UNIVERSITY OF CANBERRA

INTRODUCTION TO INFORMATION TECHNOLOGY

ASSIGNMENT-1 (The Solving Problem Process)

NAME-Kamalpreet Kaur

STUDENT ID-u3310275

Background: Automated Pet Feeder System

A local animal shelter is looking for a low cost, programmable automated pet feeder that can –

* Dispense food for cats and dogs at schedules times.
* Monitor whether food has been consumed or the amount of food that has been consumed.
* Alert staff if there`s any issue (e.g., no food dispended food not eaten.

They want a solution that could eventually be implemented using low- cost components like a servo meter and sensors, but task is to design and stimulate the logic of the system.

**STEP-1 UNDERSTAND AND DEFINE THE PROBLEM (ANALYSE)**

* Feeder must include many features in it such as sensors which can measure the weight, feeder must have two parts in it one of it include water and other include food. It must involve internet connection which can give alerts if there will be any problem in the feeder so the staff can handle it wisely.
* INPUTS-It will be feeding schedule which means the time of feeding stored in the memory of the device. Sensor must be insert which detects the level of food remaining in the device. Other sensors such as weight sensors which can detect the weight of bowl if food has been eaten or not.

OUTPUTS- In the device it must be a motor control such as servo motor control which helps to dispenses the food into the bowl for pets. It will be the alert system which helps in giving emails and message if any issue in the device and it will notify to the staff members. Some indicators like visual status indicators help in providing the information about any problem, low food and many more.

* ASSUMPTIONS- firstly, food type is necessary for pets so dry food like kibble, pets can easily eat it as it helps them to get proper nutrition but you don’t put wet or gravy food in it. For pets, it must be single pet feeder as multi-pet resolved by multiple units. When it comes to usage, it will always use in indoor at normal temperature because it can damage by leaving it outside due to bad weather conditions and high temperature.it will operate offline with the help of cloud and some other app features. Storage must be limited memory and logs are rolling for example last 15 days entries. Safety constraints such as shielded motors, limitations in feed bowl and no sharp edges should inserted. Schedule interpreted in local time of installation in the device.

It can be easily comprehended from the following block diagram –

**BLOCK DIAGRAM-**

Power

Supply

Micro

controller

Timer

Pet

Feeder

**STEP-2 ORGANIZE AND DESCRIBE THE DATA**

To organize and describe the data we must have a list of inputs, outputs and provide sample values with operational constraints.

* List Input types
* List output
* Provide sample values

FOLLOWING ARE THE DATATABLE WITH INPUT/OUTPUT AND OPERATIONAL CONSTRAINTS-

|  |  |  |  |
| --- | --- | --- | --- |
| INPUT | TYPE | SAMPLE VALUES | CONSTRAINTS |
| Feeding time | (HH:MM) | 09:00,15:00 | 3-5feeding/day |
| CLOCK | Time | System time | 24-hour format |
| Bin level | Boolean | YES/NO | Must be yes to feed |
| Weight of bowl | Grams | 0-700 | Measured before and after feeding |

|  |  |  |  |
| --- | --- | --- | --- |
| OUTPUT | TYPE | SAMPLE VALUES | CONSTRAINTS |
| Servo motor | Order | Move | Duration <10sec |
| Alert message | Text | Empty food bin, food didn’t eat by pet | Internet |
| Indicator | Color | Red/Blue/Green | Active at time |

**STEP-3PLAN THE SOLUTION (ALGORITHM DESIGN)**

Decision logic level: it can be easily comprehended by following steps:

* Schedule checking- checking the schedule is first thing to do while feeding the pet, it allows to program feeding time and examine the feeder if it is ready.
* Weight measure- it enables to measure the weight of the bowl before and after feeding the pet from it.
* Detect issues- it helps to detect the issues in the pet feeder and give alert to the staff members to check it properly.
* Verifying feeding- after some time if the weight of the bowl is same still after consumption that means pet didn’t eaten that food so then it will send alert to staff.
* Events – it means that it will save all events of the device which means time when we have to feed the pet, sensors and to perform actions.
* Checking presence- it will only proceed when the pet is around that means it will detect the pet and feed it.

**FLOWCHART-**

* Flowchart is uploaded on git hub.

**Step-4IMPLEMENTED THE SOLUTION (WORD CODING)-**

Consecutive tasks with explanations:

* Monitoring – The monitoring loop runs constantly to check the schedule which will be set by staff for feed times or any other manual things.
* Loading configuration- it is used when we have to retrieve schedule which is stored, check the size of portion in the device and give alert to staff. this means that the operating of feeder is done through manual settings.
* Getting current time- staff can get the current time of the device with the help of RTC (real time clock), it helps to match the feeding time for the pets.
* Feed trigger- it will come in action when it is the time for feeding the pet and if command is received manually.
* Verifying level of hooper- it is used when the food container is empty then it will send a food alert to the staff, it also helps to log the events.
* Pet presence checking- if pet come around it then it will provide food to them but if it is enabled to detect the pet then it will skip the time of feeding to avoid the wastage of food for the pets.
* Record pre-dispense bowl weight- it is only used to record the weight of bowl that if it is same after feeding or any changes happened to it.
* Dispensation of food- it will record through the motors and servo which will provide the proper specified portion of the food to the pet.
* Post-dispense weight of bowl- it is used to check that if the correct amount of food was dispended into the bowl.
* Waiting – it allows the pet to weight for 10 minutes to eat the food provided by the device.
* Food intake records-it will compare the pre-dispense weight and post-dispense weight to confirm that what amount of food were consumed by the pet.
* Alerts – it will send alerts to members if the level of consumption is below the ideal level of food consume then it will send alerts like partial meal alerts.
* Events details- it will store the time, size of portion eat by pets, activations of sensors and any alerts in the memory.
* Errors handling- it will handle all the errors by sending alerts to solve the problem.
* Again monitoring- after all process the system will continues to its work until the time of shutdown.

**STEP5- TEST AND DEFINE THE SOLUTION (DEBUG AND VERIFY)-**

Test scenarios

* Pet eats as expected – it means that the weight of bowl decreases by the expected amount of food in the bowl with this no alert will send because the event will log successful.
* Pet does not eat – it means that the weight of bowl will not change after 10 minutes which means that the pet does not eat food. It also sent missed meal alert to the staff
* Food bin empty- if feed is skipped then it will send low-food alert.
* Jam detected- if motor stalls then retry is attempted if it fails in work then jam alert will send.

Refinements-

* Add some adaptive portioning which will based on the eating habits pet
* Include camera snapshot at the feeding for the verification.
* Improve jam detection with the current sensing and encoder feedback.

**PART-2ON THE USE OF TECHNOLOGY (GITHUB)**

All the process is done on GitHub.

Link-

**PART-3 ON AI AGENT INTEGRATION-**

In order to improve my project, I took the help of an AI assistant (Microsoft Copilot) to correct and optimize my reasoning as well as my documentation. The first thing I did with the AI was to have them go over my Step 4 Word Code and make improvements. The AI suggested using modular variables names and adding a timer so that one would not be in a perpetual loop and it contributed to my solution being more efficient. I also searched real life implementation by querying how this might be constructed using hardware such as Arduino or Raspberry Pi. The AI elaborated on the way servo motors and weight sensors can be integrated, making me understand in a better manner how it can be deployed. Lastly, I employed the AI to compose a professional README.md file of my GitHub repository and made it clearly and organized. This procedure served to demonstrate the advantages and the drawbacks of AI. It assisted me in ensuring that the quality of my logic and documentation is high, yet I learned that it is important to validate what the AI-based suggestions are to ensure that one does not accept such suggestions blindly. On balance, the AI integration has allowed my project to become more professionally feasible and realistic, also enhancing my knowledge of the given process in constructing and deploying automated systems.