Cat feeder

Shimon Mimoun
Naor Eliav
Nahama Weill

Instructor : Yossi Zaguri

Software Design Document

Date: (24/03/2020)

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Purpose

This software design document describes the architecture and system design of cat feeder, TO Help cat owners feed the cat automatically and controlled.

1.2 Scope

The system is divided into 2 parts: hardware (sensors, weight, etc.) and software (data base, application, etc.).
The system will track the cat's behavior and update the cat owner.
The system will absorb the cat's arrival, decide whether to allow it to eat, and weigh the cat.
The system studies the cat's behavior and sees if anything is abnormal in its behavior.

1.3 Overview

This document talks about the system and product we are building, and the way to feed cat automatic and controlled.

1.4 Reference Material

This section is optional.

1.5 Definitions and Acronyms

This section is optional.

2. SYSTEM OVERVIEW

The system helps cat owners to feed the cat by:
The cat have a collar with a specific chip, this chip is identified by the system and thus allows the cat access to food. Thanks to this identification we will know the desirable amount of food we want for a cat and at the same time eating and weighing data of the cat will be kept and updated in the system.
The cat owner can track this data, give other cats eating privileges, and give eating permissions by hours and dates.
Thanks to learning the cat routine we can update the cat owner and the veterinarian in exceptional cases.
The product will mimic the normal cat routine as much as possible.

3. System Architecture

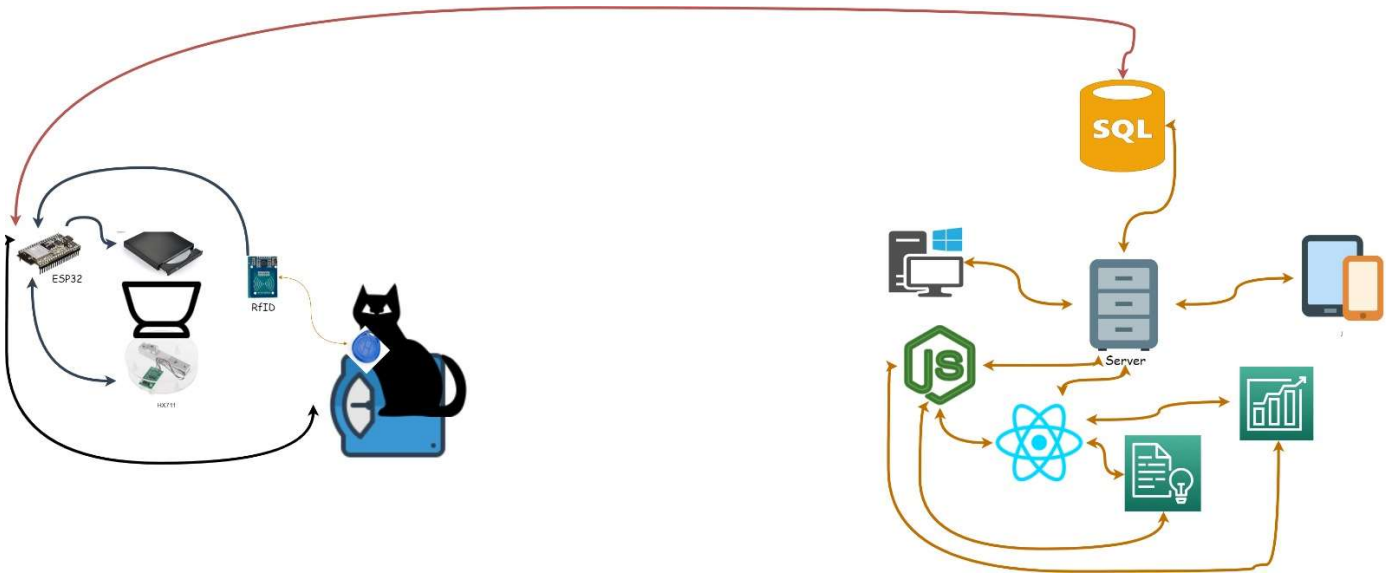
3.1 Architectural Design

Our system is divided into several parts: the hardware part and the software part, these parts are also subdivided.

Hardware - We have the main part (the brain of the system) the ESP-32, this part connects and communicate between all parts of the hardware. We have the chip, the RFID that allows us to identify and differentiate each user. In addition, we also have a number of sub-hardware like weight, which can be added according to user's request.

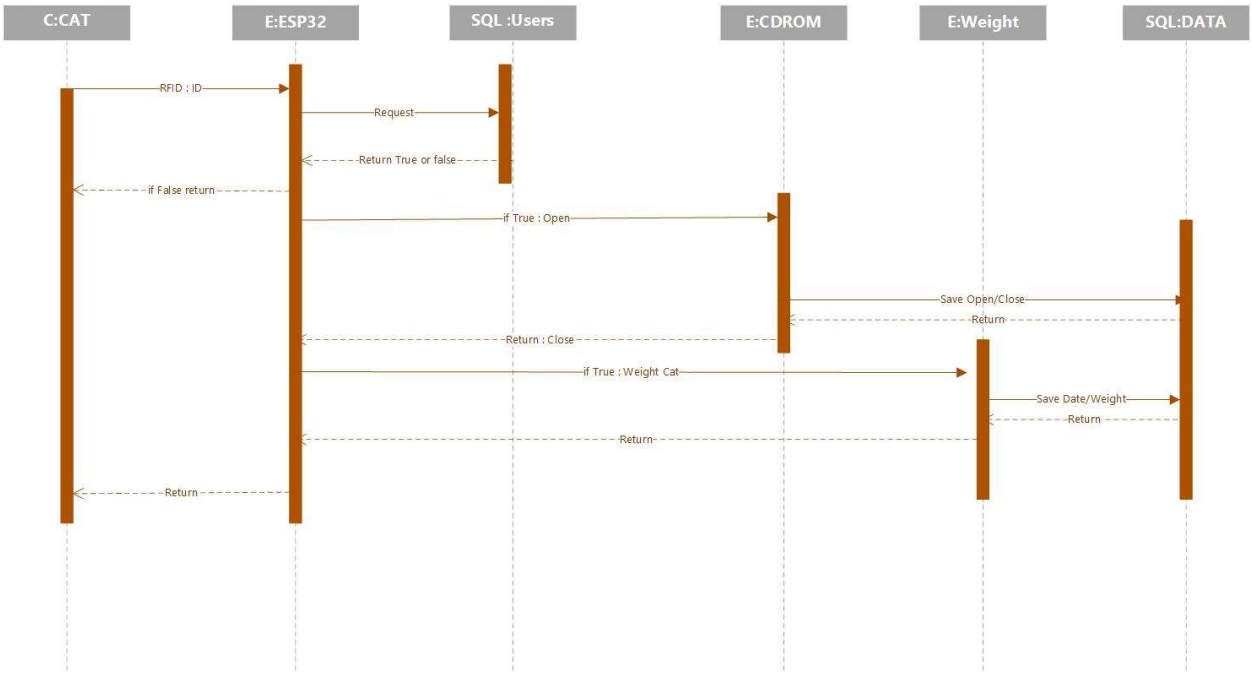
Software - The software is divided into several sub-sections. We have the central server with which we will transfer and store all information. We will use a database MYSQL to save user data. In addition, we have the Machine Learning section that allows us to study the cat routine and identify exceptional cases. The last part is the app (UI) which allows the user to watch all the data and make possible changes.

The hardware part and the software part are connected and communicated with each other with the help of the server.
We have divided this way to make sure that each part will have it's unique task. Therefor we will be able to track the system's operation and in case of disfunctions identify the source of the problem immediately.

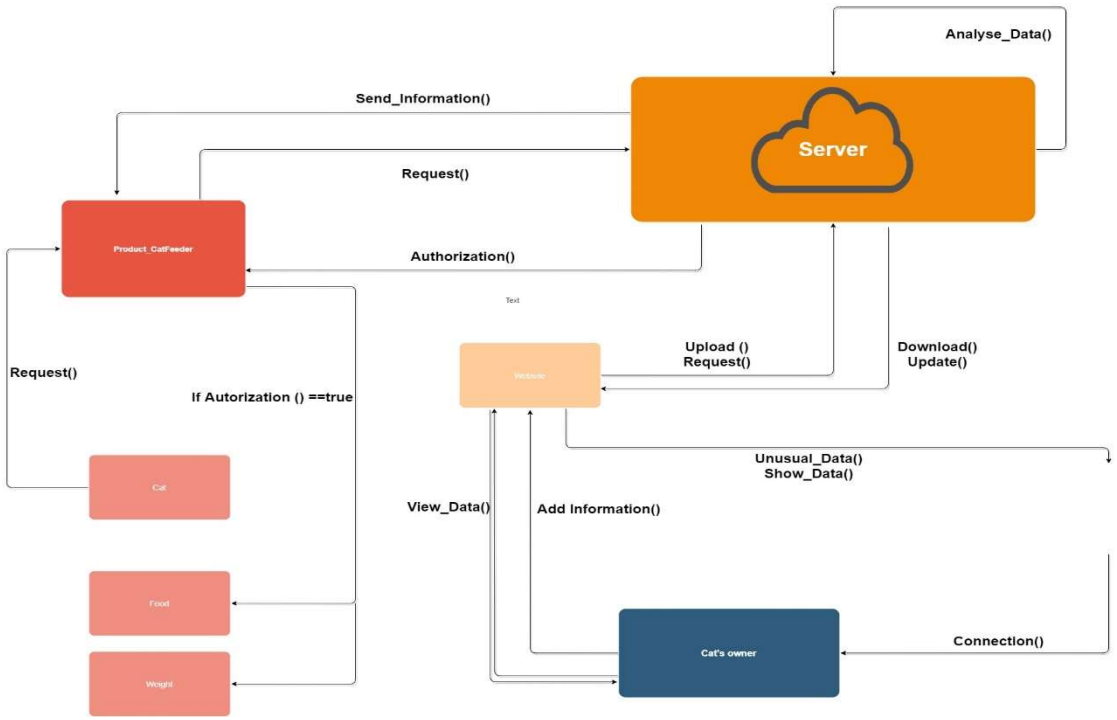


3.2 Decomposition Description

Sequence diagram:



Data flow diagram:



3.3 Design Rationale

At first, we debated many types of hardware: ARDUINO, RPI, ESP-32, there were many considerations but in the end we decided to use ESP-32.
For our project we needed the smallest component as possible, not too expensive and include in it all the hardware for internet connection.
In addition, we decided to use MySQL and not Neo-4j or MongoDB, since we do not produce much data and all the data we produce is similarly concentrated, so we do not need a complex database and we wanted to use a rational database that is more reserved and reliable.
We also debated between RFID and NFC. after many discussions on the subject, we decided to go for RFID, because no too secure component was needed and priority was given for Absorption for greater distance.

4. DATA DESIGN

4.1 Data Description

In each interaction between the cat and the system, a new data is created.
The data contains the date and time the cat connected to the system, and at the request of the cat owner, we will also keep the cat's weight and food weight.
This data is stored in the database MySQL by the user's account number and inside it is arranged by a unique number of the cat.

4.2 Data Dictionary

Cat: Request().
Cat's owner: Add_information(), View_Data().
Product- CatFeeder: Request(), Autorisation().
Server: Analyse_Data(), Upload(), Request(), Autorisation(), Send_Information().
Vet: Connection().
Website: View_Data(), Unusual_Data(), Show_Data().

5. COMPONENT DESIGN

Autorization()- This is a function that allows us to connect to the hardware and check whether this user is familiar to the system and whether it has permissions access. If so, it checks to see if it is within the allowed hours and sends a signal to confirm the system's opening. And if not, the function notifies the system that it cannot connect.

Send_Information()- The server will send updates to the hardware on users' permissions and free hours where they can access food.

Request()- This is a function that sends a request + data about the login request.

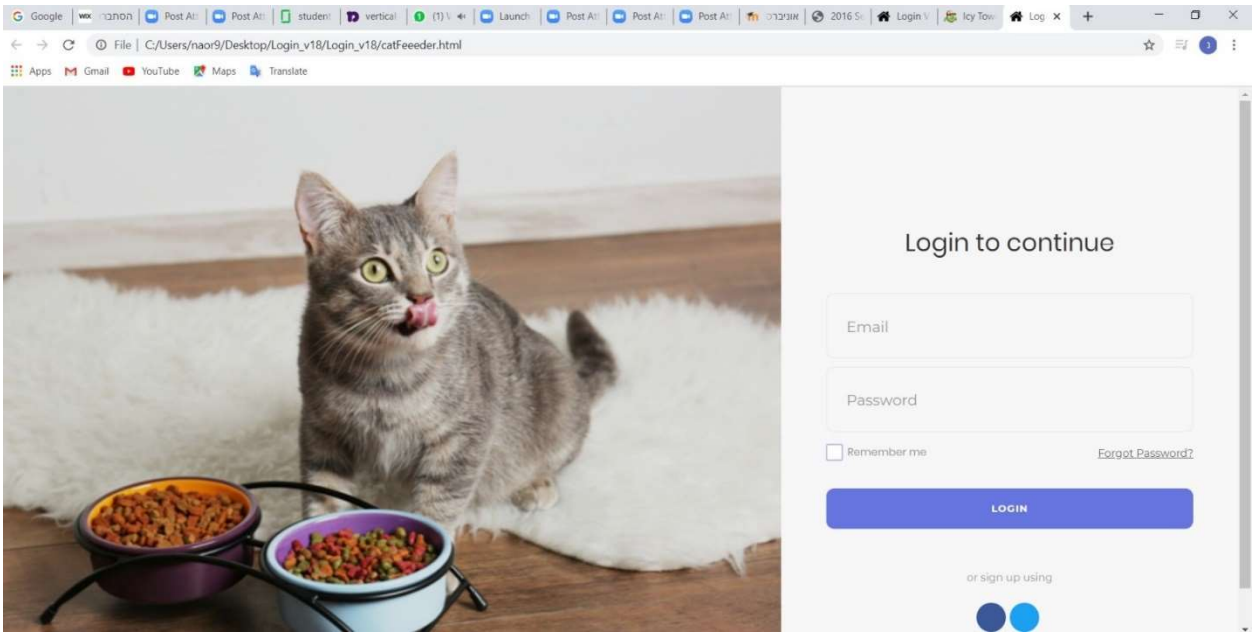
Analyse_Data()- This is a function that allows us to support machine learning behind the scenes. The server will analyze the data thus creating the cat routine. With the help of the routine it will be able to know when there is exception in the cat behavior.

6. Human Interface Design

6.1 Overview of User Interface

The system is basically split in two. The Cat Part: The cat comes to the feeding device and as soon as the system identifies the cat by the chip it serves the cat its intended food. In addition, there is a weight that will weigh the cat and keep the data on the site intended for the cat owner. The part that is intended for the cat owner: The part that shows the data about the cat's condition. We'll show this through an website where the user can see the cat's condition using graphical tools. The user will be able to see the amount of food the cat ate, the eating hours, the eating frequency, the weight of the cat and more.
The user can also decide which cat have the permissions to access the food.

6.2 Screen Images



7. APPENDICES

This section is optional.

https://www.espressif.com/en/products/hardware/esp32/overview
https://fr.wikipedia.org/wiki/ESP32
https://letmeknow.fr/blog/2019/08/01/programmation-dun-esp32-basic-over-the-air-ota-avec-lide-arduino/
https://www.instructables.com/id/Connected-Cat-Feeder-Using-a-Strain-Gauge-and-an-E/
https://randomnerdtutorials.com/esp32-esp8266-mysql-database-php/
https://iotdesignpro.com/projects/iot-controlled-led-using-firebase-database-and-esp32
https://lastminuteengineers.com/how-rfid-works-rc522-arduino-tutorial/
https://www.pinterest.fr/pin/102879172721540238/