Keeping connected with your cat through IoT

The development of an IoT food bowl and toy for cats

Ali Mussayab

Anna Ronnhed

Jakob Håkonsson

Paul Vinegren

Institutionen för data- och systemvetenskap

Internet of Things HT2019 (7,5 hp

Autumn 2019

Stockholms universitet

Table of contents

Abstract

This report presents an attempt to design and prototype a system for wellbeing of cats using IoT paradigm. Initial design concept aims to create a system with IoT system characteristics such as new knowledge creation for a context-aware application with autonomous features. This scope has changed over the course of project and focus was dedicated to performing a context acquisition as well its immediate distribution. In the prototype implementation were used Raspberry Pi with a compact sensor network, motorised actuators and an app designed in Android Studio to serve as an interface.

Key words

IoT, design concept, prototype, Raspberry Pi, context awareness, reasoning

Introduction

Background

Thanks to the ever-expanding world of internet the internet has started to come out from our computers and telephones and are now being implemented into our physical environment. A term widely used for this new area of the internet is called Internet of Things or IoT (Höller et.al., 2014). But it is not only the internet that has been evolving over the past decades, but also technologies such as sensors, and actuators that allows physical objects to digitally identify themselves and observe and control the physical environment. Thanks to the technological advances of sensors, actuators and microcontrollers that before were expensive have rapidly decreased in costs and today only costs a couple of cents, further expanding the opportunities for the interweaving of the internet and physical things (Höller et.al., 2014).

This has created a large interest in the market and the growth opportunities are vast. Internet of things can be found in everything from large industries to individuals' homes. People are now able in real time to monitor and control their environment in a completely different way than before (Höller et.al., 2014) One area of people's environment that is highly interesting and posts interesting opportunities in regard to people and connection is the pet industry.

According to a survey done by Novus on behalf of Agria Animal Insurance, SKK - Swedish Kennel Klubb and Royal Canin there are approximately 1.4 million cats in Sweden as of 2017. This means that almost 20% of the households in Sweden have cats (Novus, 2017). Even though many view their cats as a family member and try to spend as much time with them as possible, the reality for many is that they have to leave their cats at home for large portions of the day. This leaves an extensive amount of time where the animal is alone, and the owner is unaware of their activities and wellbeing. This means that people can today have a better connection to their ventilations system at home when they are away then to their pets. These hours away can cause stress not only to people that worry if their pets are ok but also for the cat itself who might become hungry and bored at home.

Some people buy cameras to monitor their home to see what their pets are up to, but they have no way to interact with the pet. And buying a camera might not suit everyone's needs. However, in the recent years the Internet of things has connected people with millions of objects or things and in some cases even people with their pets. There have been several companies that have worked to bring consumers products in relatively low-tech space including the pet industry (Lee, 2019). Some companies have tried to create new and improved versions of older products such as food bowls into smart bowls, something also explored in academics (Adriansyah et.al, 2016) but also new products such as "gaming consoles for dogs" (Lee, 2019). As pets are also residents of smart homes, one could also consider them potential users of smart home applications. For their owners, pets are often considered a part of the family or household. When designing future smart home applications, the pets' wants and needs should therefore also be considered in addition to its human owners, if one were to design with the whole household in mind. This makes the area regarding internet of things and pet care highly interesting since it offers a vast number of opportunities to connect people with somethings close and dear to them that they normally cannot during the hours they are at work, in school or just not home to stay connected with their beloved pets.

Method

At the start the group held two brainstorming sessions to figure out what area and later product that would be developed. The idea had to be feasible to complete during the given time with the knowledge that consisted within the group. During the brainstorming sessions the group used rapid prototyping and sketches to visualize their ideas and try and reach a consensus of what the final product should consist of. By using sketches, a first draft of how the IoT architecture should look was also created which gave an indication of what level of technical skills were required. With help from the course previous lab sessions the team could choose which technologies to implement and decided to use the MQTT broker system, which will be described in detail in the Design and Development section of this report.

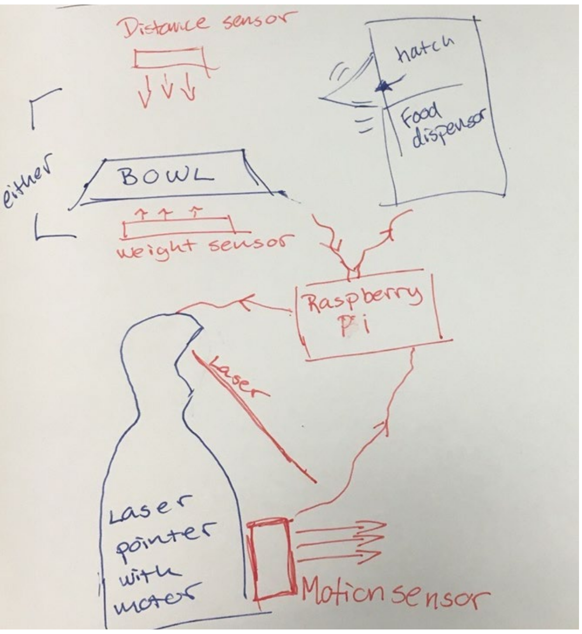

Figure 1 *Sketch of components*

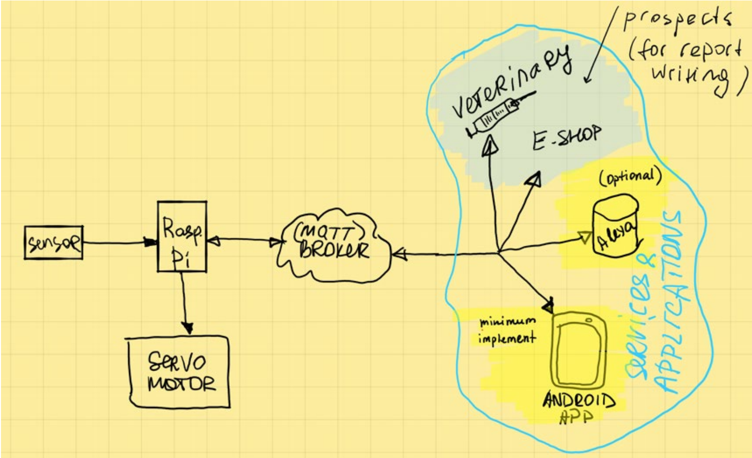

Figure 2. *Sketch of initial architecture*

The team chose to focus on two important concerns for pet owners when they are away from their home, does the cat have food, and is it entertained? These are two areas where products have been offered before but never together as a complete solution. Once a general idea was set the group created a project plan to help with the structuring and time management of the project. In order to work together towards a final product, the group established goals that had to be achieved with the project.

Goals

1. The system will sense that a cat's bowl is empty and send a notification to a mobile android app used by the owner of the cat.
2. The system will use a feeder that can be activated by the user through the mobile app to ensure the cat is fed at the correct times with the capacity of up to 2 meals.
3. The system will use a laser pointer that can be activated by the user through the mobile app that would run for a time period.

In order to achieve the goals, the work had to be split up between the group members and planned. The team put together a GANT schedule and identified what components and task was needed to put the product together this helped the group get. An overview of what needed to be developed and also what had dependencies with each other. Some things could not be developed before certain aspects was already in place, therefore the group started with the basic setup of getting the sensors and actuators working and later implementing the results from these with the MQTT protocol and Android studios.

Milestone Description	Category	Assigned To	Progress	Start	No. Days	W	T	F	S	S	M	T
Base Setup - physical part												
Configuring Raspberry Pi 1	Med Risk	Ali	0%	08/12/2019	3							
Connect components	Low Risk	Ali & Anna	0%	10/12/2019	2							
Build food dispenser	Med Risk	Ali & Anna	0%	11/12/2019	1							
Merge with control part	Milestone	Ali &Paul		12/12/2019	1							
Base Setup - control												
Configuring Raspberry Pi 2	Med Risk	Paul & Jakob	0%	10/12/2019	1							
communication via MQTT	High Risk	Paul & Jakob	0%	10/12/2019	1							
Basic App design	Low Risk	Anna & Jakob	0%	11/12/2019	1							
Merge with physical part	Milestone	Ali &Paul		12/12/2019	1							

Figure 3. *Screenshot of GANT schedule*

Design and Development

Initial design & Concept

This project faced a design challenge based on its initial concept. The group intended the owner of the cat to be aware of the cat's presence and activities real-time. However, the only value it was possible to create were cat's proximity detection to the sensor and food availability. As pet owners have more affectionate relationships to their pets than with a technical gadget, it was decided that the UI of the app would inform about presence of the cat in a more affectionate way to reflect this. Instead of a message describing the state of the sensor or its data, the app would instead show what that data actually meant in this context, in a very direct and visual way. If the cats' food bowl was empty, there would be an illustration of a cat on the screen with a speech bubble saying "I ate all my food. Feed me!". If the cat was close to the sensor of the toy, the UI would then show a different illustration saying: "I'm bored, play with me!". The user would then interact directly with this representation of the cat by pressing the speech bubbles, whereupon an animation would play of the cat doing the activity – eating or playing with the toy. If the cat wasn't present and asking for something, an animation of the cat sleeping would be shown on the screen.

Figure 4. *The first sketch of the User Interface*

One of the aspects of IoT solutions is that the raw data is collected, aggregated and through reasoning is creating new knowledge (Höller et al., 2014b). The team had an ambition to implement this process but due to the time constraints it was decided to divide the project in the "base" and "extended" setup which was optional. In the base setup the system would present new knowledge derived through reasoning to the user without storage or further opportunity for analysis in the future.

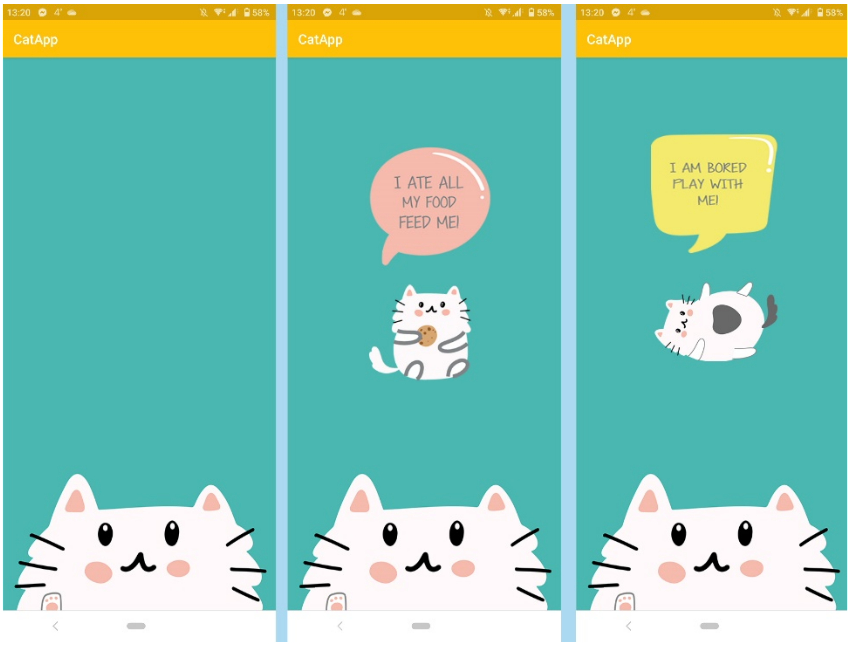

Figure 5 *The final User Interface*

Context awareness & Reasoning

The system acts as a solution to be implemented in a smart-home context for the human user to provide knowledge and control over household devices. Other devices in such a system can be control over lights, locks, temperature or cooking appliances. What is added here is the ability to monitor the cats' behaviour and needs, considering a cat as part of the household. The other side to this is to consider the cat as a secondary user, where it has certain control over the smart gadgets, and can use them to communicate to its owner.

In both cases, the system exists in the context of a cat's life and its food consumption, or need for nutrition, as well as the cat's behaviour and presence. Depending on the engagement of the owner, it further exists in the context of the owner's behaviour with his smart home and pet.

The prototype utilises pull method for acquiring context information. Both discrete and continuous events are used when generating a context. Context model is static as it exists only in predefined structure and states (Höller et al., 2014).

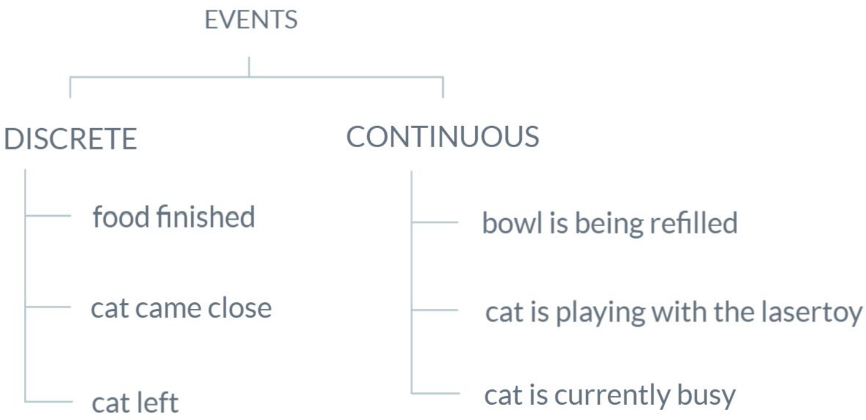

Figure 6– type of events employed for context generation

No-application model is employed for context-awareness. It was simplest solution to implement context acquisition and reasoning within application boundaries. No storage of information was necessary for the type of implemented prototype. As the system has only two measurements at a time, pre-process of the information was done without using any third-party cloud solutions. Part of the reasoning related to context-aware features is happening in the app which serves as interface to the end-user.

Context-aware features include presenting to the user only options relevant at any moment of time (Dey, n.d.). For example, feature to add more food in a bowl is only shown when the system determines that the bowl is sufficiently empty. In the same manner activating laser toy is not an option if there is no cat in the vicinity of the system.

The prototype was implemented with passive context-awareness (Dey, n.d.). Measurements from the sensors are continuously retrieved in order to provide the most relevant features for the user at any moment of time. Active awareness was considered by the team. This alternative was discussed as an option to dispense food automatically either immediately after event “food finished” occurred or with a predefined time-delay.

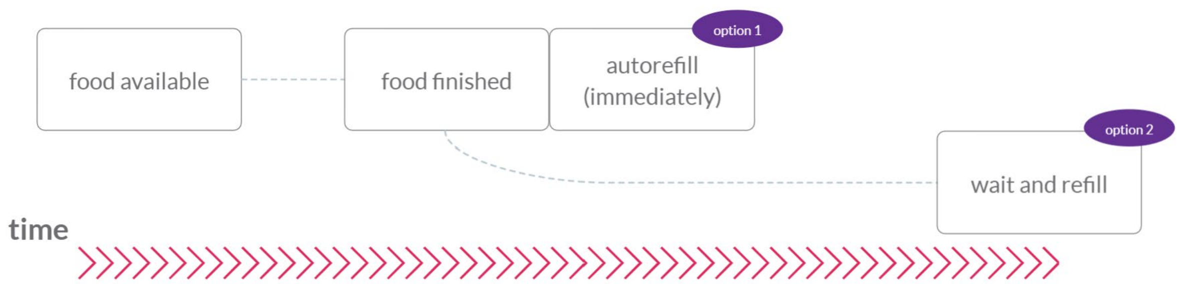

Figure 7– Active awareness considered to be implemented

Automatic execution of those feature is an important aspect of IoT paradigm (Höller et al., 2014); however, it was removed out of the scope of the project due to time constraints. More details of this concept are described in the section Future work.

Hardware design– sensors, laser pointer & servomotors

For this project, a substantial amount of work was put into the physical prototyping and development of the artefact. The team aimed for working prototype which could demonstrate the functionality and let to evaluate the results in the right context.

Input devices – choice of sensors

Input required measurement of two separate ‘things’. For the feeding system, this was whether the food bowl for the cat was empty or not. Several sensors were considered, such as a distance sensor pointed down to the bowl. If it was empty, the distance would be the greatest, as anything filling the bowl would make the distance smaller. An alternative of weight sensor under the bowl seemed most optimal for producing accurate result of food amounts present in the bowl. Finally, the decision was made to put a light sensor at the bottom of the bowl pointing up. This was done after discussing the scope of our solution. As we were working with a Raspberry Pi, the system could only receive a digital value, not an analogue one. It was considered to use an Arduino board with an analogue sensor such as the weight sensor attached which could measure a range of values and then communicate that digitally to the raspberry, but after a discussion it was decided that this would be too big a scope given the time limitation and it wasn't necessary for our solution – 'base setup' required only empty/full type of information, not how empty it was, and thus, a light sensor sufficed to provide us with that data.

Similarly, the choice of sensor to detect if the cat was present or not was made to be a simple proximity sensor. No exact knowledge was necessary of what the cat was doing, only that it was present, which interpreted directly by the group's assumption that it wanted to engage with the toy.

Output devices – choice of actuators

In the food dispenser, we decided to use a servo motor to open a hatch. This was done since we could program exactly how much (how many degrees) the hatch would open and for how long. This way, we could potentially control the portion sizes, and in future development, it would be possible to have this be adjusted either by the user or by the system itself. This would only need to be two adjustable variables in the code, degrees and duration which wouldn't affect the physical design.

The choice of the laser toy was based on the fact we had access to an old toy we could salvage and redesign. The toy consisted of a laser diode attached at an angle to a DC motor, making it so that when the toy ran, the laser moved in a big ellipse. Here, we would only need activate this toy using the Raspberry's signal ports. This was a difficulty at first as the ports weren't capable of outputting enough voltage to run the toy, but this was quickly solved by using a transistor as a gate. By doing so, the voltage from the port could act as a signal to let power through to the toy from an external power source. The full wiring is illustrated below, with the exception that some wires were longer than shown in order for all components to be at different places.

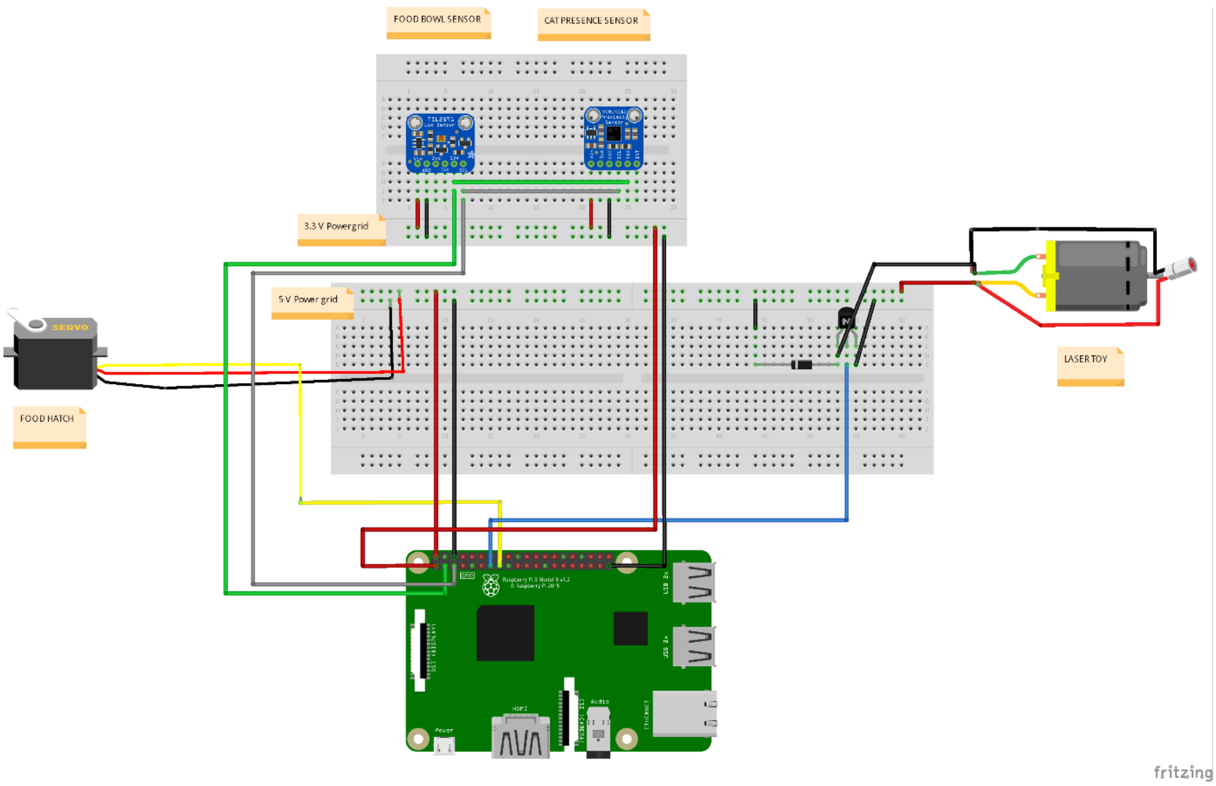

Figure 8 *The wiring of the system*

Lastly, this whole setup was mounted in a physical low fidelity prototype made by cardboard, in order to be tested in the real context. This prototype was not meant to communicate or test the look of a final product, but rather its functionality.

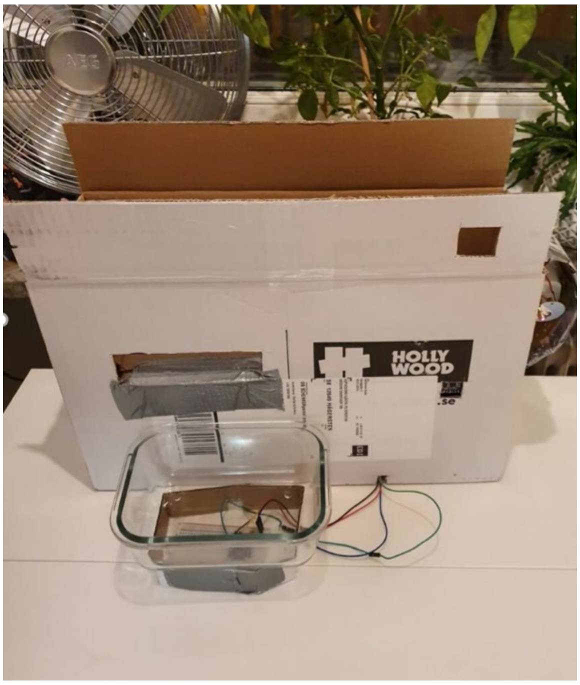

Figure 9 *The assembled physical artefact for the system*

Software design

Python Code

The core of the system is a Python Code located and run on Raspeberry Pi. The code comprises of several functions such as pulling information from the sensors, reasoning, actuators operation and context distribution.

Pulling information from the sensors is performed with the help of libraries available for the sensors from 'adafruit'. Each library is sensor-specific and provides unique methods to receive the reading. Communication with them happens via I2C bus, which is initialised in the beginning of the code.

Actuators are controlled via pins 22 and 27. For servomotor the control pin is used for pulse-width modulation with different duty cycles to move the hatch in a desired position. To operate laser, respective pin changes its state to whether high or low which in turns influences transistor it is connected to.

Reasoning is located inside 'while-true' loop. Every 2 seconds a new loop starts with pulling new reading from the sensors and then going through a set of 'if-then' operators in order to pre-process and tag collected raw data.

Context distribution is also placed in the same 'while-true' loop right after data tagging. Generated context is further distributed via publish/subscribe employing MQTT protocol which is more explained further below.

Android Studio

The app serves as a main interface to interact and control the system. Application prototype was chosen to be done in Android Studio as the developers were familiar with the IDE, thus it was assessed to be less risky. The application is representing one screen with several building blocks containing textual information, visual information and buttons for interaction. A code written in java is steering which blocks are appearing depending on the system status. System can be in the following statuses:

BOWL_EMPTY, BOWL_IS_BEING_REFILLED, BOWL_FULL, CAT_HERE, CAT_AWAY, LASER_ACTIVE

The system receives messages from the remote hardware module (HM) described above. The messages are published at regular time intervals (2 seconds) when the HM is sensing the environment. This allows to update the interface in a responsive manner. When one of the actuators is enabled the messages are not being publish until the operation is completed. To achieve correct representation of the user interface during this phase, the application is flagging 'actuatorEnabled' Boolean variable.

layoutAdapt() function is responsible for the correct interface representation. It contains a set of 'If-Else' blocks which determines which blocks to show/hide for all possible combination of system statuses.

Due to the limitations of Android studio, it was not possible to create the animations envisioned in the initial concept. Instead, interface representation only showed and hid different blocks which were illustrated to as static images of the cats states – eating, playing or being away.

MQTT selection

MQTT is a lightweight messaging protocol which was selected as the main communication technology for this project. A great value with this MQTT is that the sender, or publisher, sends the data in topics which the receiver can choose to subscribe to (Jaffey, 2014). By selecting which topics to subscribe to, only the relevant data will be received, instead of everything the sender sends out. For an IOT application this is a useful structure since many smart devices, especially in a smart home environment, are only interested in sending or receiving a certain kind of data. This could be home temperature, status on locks, lights and similar. This efficient structure reduces bandwidth and overall communication needed. In other words, it creates more efficient data distribution and more scalable solutions (IBM, 2012).

The main reason MQTT was chosen for the project, apart from being a fitting technology within an IOT-enabled home, was because it uses a broker in between the client which is sending data and the client which is receiving. Because of this broker, the publisher and subscriber could be developed independently and simultaneously. As long as they publish and subscribe to the same topic, the communication worked.

This approach made it possible to efficiently divide the work throughout the project. Some group members could code the Raspberry Pi with the connected sensors to publish topics containing sensor data using the programming language python, while other group members could work on the app development using android studio, where they coded in Java. A final group member could code the xml document while working on the look and feel of the app. This didn't affect the functionality of the app as it only focused on how it should be presented, choosing what from the java code to display in the UI and how. As all components interacted with each other solely through the broker, no code ever interfered with another.

Evaluation

Over the course of 4 seminars we have gone from ideation, design, development and implementation phases. In each of the phases we shared our progression with our peers and received feedback from them. Initially, during the first and second seminars their feedback became a part of our design process and help us decide how to develop Cat Connect efficiently. Furthermore, we scheduled and developed the prototype.

Figure 7 *The artefact being tested by a cat in the context. Left: the feeding functionality, right: playing*

To make sure the prototype components would work when put together in the final artefact (in our case a cardboard box) each component was individually connected to the Raspberry Pi and tested, once a connection was made through the Raspberry Pi to the Android app the component was ready for assembly in the prototype box. Once each component had been tested to be connected individually and together outside the box, they were all placed in the box and finally tested by a cat.

By testing with an actual cat, the focus was brought from “does each component work together connected through IoT?” to “is this functional in a real-life scenario?” It quickly became apparent through testing that the food bowl part of the prototype worked well with detecting if the food bowl was empty or full and the refilling of the bowl worked smoothly even though the sound of the servomotor opening the hatch spooked the cat a little in the beginning. The other thing that became apparent through testing with the cat was that the laser toy was not an engaging toy whatsoever. Even though the prototype could detect if the cat was close by and turn on the laser toy the cat did not pay much attention to it. There was no way to get feedback through the app either if the cat was engaging with the toy making the laser spin around in circles for no good reason for a few minutes. This is something that is worth further experimenting with, how to create an engaging toy for a cat to play with when a human is not present.

Discussion

As described in the introduction, the project is relevant as a component in a smart home system – both considering the cat as a user and as part of the rest of the "things" in a household that would be connected through IOT implementation in the home.

In order to be useful however, some parts of this prototype would need to be improved. For instance, when assembling the laser toy component and testing it on a cat, the cat decided to engage with the toy only briefly. Uninterest showed after the system had been running only a few seconds. Part of the reason for this could be that in this prototype, the laser only moved around the floor in a constant, never deviating ellipse shape, as it was simply mounted on a spinning DC motor. Compared to how a human would operate a laser pointer, this is a very simple movement. In contrast, when a person plays with a cat using a laser, not only is the movement more random, but it also responds to cat's attention and anticipates its intent. Moving away a laser in a right moment when the laser is almost "caught" seems to be critical for the play to be attractive. A pre-programmed or random pattern for laser movements would never achieve this. The system must understand exact location, orientation and possible intent of the cat. This implies a need for a tracking device working together with a machine learning model trained for that purpose.

The feeding system performed as expected. The food dispensed unhindered during the time while the hatch was opened. Execution of the dispensing device, however, did not seem to allow for a control of the food amount released. Current execution could be suitable for a one-time bowl refill if the owner is away for a shorter period. More prototypes must be created to find an optimal construction allowing for more control of the food amounts released.

Conclusion

The project has been implemented at the level of "base setup" as presented in the Method. The prototype had all featured functioning and responsive to the level expected by the team. Ambitions to implement active awareness were restricted by the tight time schedule. These topics were discussed thoroughly at multiple stages of the project and despite not being implemented were source of inspiration. While in the end being a functional prototype, at its current state the created artefact would not yet be useful to implement into an IoT-enabled smart home. In order reach that goal, further development would need to be put into the artefacts reasoning and overall intelligence for deeper and more complex interactions between the cat, the owner and the system. Still, the group believes that the project gave a greater understanding of IoT paradigm and its place in modern and future worlds.

Future Work

During the course of the project and with the final prototype finished and evaluated it was clear that the prototype itself would not be suitable in e real-life scenario but with further development there was a lot of potential. In this section five potential future development of the prototype is presented, Smart Home, Auto Feeding, Auto Activation of the Auto Feeder, Modular Multi Feeder and Smart Toy.

Smart Home

To further develop Cat Connect one of the essential features would be to connect it to smart home gateways like Amazon Alexa or Google Home. Smart home devices can use the audio notifications to inform the user of the empty bowl and voice activation for the pet owner to give a verbal command to the system to release the food.

The benefit of having a smart home connectivity feature is that it could be used big houses or busy household.

Auto Feeding

The product could have an automated mechanism to feed the pet. This feature could be either activated by a timer or using machine learning where the system determines the eating habits or routines and using that data it automatically feeds the cat at the most appropriate time.

Auto Activation of the Auto Feeder

The product could be developed to work with the GPS and location service on the phone where it can detect that the owner has left the house and hasn't come back for a long period of time, hence triggers the auto feeding and takes over the control of the system.

Modular Multi Feeders

If the Cat owner owns multiple cats, the system could be developed with a modular design to integrated with multiple bowls and laser addons.

Smart Toy

Through the testing it became clear that a "smarter" toy was needed in order to activate the cat. Further developments could be done to make the toy "smarter" reading the motion and attention of the cat to create better patterns to engage in play. This would require a camera both for the motor that would control the toy but also a camera for the owner to see if the cat is actually present in front of the device engaging in play, since the prototype today does not give any feedback to the human if the activation of the toy actually leads to engagements from the cat.

References

Adriansyah, Andi & Wibowo, Muchd & Ihsanto, Eko. (2016). *Design of Pet Feeder using Web Server as Internet of Things Application.* Conference: 2nd International Conference on Electrical Engineering and Informatics (Icon-EEI 2016), At Pekanbaru, Riau, Indonesia

Dey, A. K. (n.d.). *Understanding and Using Context*. Retrieved 2020-01-18 from www.cc.gatech.edu/fce/contexttoolkit

Holler, Jan, Mulligan, Catherine & Tsiatsis, Vlasios, From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence [Elektronisk resurs], Academic Press, 2014

IBM. (2012) *What is MQTT and how does it work with WebSphere MQ*. Ibm.com 2012-03-16 accessed 2020-01-18 from https://www.ibm.com/developerworks/mydeveloperworks/blogs/aimsupport/entry/what_is_mqtt_and_how_does_it_work_with_websphere_mq?lang=en

Jaffey, Toby. (2014) *MQTT and CoAP, IoT Protocols*. Ecplise.org 2014-02 accessed 2020-01-18 from https://www.eclipse.org/community/eclipse_newsletter/2014/february/article2.php

Lee, Ben. (2019). *What the pet technology boom tells us about the fourth industrial revolution*. Business.com 2019-09-19 accessed 2019-12-02 from https://www.business.com/articles/what-the-pet-technology-boom-tells-us-about-the-fourth-industrial-revolution/

Novus (2017) *Hundar och Katter allt vanligare svenska hem.* Available at: https://novus.se/nyheter/2017/11/hundar-och-katter-allt-vanligare-svenska-hem/