

Design and Implementation of a Pet Care System

Huong Nguyen
Degree Programme in Information Technology
Bachelor's Thesis, 15 credits

Need for a Pet care system

The subject of this thesis was to build a pet care system using ReactJS, NodeJS and Raspberry Pi as well as analyze different aspects of this approach.

The main aim was to create a pet feeding machine with Raspberry Pi as the main component (Figure 1) and a progressive web application used for remotely control the machine.



FIGURE 1. Raspberry Pi connected with servo

Pet feeding machine

The pet feeding machine includes a food dispenser, a servo and Raspberry Pi. Pet's food could be released from the food dispenser as a result of Raspberry Pi triggering the servo to rotate the dispenser's spinner.

Pigpio – a library of C language – was used to control the servo's motion. A web server was developed and deployed in Pi applying Pigpio and NodeJS so that it could handle the requests

from the web application. The web server could solve both requests for immediate feeding and timed feeding.

Progressive Web App

The web application was built using ReactJS, React-bootstrap and Date-time-picker. This application was a progressive web application meaning that it could be installed in mobile devices.

It has a page including two buttons corresponding to two core features (Figure 2). *Feed now* was made so that it could send a request for immediate feeding, while *Schedule* would show a date-time picker for the user to choose



FIGURE 2. Web app's UI

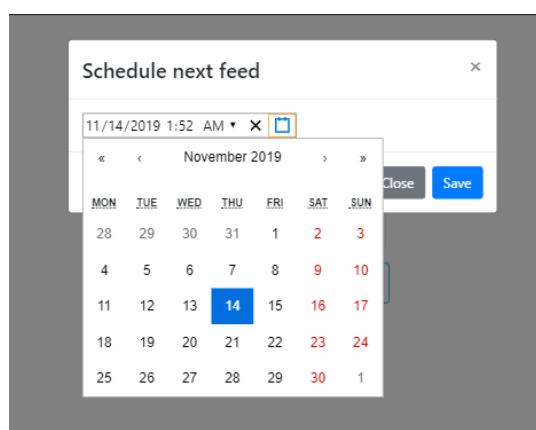


FIGURE 3. Date-time picker

Even though the UI was simple, it was responsive and offered all necessary components including feedbacks and a date-time picker.

Results

A pet feeding machine and a progressive web application were built successfully. Core features including *Feed now* and *Schedule next feed* were implemented. IoT and web technologies were studied carefully and applied to produce the system. Figure 4 illustrates the completed pet feeding machine.



FIGURE 4. Pet feeding machine

Evaluation

The results were satisfying and they fulfilled the requirements planned at the beginning. Although the system was built to work in the local network environment only, it could be optimized to work in WAN, meaning that it could be accessed from anywhere. The success of the project proves that it is possible to develop an effective pet care system with ReactJS, NodeJS and Raspberry Pi.