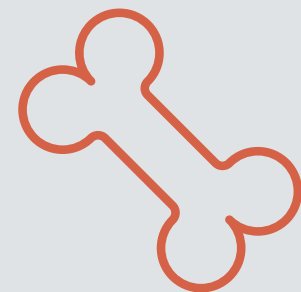
Example Value | Schema

```
[  
  {  
    "id": 0,  
    "por": 0,  
    "ts": "string"  
  }  
]
```

GET

/get-feeding-history

Returns a list of feeding history that contains id, timestamp and portion.



Description

Success

Media type

application/json ▼

Controls Accept header.

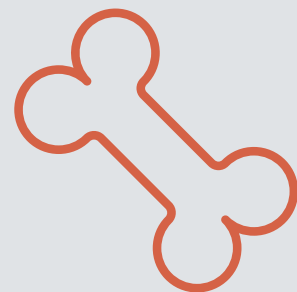
Example Value | Schema

```
[  
  {  
    "id": 0,  
    "por": 0,  
    "ts": "string"  
  }  
]
```

GET

/get-meal-plan-data

Get list of all meal plan data



Description

Successful response

Media type

application/json



Controls Accept header.

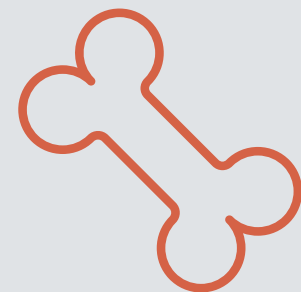
Example Value | Schema

```
[
  {
    "days": [
      "string"
    ],
    "enable_status": 0,
    "por": 0,
    "schedule_id": 0,
    "time": "string"
  }
]
```


GET

/get-tank-data

Send feeding_status and remaining_percentage



Description

Success

Media type

application/json ▼

Controls Accept header.

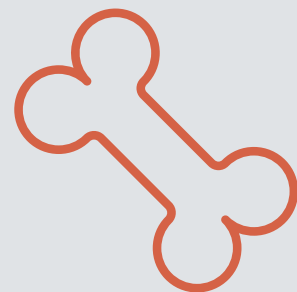
Example Value | Schema

```
[  
  {  
    "feeding_status": 0,  
    "id": 0,  
    "remaining_percentage": 0  
  }  
]
```

POST

/post-meal-plan-data

Receive the data on portion of pet food
and insert into the feedingData database



Request body **required**

Data to be sent in the request body

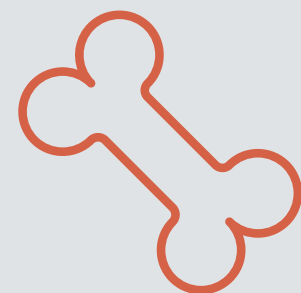
Example Value | Schema

```
{  
  "days": [  
    "string"  
  ],  
  "enable_status": 0,  
  "por": 0,  
  "schedule_id": 0,  
  "time": "string"  
}
```

POST

/post-portion-data

Receive the data on portion of pet food
and insert into the feedingData database



Request body required

Data to be sent in the request body

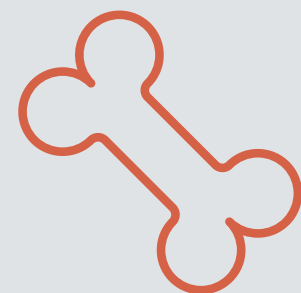
Example Value | Schema

```
{  
  "por": 0  
}
```

POST

/post-tank-data

Receive the data which are feeding_status
and remaining_percentage



Request body required

Data to be sent in the request body

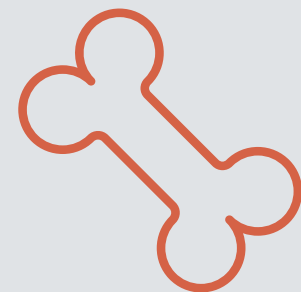
Example Value | **Schema**

```
{  
  "feeding_status": 0,  
  "remaining_percentage": 0  
}
```


GET

/roomdata

Returns a list of room data.

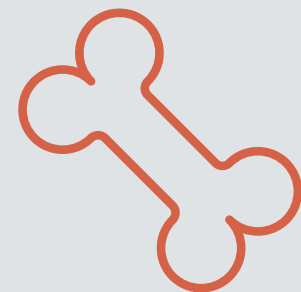


e	Description
	Success
	<div>Media type</div> <div>application/json</div> <div>Controls Accept header.</div>
Example Value	Schema
<pre>[{ "dis": 0, "id": 0, "por": 0, "ts": "string" }]</pre>	

GET

/data

Receive the data of room, tank, and env data



Description

Success

Media type

application/json

Controls Accept header.

Example Value | Schema

```
[
  {
    "env_hum": 0,
    "env_pm": 0,
    "env_temp": 0,
    "room_hum": 0,
    "room_pm": 0,
    "room_temp": 0,
    "tank_hum": 0,
    "tank_temp": 0
  }
]
```

GET

/visual/food-tank-data

Receive maximum and minimum of temperature
and average of humidity of 7 days



Description

Success

Media type

application/json

Controls Accept header.

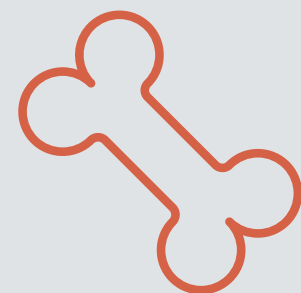
Example Value | Schema

```
[
  {
    "date": "string",
    "hum_avg": 0,
    "temp_max": 0,
    "temp_min": 0
  }
]
```

GET

/visual/hum

Receive humidity and eating time of 7 days



Code

Description

200

Success

Media type

application/json



Controls Accept header.

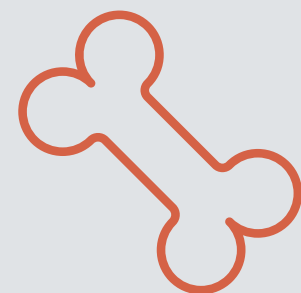
Example Value | Schema

```
[
  {
    "eating_time": [
      0
    ],
    "hum": 0
  }
]
```


GET

/visual/pm25

Receive pm25 and eating time of 7 days



Description

Success

Media type

application/json



Controls Accept header.

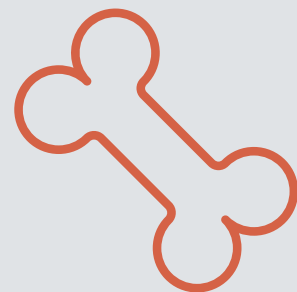
Example Value | Schema

```
[
  {
    "eating_time": [
      0
    ],
    "pm25": 0
  }
]
```

GET

/visual/room-data

Receive maximum and minimum of temperature and PM of 7 days from room data and receive average of humidity from room data of 7 days



Description

Success

Media type

application/json

Controls Accept header.

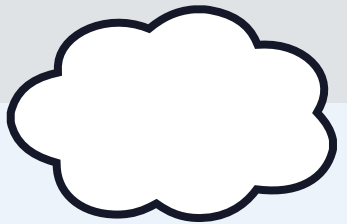
Example Value | Schema

```
[
  {
    "date": "string",
    "hum_avg": 0,
    "pm_max": 0,
    "pm_min": 0,
    "temp_max": 0,
    "temp_min": 0
  }
]
```

GET

/visual/temp

Receive temperature and eating time of 7 days



Description

Success

Media type

application/json

Controls Accept header.

Example Value | Schema

```
[
  {
    "eating_time": [
      0
    ],
    "temp": 0
  }
]
```

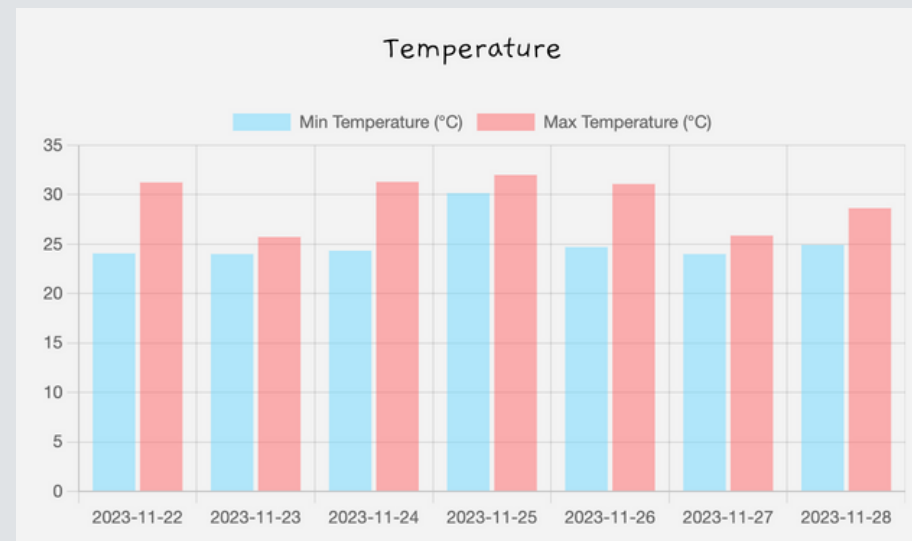


DATA VISUALIZATION

SEE MORE

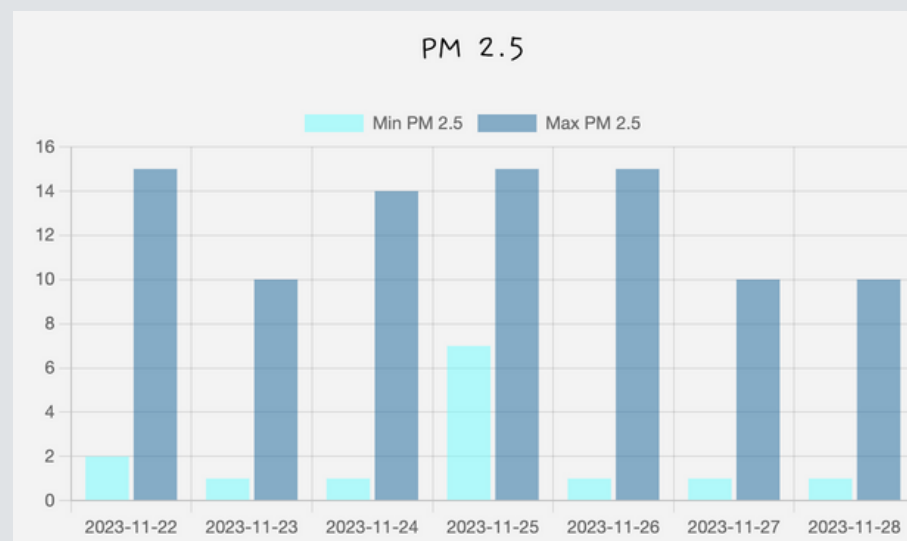
DATA VISUALIZATION

ROOM DATA GRAPHS



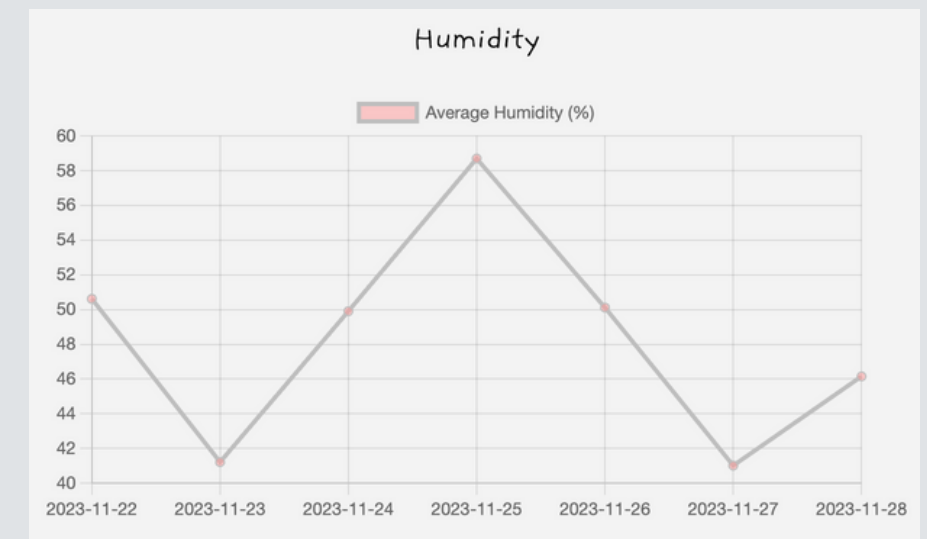
MIN-MAX
TEMPERATURE

Double Bar Graph



MIN-MAX
PM 2.5

Double Bar Graph

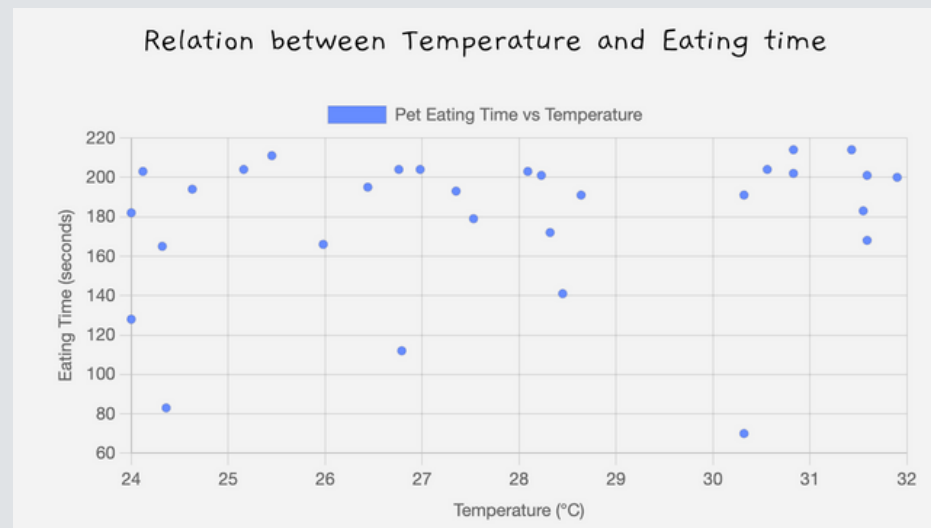


AVERAGE
HUMIDITY

Line Graph

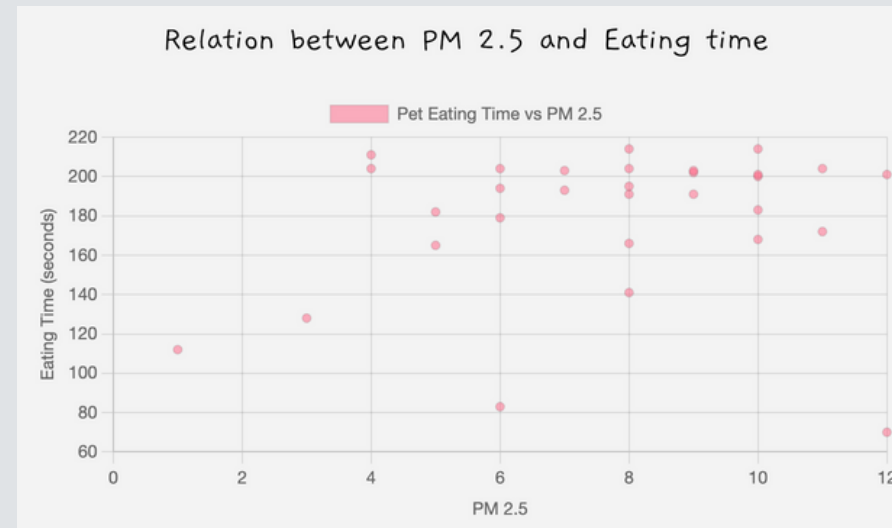
DATA VISUALIZATION

ROOM DATA GRAPHS



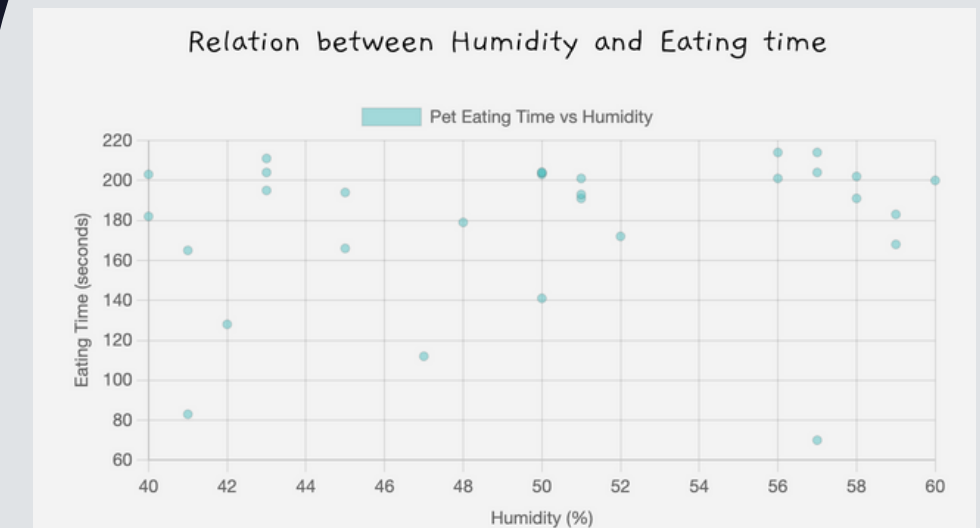
TEMPERATURE VS
EATING TIME

Scatter graph



PM 2.5 VS
EATING TIME

Scatter Graph

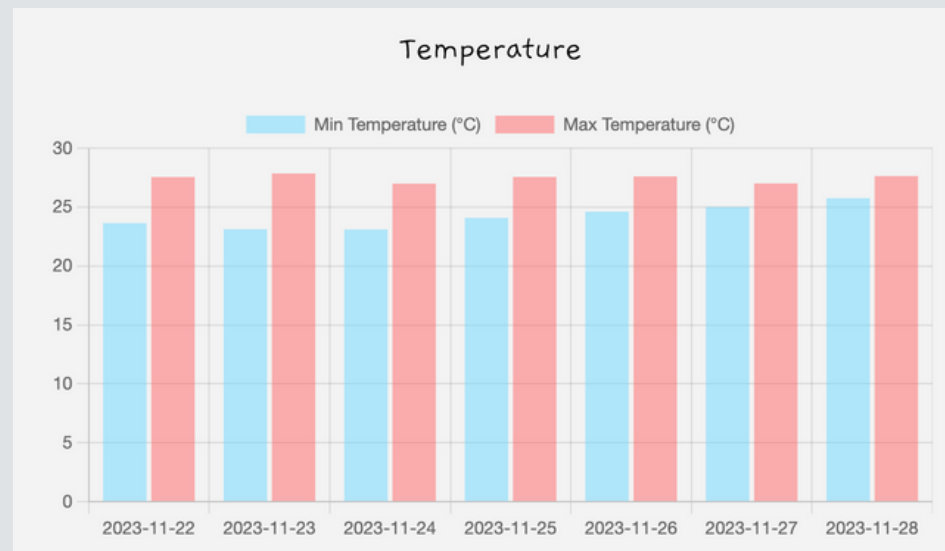


HUMIDITY VS
EATING TIME

Scatter Graph

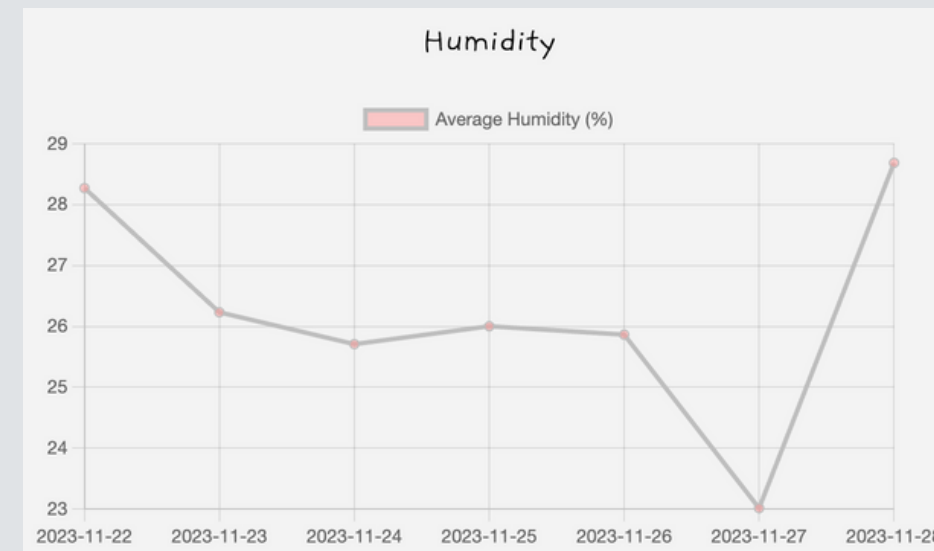
DATA VISUALIZATION

FOOD TANK DATA GRAPHS



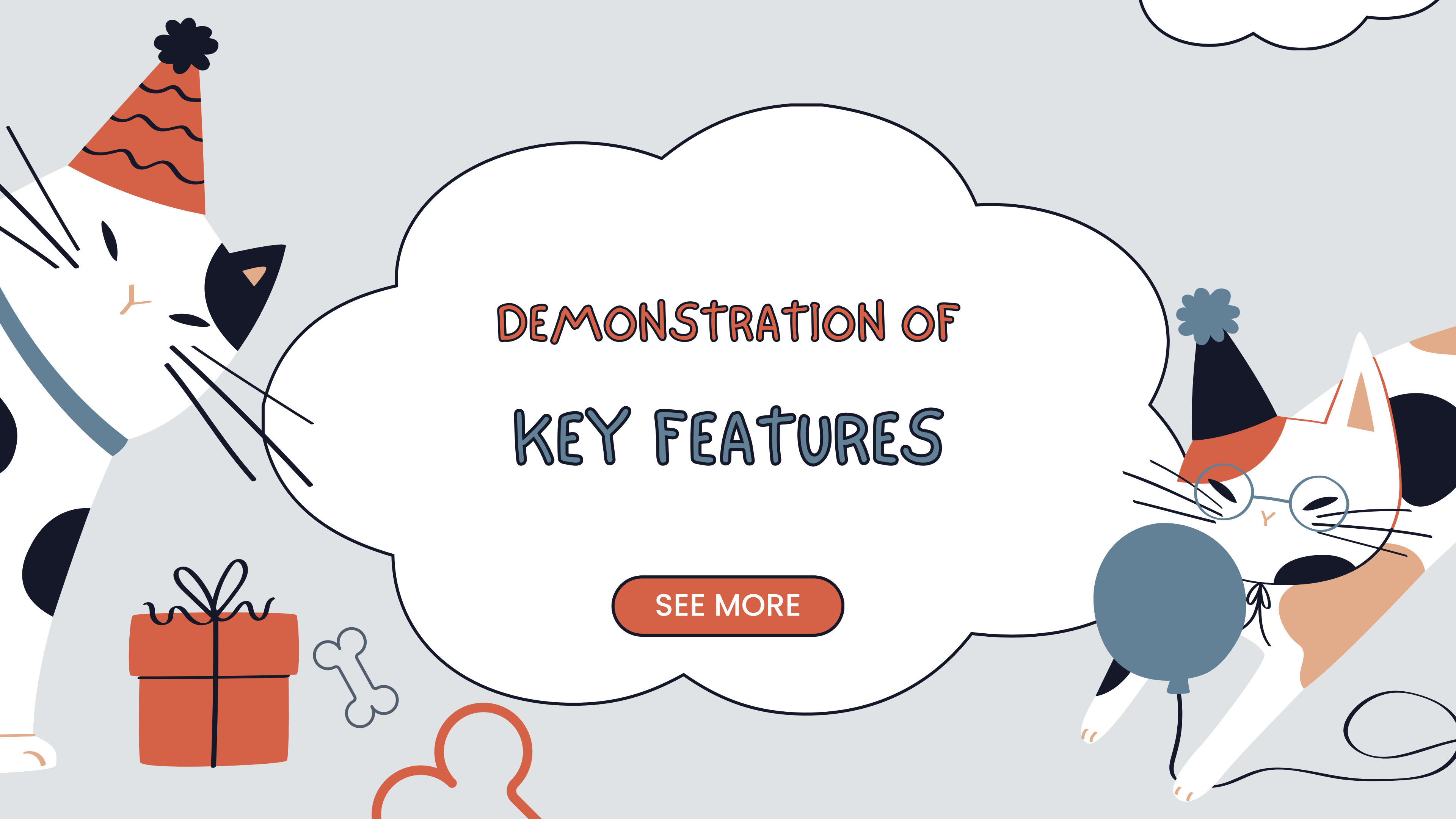
MIN-MAX
TEMPERATURE

Double Bar Graph



AVERAGE
HUMIDITY

Line Graph

The background features a light gray surface with stylized illustrations of two cats. The cat on the left is white with dark patches and wears a red party hat with a black pom-pom. The cat on the right is white with dark patches and wears a black party hat with a blue pom-pom. A red gift box with a black ribbon is on the left, and a white bone is in the center. A red streamer is at the bottom. A large white cloud with a black outline contains the text.

DEMONSTRATION OF KEY FEATURES

[SEE MORE](#)