CSE1901 - Technical Answers to Real World Problems (TARP)

Project Report

PET SITTER

By

Mohamed Ashraf Ali K : 20BCE1630
Sanjil K C : 20BCE1855
Lenin Vasan : 20BCE1892

B. Tech Computer Science and Engineering

Submitted to

Dr. Sweetlin Hemalatha C

School of Computer Science and Engineering

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

April 2023

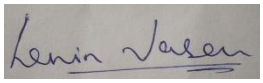
DECLARATION

We hereby declare that the report titled "**A pet care system"** submitted by me to VIT Chennai is a record of bona-fide work undertaken by me under the supervision of **Dr. Sweetlin Hemalatha C**, School of Computer Science and Engineering, Vellore Institute of Technology, Chennai.

Signature of the Candidates

Mohamed Ashraf Ali K
Reg No. 20BCE1630

Sanjil K C
Reg No. 20BCE1855



Lenin Vasan
Reg No. 20BCE1892

CERTIFICATE

Certified that this project report entitled "**A pet care system"** is a bonafide work of **Mohamed Ashraf Ali K(20BCE1630), Sanjil K C(20BCE1855), Lenin Vasan(20BCE1892)** and they carried out the Project work under my supervision and guidance for CSE1901 - Technical Answers to Real World Problems (TARP).

Dr. Sweetlin Hemalatha
SCOPE, VIT Chennai

ABSTRACT

Pet Sitter (A pet care system) is an alternative to ruthless shelter homes or expensive pet care centers, where the pet owner can scroll through dozens of pet lovers with whom he wishes to keep his pet. The main aim of our platform is to relieve pet owners from the guilt associated when traveling without their pet or leaving them in a boarding kennel. On our platform, the pet lover has to make an account and set up their profile mentioning where they stay, their experience with the types of pets, their cost to keep different kinds of pets, and whether or not they will provide food during the stay. Once a profile has been set up, it will be visible in the listings to the pet owner. Pet owners can make a decision based on ratings and reviews as to whom they want to keep their pet with.

The pet lovers can register themselves by depositing a security amount, identity proof, and other details. Their years of experience and ratings will move them on top of the search result of the pet owner.

CONTENTS

1. Introduction

Industry estimates show there are around 19 million pets in India (around 80 percent of these are dogs), and on average, 6,00,000 pets are adopted every year. Pets become a part of the family the moment they step into our house. Pets become very used to human company, unlike stray animals who usually live on their own. It’s a sad fact for pet owners that vacation planning can be such a hassle. So,we came up with the idea of a pet care system to fill in the void between pet owners, pet lovers, and the pet. There are many people who are excellent pet caretakers but cannot own a pet due to their work lives. But such people are willing to take care of others' pets during weekends or during their free time. Connecting such people with pet owners and creating and expanding this community is the main goal of this solution.

1.1. Objective and goal of the project

Pet Sitter is a platform that connects pet lovers and pet owners where the pet owner will be able to choose a pet lover nearby his place who is willing to take care of his pet on his behalf. The main aim of our platform is to relieve pet owners from the guilt associated when traveling without their pet or leaving them in a boarding kennel. On our platform, the pet lover has to make an account and set up their profile mentioning where they stay, their experience with the types of pets, their cost to keep different kinds of pets and whether or not they will provide food during the stay. Once a profile has been set up, they will be visible in the listings to the pet owner. The pet owners can make a decision based on ratings and reviews as to whom they want to keep their pet with.

1.2 Problem Statement

Pets provide emotional support to many people around the world. But they need constant care from people they stay with. Rapid urbanization, coupled with rising disposable

incomes and a shift to nuclear families, are driving more people to get pets. The pet and the pet owner kind of become dependent on one another. Owning a pet comes with certain responsibilities and lifestyle impacts. Most people cannot go on a vacation or even someplace nearby unless there is someone there at home to take care of them. Owning a pet by a nuclear family brings in a lot of lifestyle restrictions to the family. The entire family cannot go out on vacation as someone has to stay with the pet. Nearly 23 percent out of 2000 surveyed take a vacation once every few years, some even less often than that.

1.3 Motivation

The key motivation to develop Pet Sitter was to help pet owners enjoy a peaceful vacation and the pet lovers who cannot own a pet for any reason to be able to own a pet just for a few days. The Pet Sitter application connects the pet owners to pet lovers or carers near them by using geolocation through a booking system. The nearest pet lovers are displayed to the pet owner based on both their geo locations. The amount is charged by the pet lover on a per day basis. One unique feature of Pet Sitter is the tracker. The pet owners can track their pet through their phone to ensure that their pet is safe.

1.4 Challenges

The key challenges that were faced while developing the Pet Sitter application are :

1.4.1 Software

Satisfying all the software and version requirements.

1.4.2 Hardware

The tracker is a unique feature of the Pet Sitter application, but it came with certain challenges.

- Difficulty in finding the GPS module online as it was mostly not deliverable or out of stock.

- The GPS module which would send and receive the trackers coordinates faced connectivity issues which were solved later.
- Placing the tracker on stable level ground due to bulkiness

2. Literature Survey

Design of handheld positioning tracker based on GPS/GSM

GPS positioning technology and GSM wireless communication technology have been widely used in the military field and urban transportation and other civilian areas. Aiming at the singularity of handheld positioner function on the market, this paper designs a combination device of handheld positioner and tracker. By extracting GPS navigation information frame parameters and GSM wireless data transmission. The realization of the tracker in the TFT LCD screen real-time display itself and the tracking side (i.e., the position of the relative position information) function, completed conversion from latitude and longitude coordinates to Cartesian coordinates. The device can be used for the loss of items, the elderly children tracking and geo-location data mapping, data accuracy, and easy operation.

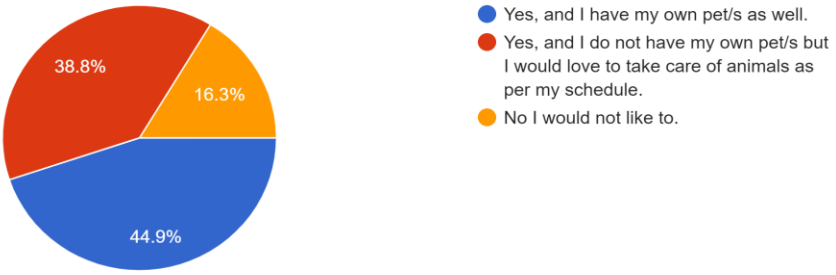
Designing a Low-Cost Location Tracker for Use in IoT Applications

This paper reports on a new low-cost location tracker design, utilizing GPS/BeiDou and 2G rather than the relatively costly, and much more limited coverage, 4G cellular solution. The target retail cost of the tracker is 7 Euro or less. The target market is in IoT asset tracking applications. The tracker has been successfully demonstrated in a pilot test in China for tracking 50,000 auto parts boxes.

2.1 Our Survey

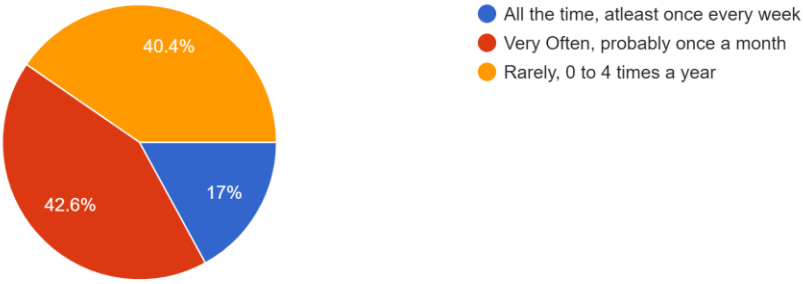
Would you be willing to take care of someone else's pet in your free time and earn some money from it ? If yes do you have a pet of your own as well?

49 responses



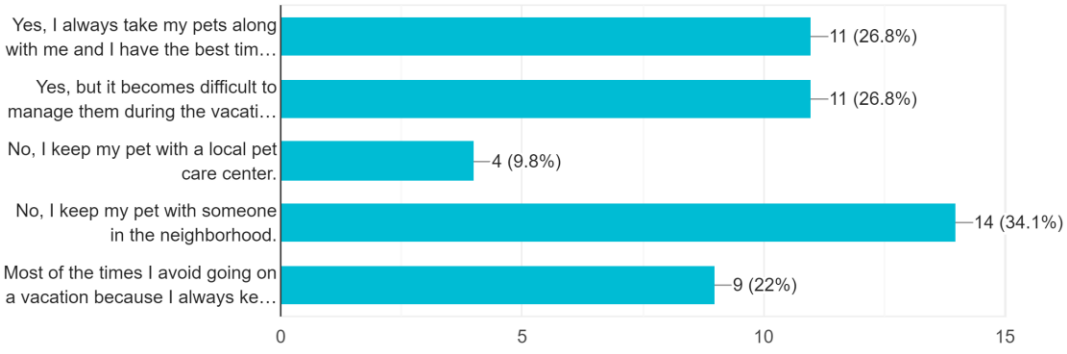
How often did you take a day off to explore places around your neighborhood or a vacation before the pandemic ?

47 responses



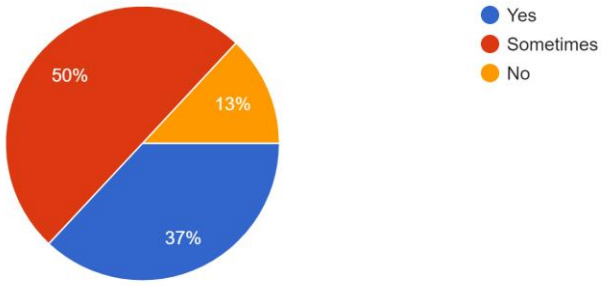
Did you take your pets along with you for a long vacation before the pandemic?

41 responses



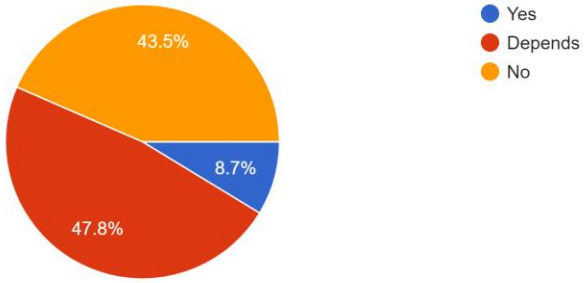
Does having a pet make it difficult to plan vacations?

46 responses



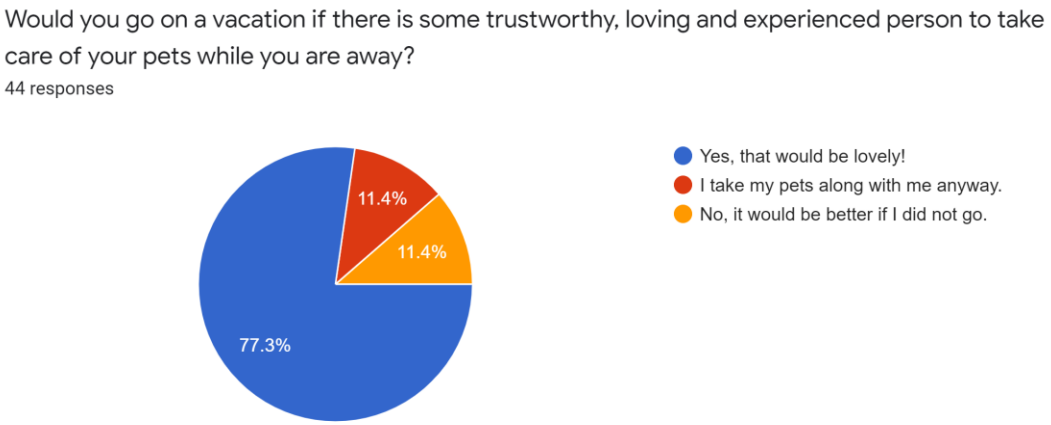
Do you trust your local pet care centers while you are away for a vacation?

46 responses



Does it bother you when you leave your pet behind with someone while you go for a vacation?

43 responses



QUESTIONARIE FOR FIELD VISIT:

About Honden kennel - This Kennel is maintained by one person and he has been maintaining it for the past 7 years. It's location is near Irumbiliyur, Tambaram - Visited by Lenin

1. How many pets are you taking care of?

   5 dogs boarding, 12 own pets.

2. What are the pets you are taking care of?

   Dogs alone

3. How many employees are there in your centre?

   Two people – the owner Naveen and Madan

4. What kind of services do you provide?

   Grooming, Day care, Whelping, Mating, Dog training.

5. How many pet owners are leaving their pets in the centre?

   In the working days we get minimum 5-6 dogs.

   During holidays around 10-15 dogs.

6. Why do the pet owners leave their pets here?

   They are working or sometimes far from town.

7. What do pet owners expect from your side?

   - Owners expect space for dogs rather than cages because the dogs roam around

regularly when they are in their homes.

- Safety of their pets.
- Owners expect their dogs not to be tied.
- Food expectations are high and particular, some owners expect only veg food to be given.
- Expect caring for more cheaper rates.

8. If the pet owner is going on a long vacation, do they leave their pets here for a couple of days?

Sometimes they leave pets for 3-7 days minimum, and people travelling far away also leave their pets for around 3 months.

9. Do you provide a 24*7 service?

Yes.

10. What are the difficulties' faced while working?

- Dog bites.
- Skin infection from dogs.
- Taking care of family planned dogs.
- Few owners leave their pets and don't receive them back due to its health condition or issues so we will have to find new owners.
- Fees aren't being paid properly.

11. Are you willing to provide the service online?

- We have used promotional pages; we will provide if necessary.

12. Do you prefer to continue as a dog care owner or do you prefer to work as a pet sitter?

- It depends on the salary provided, if we get more salary than our income in the care we would prefer as a pet sitter.

About Awesome pets - This Kennel is maintained by 4 people, they are maintaining it for past 7 years. It's location is near, Peerkankaranai, New Perungalathur - Visited by Sanjil

1. How many pets are you taking care of?

   8 dogs boarding and 4 cats.

2. What are the pets you are taking care of?

   All pets (especially dogs, cats and birds)

3. How many employees are there in your centre?

   8 people – 4 owners (Raju, Yuvaraj, Giri Prasad, Dibashika) and
   4 employees

4. What kind of services do you provide?

   Boarding, Grooming, Sitting (Going to home and taking care), Pet Sales, Transporting.

5. How many pet owners are leaving their pets in the centre?

   Per Day 2 to 3 Dogs and weekend 4 to 5 dogs and in festival time 10 above. Monthly 30 customers.

6. Why do the pet owners leave their pets here?

   - Going for vacation
   - Going for work in abroad

7. What do pet owners expect from your side?

   - How they take care at home same way pet sitters also should take care.
   - Video Call if the pet owners ask
   - Same food that that follow the pet sitters should follow

8. If the pet owner is going on a long vacation, do they leave their pets here for a couple of days?

   They leave pets for 4-5 days minimum, and people travelling far abroad leave their pets for around 2 to 3 months.

9. Do you provide a 24*7 service?

Yes.

10. What are the difficulties' faced while working?

- Ticks on dogs
- Tick Fever

11. Are you willing to provide the service online?

- Yes if the customers increase it is good for them.

12. Do you prefer to continue as a dog care owner or do you prefer to work as a pet sitter?

- Yes, Pet Sitter If more income comes it will be better.

13. Cost of service they provide?

- 500 per day for dog, 300 per day for cat and Grooming 1500.

About Dream pets - This Kennel is maintained by 1 person and he has been maintaining it for the past 5 years. It's location is near, Good Will Nagar, Tambaram West – Visited by Ashraf

1.How many pets are you taking care of?

3 dogs boarding, 4 own pets.

2.What are the pets you are taking care of?

Max all pets(especially dogs, cats, birds)

3.How many employees are there in your centre?

6 people – 1 owner and 5 employees

4.What kind of services do you provide?

Grooming, Day care, Dog training, Cage providing for pets

5.How many pet owners are leaving their pets in the centre?

In the working days we get minimum 4-5 dogs.

During vacation time 9-10 dogs.

6.Why do the pet owners leave their pets here?

They are working or going for vacation.

They also want to train the dog during that period

7.What do pet owners expect from your side?

- Safety of their pets.
- Owners need the pet to be vaccinated before leaving there
- Owners expect their dogs not to be tied.
- Food expectations are high and particular, some owners expect only veg food to be given.
- Expect caring for more cheaper rates.

8.If the pet owner is going on a long vacation, do they leave their pets here for a couple of days?

Sometimes they leave pets for 3-7 days minimum, and people travelling far away also leave their pets for around 3 -6 months.

9.Do you provide a 24*7 service?

Yes.

10.What are the difficulties' faced while working?

- Dog bites.
- Communicable diseases if it is not being properly vaccinated
- Taking care of untrained dogs.
- Fees aren't being paid properly.

11.Are you willing to provide the service online?

- We have used promotional pages; we will provide if necessary.

12.Do you prefer to continue as a dog care owner or do you prefer to work as a pet sitter?

- It depends on the salary provided, if we get more salary than our income in the care we would prefer as a pet sitter.

3. Requirements Specification

3.1 Hardware Requirements

- NEO-6M GPS Module
- Arduino UNO
- Antenna

3.2 Software Requirements

- Arduino IDE

4. System Design

4.1 Software

The website has been hosted on Heroku platform. On visiting the website, the home page will be rendered displaying what PET SITTER is all about. This page will have separate buttons to login and signup as pet owner or as pet caretaker.

SignIn as Pet Owner-

Once signed in as pet owner, he/she will have options to view his/her profile, view the about page of PET SITTER, view the previous bookings and view the services offered by PET SITTER.

View Pet Owner's Profile-

On visiting the profile page, the pet owner will be able to view his/her profile as well as the pet's profile and change the password. On clicking on the edit button, the pet owner will be able to edit his/her profile along with the profile image as well as edit the pet's profile such as age.

View Bookings by Owner-

The owner will be able to view all his previous bookings along with the names of the pet caretakers, booking date, location,etc.

Pet Boarding Service-

Profiles of the pet care takers' living in the vicinity of 16-17 Km range to the pet owner, will be displayed in sorted fashion such that care takers nearest to the pet owners at top and those far away at the bottom. On clicking on the 'book' button of any one of the caretakers, the user will be directed to the profile page of the caretaker where the user will select the boarding date and the last date when he/she wants to collect back the pet. Based upon the number of days, the cost will be calculated automatically. Once the user clicks on the 'Confirm Booking' button, he/she will be redirected to the payment gateway.

Pet Tracking Service-

This service allows the owner to get the love location of the pet. The tracker connected to the pet will send the location coordinates to the server, using which an API will show the exact location on the map.

SignIn as Pet Caretaker-

On signing in as pet caretaker, the home page will have two options- view his/her profile and view booking made to him. The caretaker will also be able to view the about page of PET SITTER.

View Pet Caretaker's Profile-

On visiting the profile page, the pet caretaker will be able to view his/her profile and change the password. He/She will also have a button to disable his/her availability so that his/her profile wont be displayed to any of the pet owners while searching. On clicking on the edit button, the pet caretaker will be able to edit his/her profile along with the profile image.

Booking made to the Pet Caretaker-

On clicking on this button present on the home page by the pet caretaker, he/she will be able to view all the past and current bookings made to him.

4.2 Hardware

NEO-6M GPS Module Introduction

The NEO-6M GPS module is a GPS receiver that can locate all locations on Earth as it is able to track approximately 22 satellites. It consists of a high-performance u-blox 6 positioning engine. Measuring 16 x 12.2 x 2.4 mm, its compact architecture along with its low power consumption makes it a good choice for IoT projects. Overall it is a good cost-effective GPS receiver.

NEO-6M GPS Chip

In the middle of the GPS module, you can find the NEO-6M chip. This is responsible for tracking up to 22 satellites and any location on the Earth on several channels. Due to its highly sensitive tracking nature, it makes the NEO-6M module a popular GPS tracker.

Some key features of NEO-6M chip include:

- High sensitivity for tracking
- Low supply current (~45mA)
- Is able to track 5 locations per second with an accuracy of 2.5m (horizontal).
- Comes equipped with PSM also known as Power Saving Mode. This mode causes very less power consumption by turning the module ON/OFF according to the need.
- Great use as GPS trackers in smart watches due to very low power consumption (~11mA)

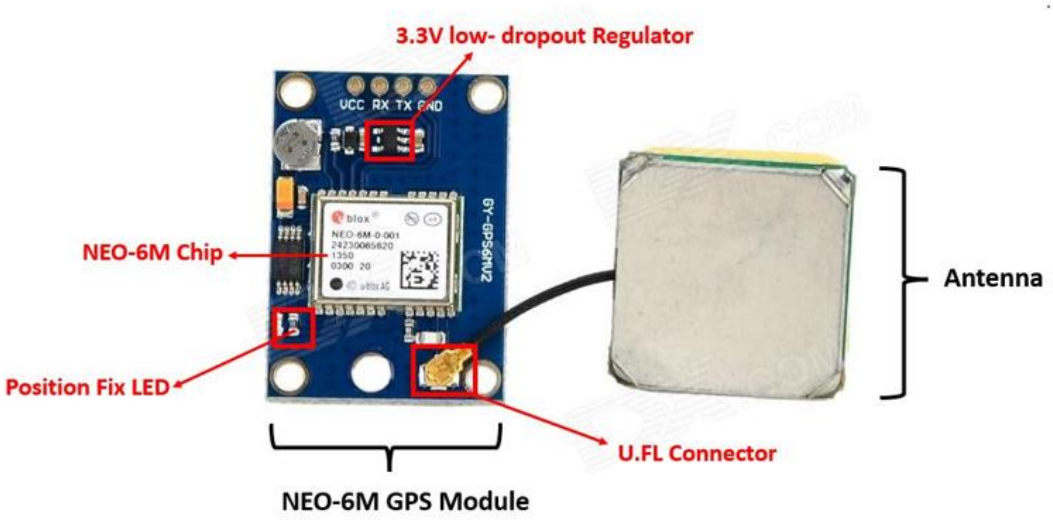
Position Fix LED Indicator

Moving ahead, the module comes with a position fix LED indicator. This LED indicates through its blinking effect whether the module is searching for satellites or has already found them. If the LED blinks after every second, then it indicates that the position fix is found. However, if the LED does not blink then the module is still searching for the satellites.

3.3V low-dropout Regulator

The module also comes equipped with a 3.3V LDO regulator (MIC5205). This provides an efficient linear voltage regulation with ultralow-noise output and very low

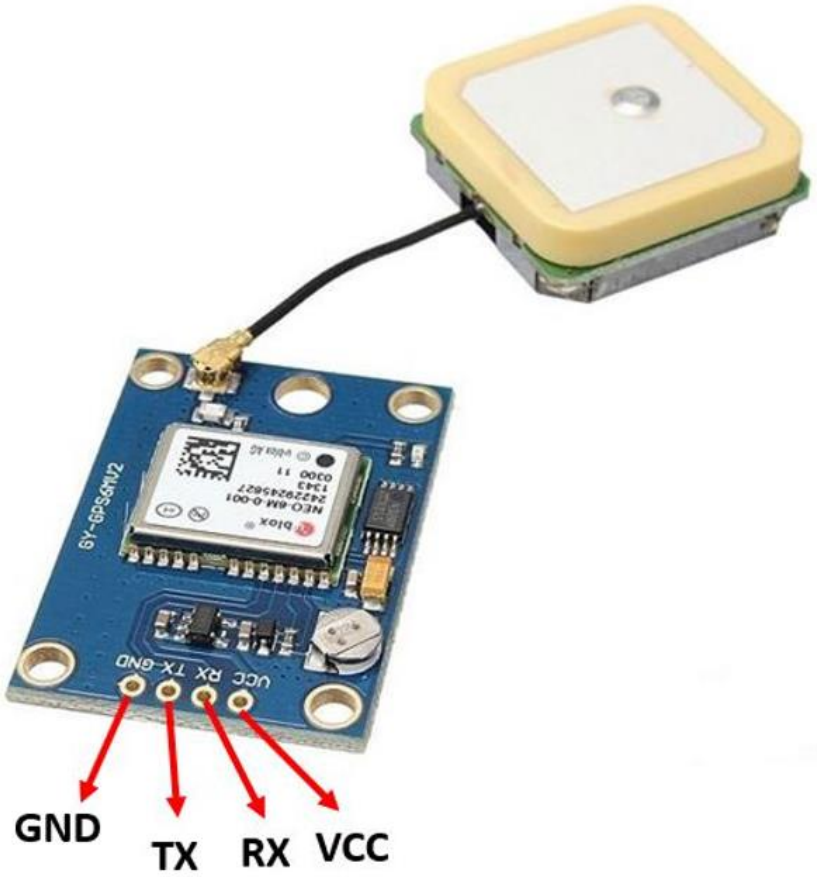
dropout voltage. Additionally, the module is can also tolerate 5V easily so programming it with Arduino is very convenient.



Specifications

The table below shows some specifications of the NEO-6M module.

Type	GPS
Supply	2.7 V-3.6 V
Operating Current	45mA
Operating Temperature	-40°C ~ 85°C
Horizontal Position Accuracy	2.5m
Communication Protocol	NMEA, UBX Binary, RTCM
Features	RTC Crystal and External Interrupt/Wake up
Interface	UART, SPI, USB and DDC

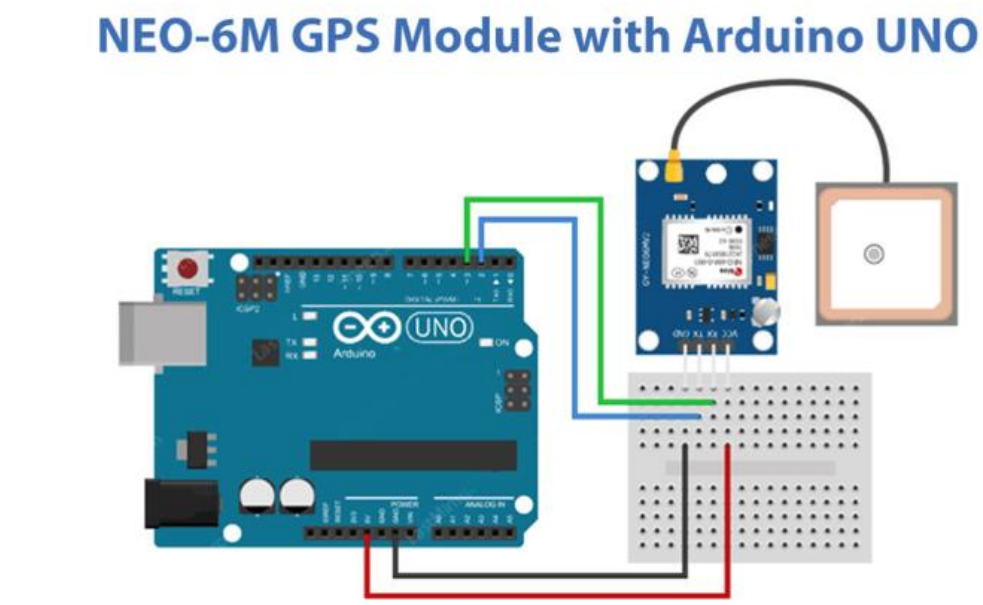


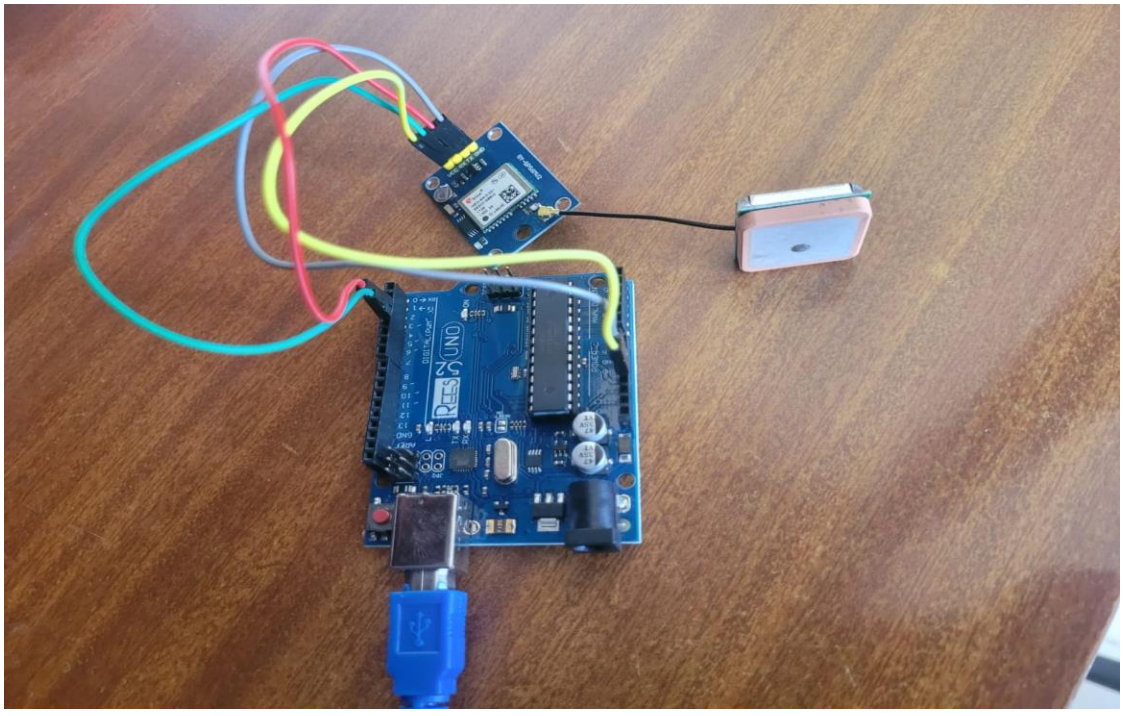
GND	This is the ground pin that will be connected with the ground of the Arduino UNO.
TX	This is the transmission pin used for serial communication.
RX	This is the receiver pin used for serial communication.
VCC	This is the VCC pin used to power up the GPS module. Connect it with the 5V of the Arduino UNO board.

5. Implementation of System

5.1 Circuit diagram:

ARDUINO:





Wiring a NEO-6M GPS Module to an Arduino

Now that we know everything about the module, we can start connecting it to our Arduino.

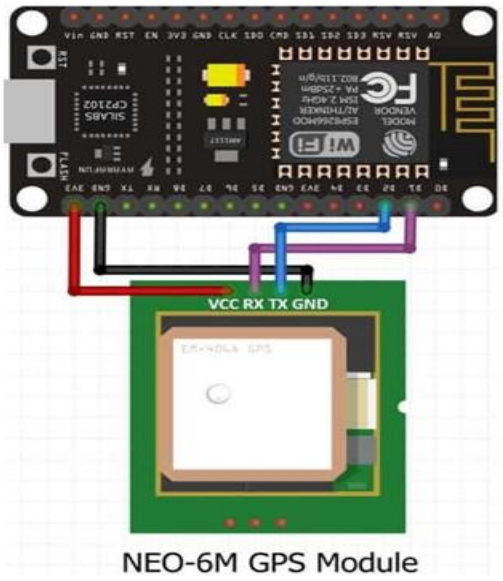
Begin by connecting the patch antenna to the U.FL connector. You can thread the U.FL cable through one of the mounting holes.

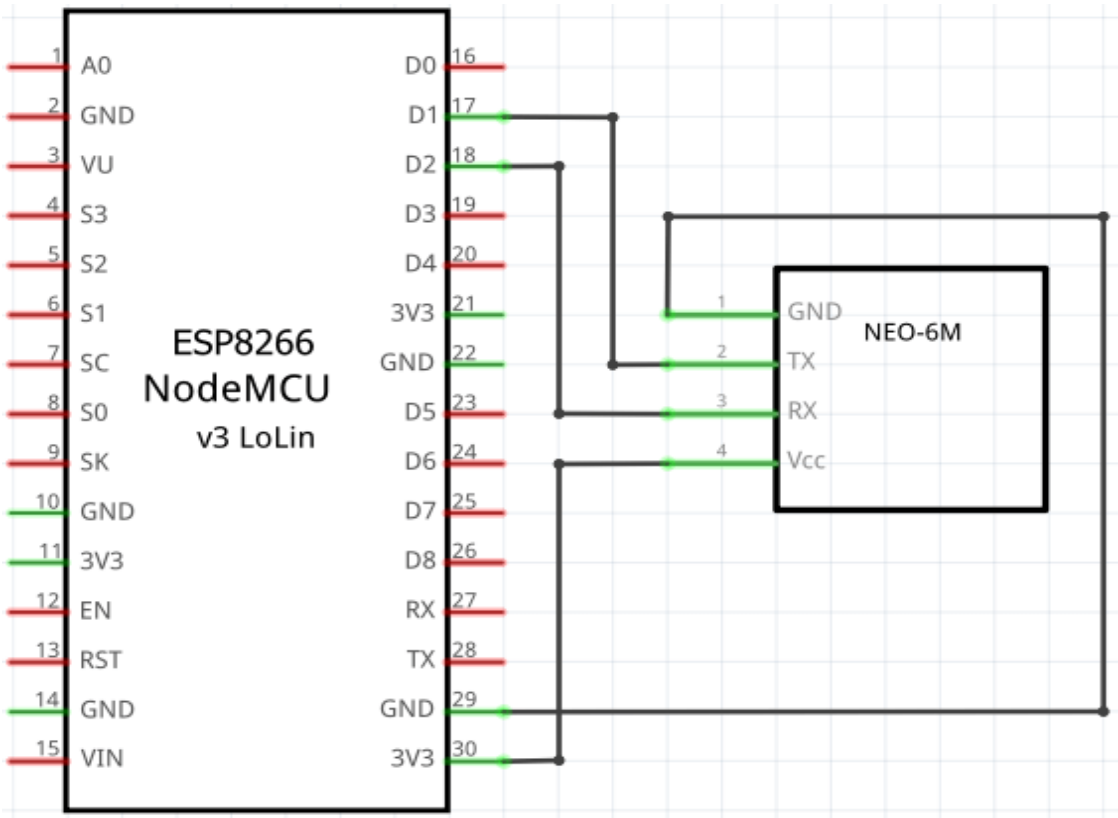
The module usually comes with unsoldered header pins. So you will need to solder them first.

Next, connect the VCC pin to the 5V pin on the arduino and GND to ground.

Finally connect the Tx and Rx pins on the module to digital pins #2 and #3 respectively.

NODE MCU:





GPS interfacing with nodeMCU to track location for PET SITTER

NodeMCU is ESP8266 based development board. It features ESP-12E as its processing core. It is a 32bit MCU. It has 14 GPIO pins, single channel 10 b it integrated ADC. It supports UART, I2C, SPI communication. It is 3.3V compatible, it cannot handle 5V. .

The connections between NodeMCU and GPS module

NodeMCU	GPS module
3V3	VCC
GND	GND
D1 (GPIO5)	RX
D2 (GPIO4)	TX

GPS module takes some time to capture location details once it is powered on. NodeM CU starts webserver and waits for a client to get connected to the webserver. Once clie nt is connected to the webserver, NodeMCU sends location details to connected client. The location details are displayed in a simple webpage designed using HTML.

Steps:

1 Connect the circuit as shown in the schematic.

2 Upload the code after changing ssid and password.

3 Open serial monitor in Arduino IDE and note down IP address of the webserver.

4 Open any Browser and enter the IP address of the webserver.

5 It will display Location details, date, time and Google maps link.

6. Results and Discussion

The Pet Sitter application successfully connects the pet owner and pet lover. Pet owners can book pet lovers by going to the services section in the application Top pet carers near them will be displayed to the pet owner. The pet owners and pet lovers can edit their profile at any time. Availability option present in the pet carer profile gives control to the pet carer as to when he would like to take bookings. The tracker system built with the gsm gps module successfully sends and receives messages to the owner of the pet.

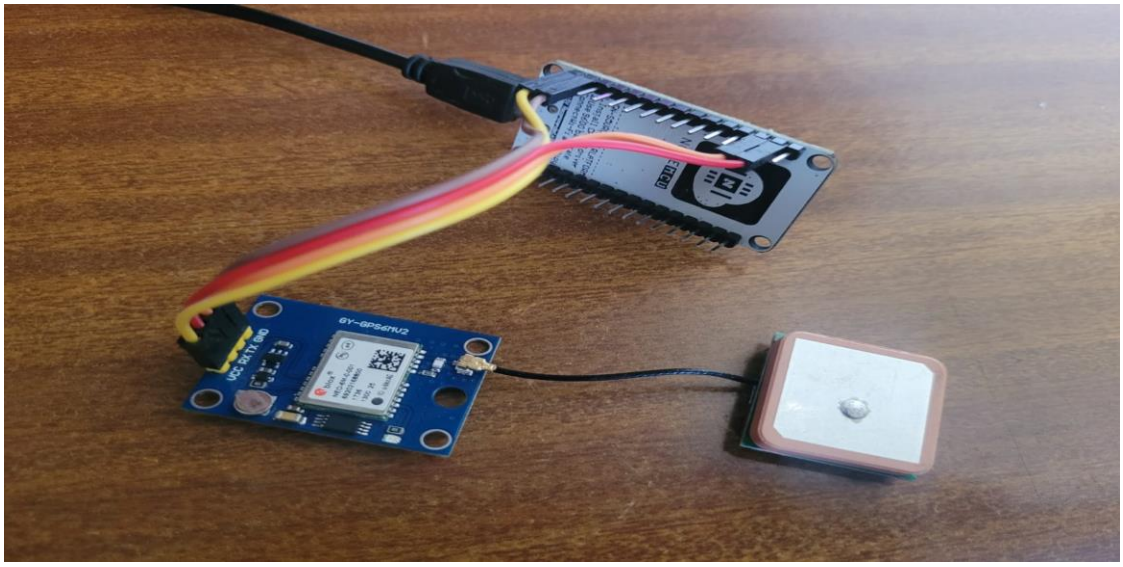
The pet owner has to just send a “Hi” and he/she will receive the reply message. The reply message contains the coordinates and a google map link of the tracker at that given time. This ensures safety of the pet and assurity to the pet owner that their pet is at the location he was dropped off to the pet lover.

7. Conclusion and Future Work

The Pet Sitter application is a complete system which can connect pet lovers and owners. The owner can track their pet through their phone by just sending a “Hi” message. Owners can enjoy their vacation while pet lovers can take care of these pets in their free time. The pet owners can review the pet carers making the system reliable.

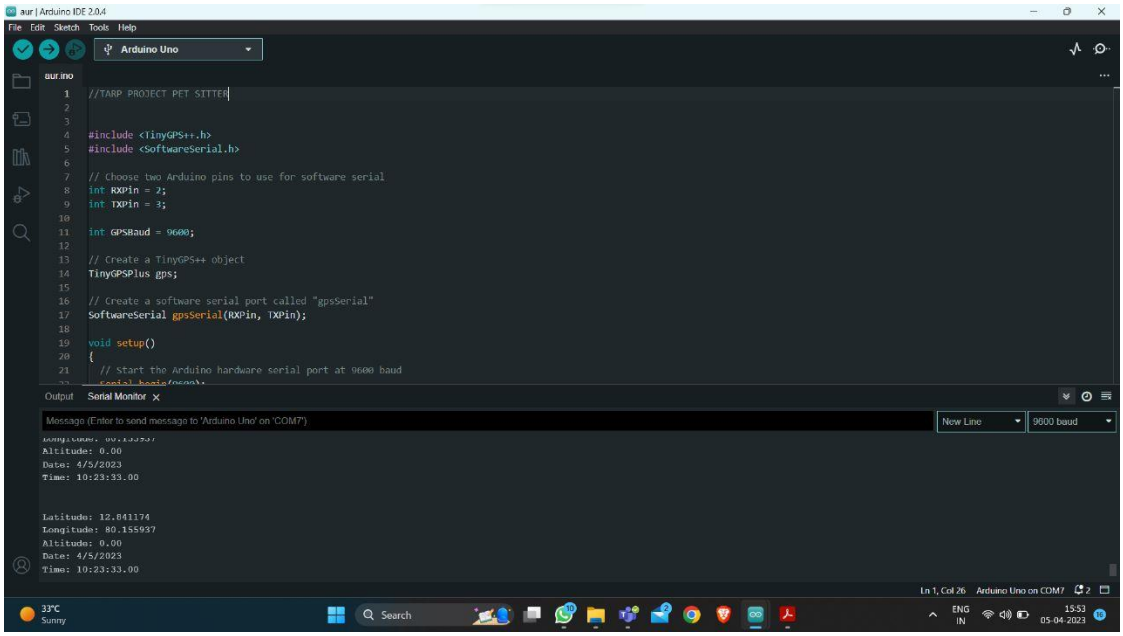
In the further versions we wish to make the tracker smaller and provide a casing for the tracker to make it wearable and aesthetic. Connecting pet owners to pet shops and doctors is what we wish to integrate in the Pet Sitter application.

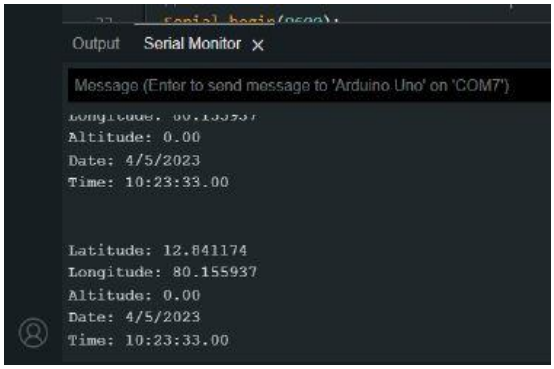
Tracker Setup-



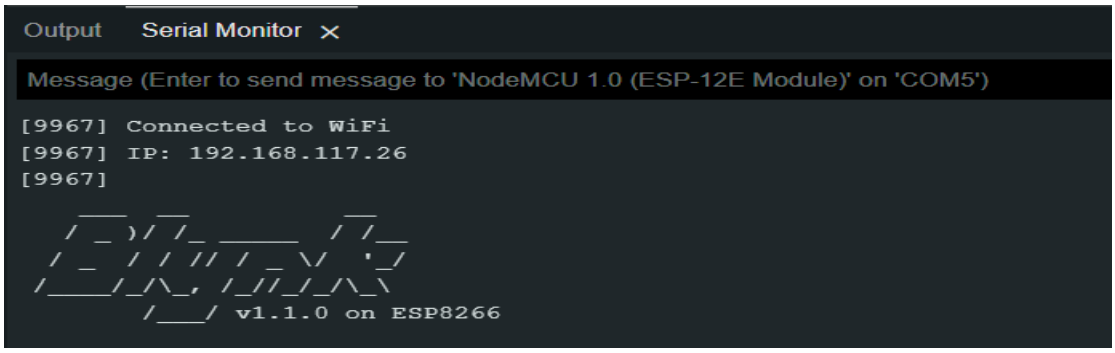
OUTPUT:

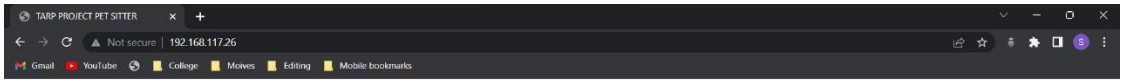
ARDUINO:





NODE MCU:





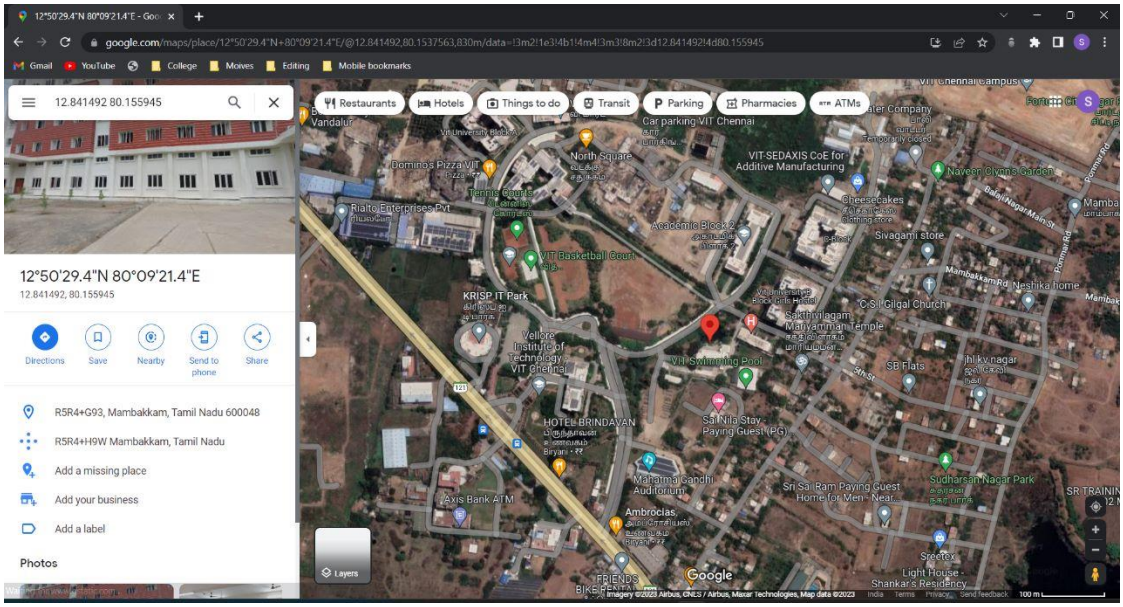
TARP PROJECT PET SITTER

Location Details

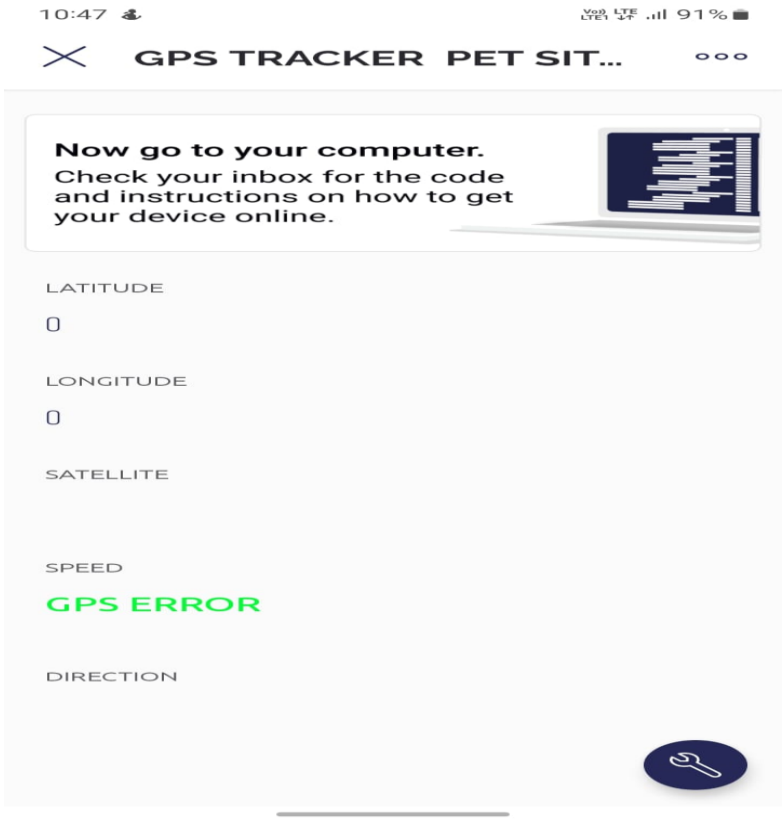
Latitude	12.841492
Longitude	80.155945
Date	05 / 04 / 2023
Time	04 : 00 : 18 PM

Click here! To check the location in Google maps.

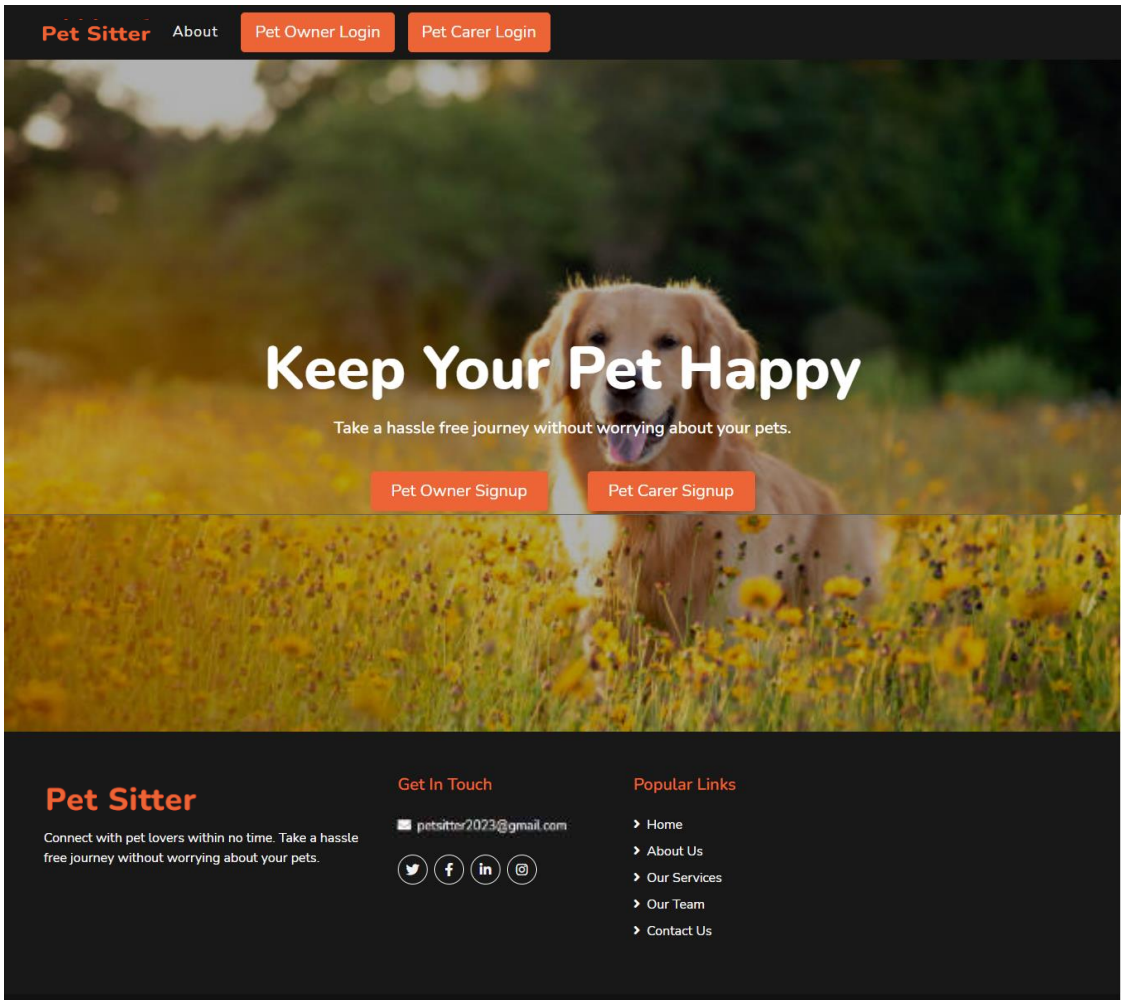
Viewing on the Google Map:



BYLNK:

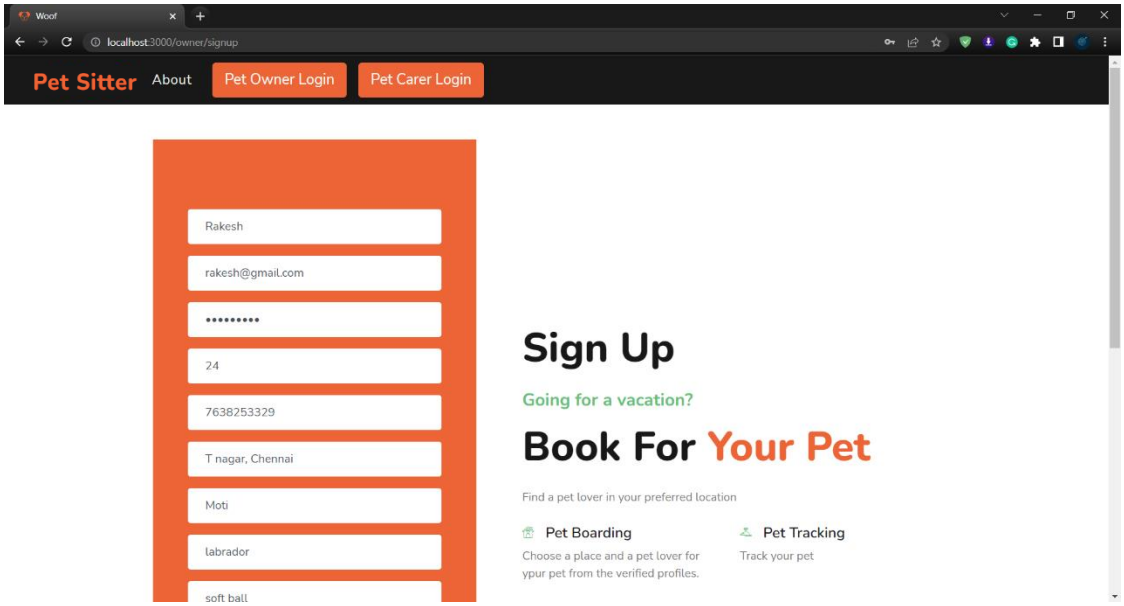


Home Page before signing-

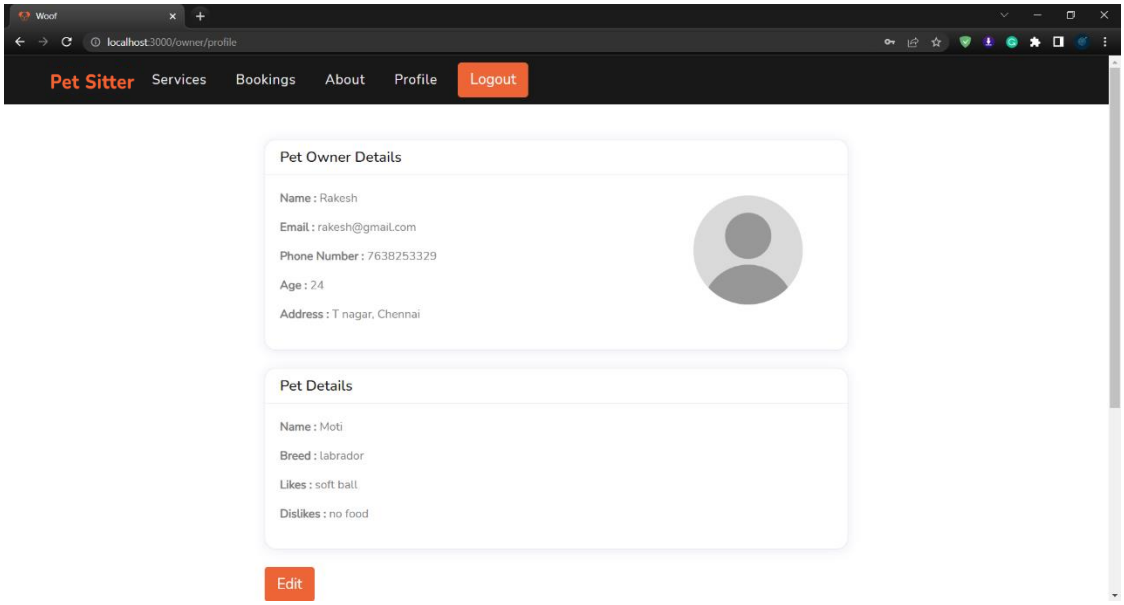


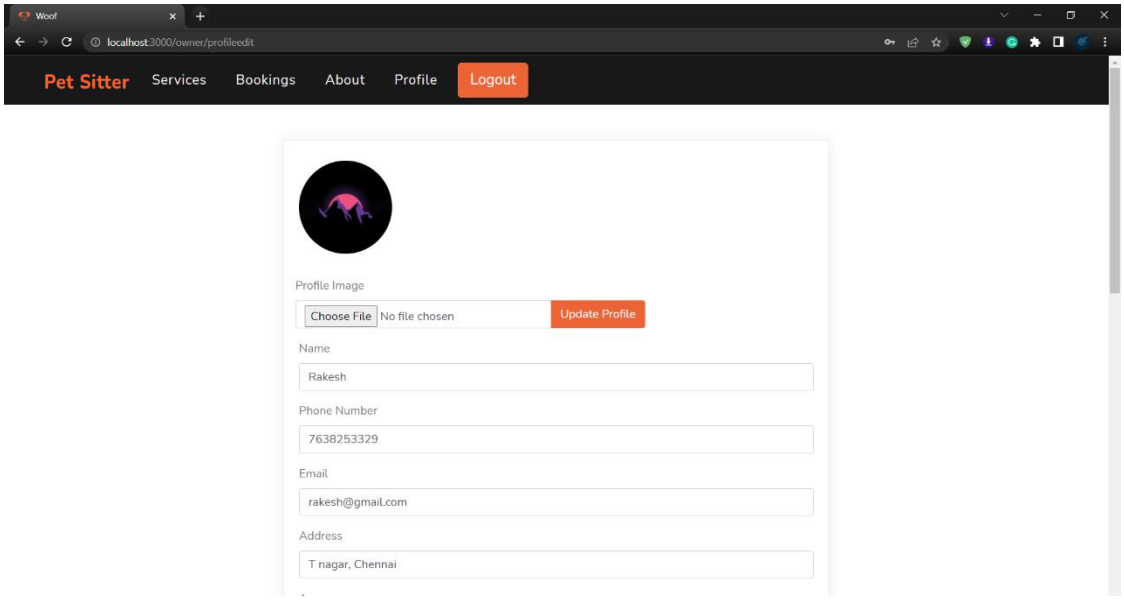
About Page-

Pet Owner SignUp-

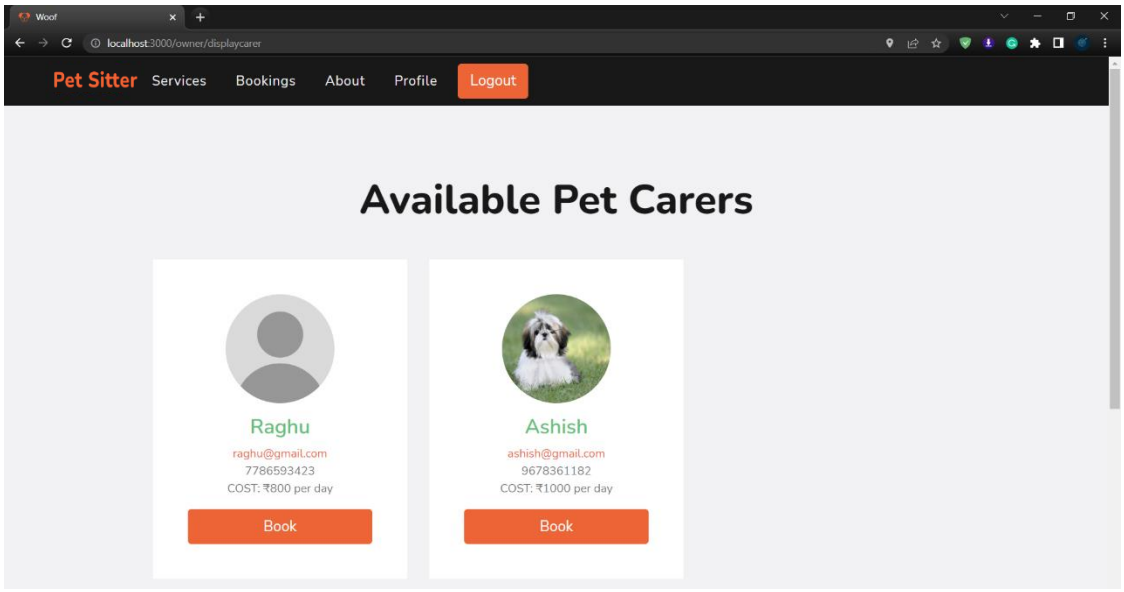


Pet owner profile page:

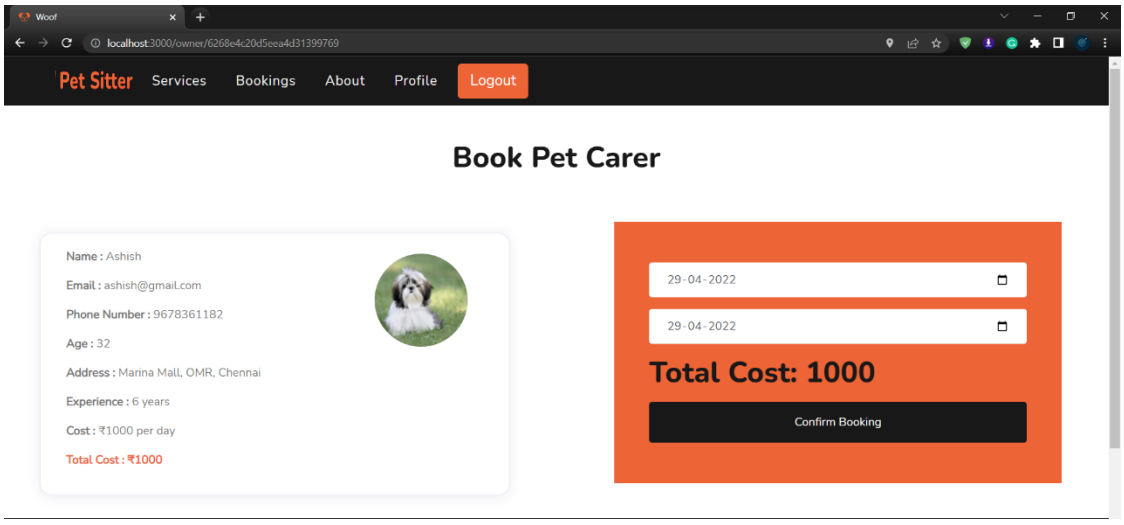




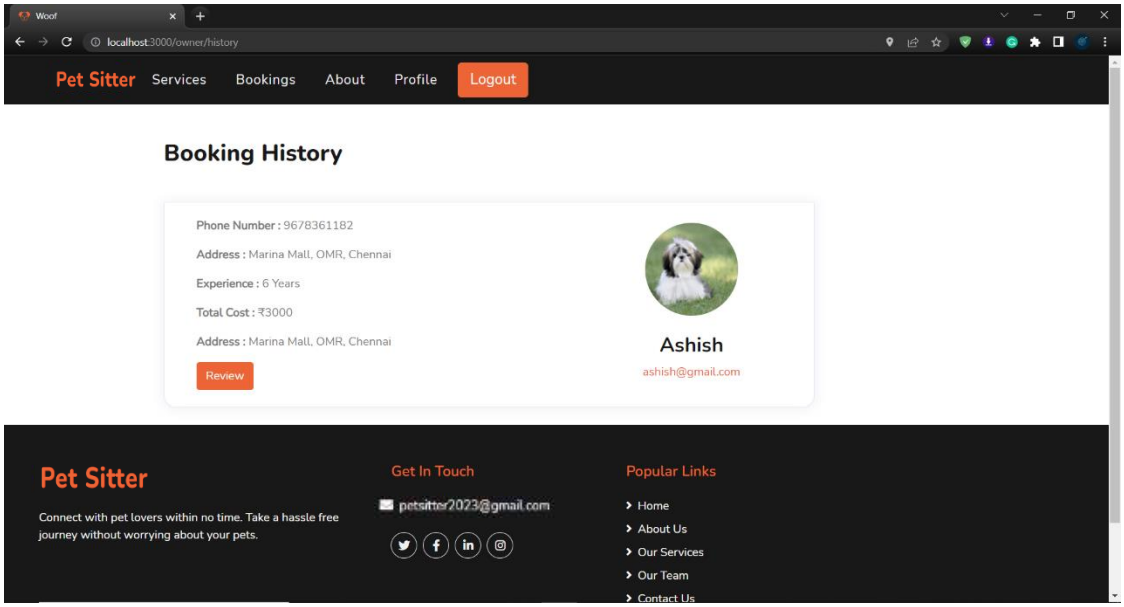
Pet carers near me display:



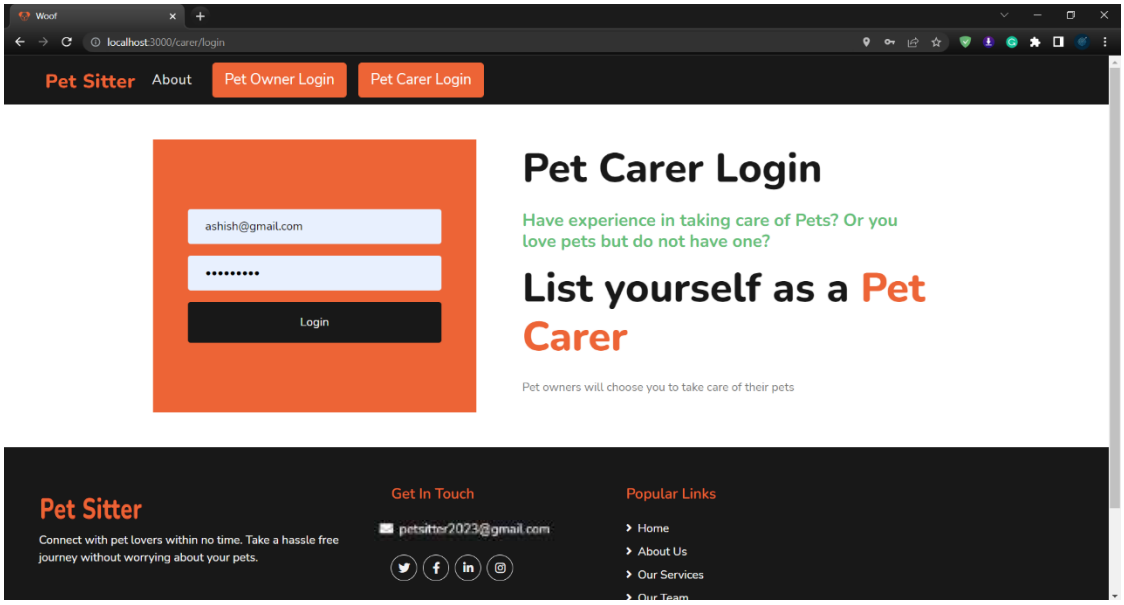
Booking pet carer:



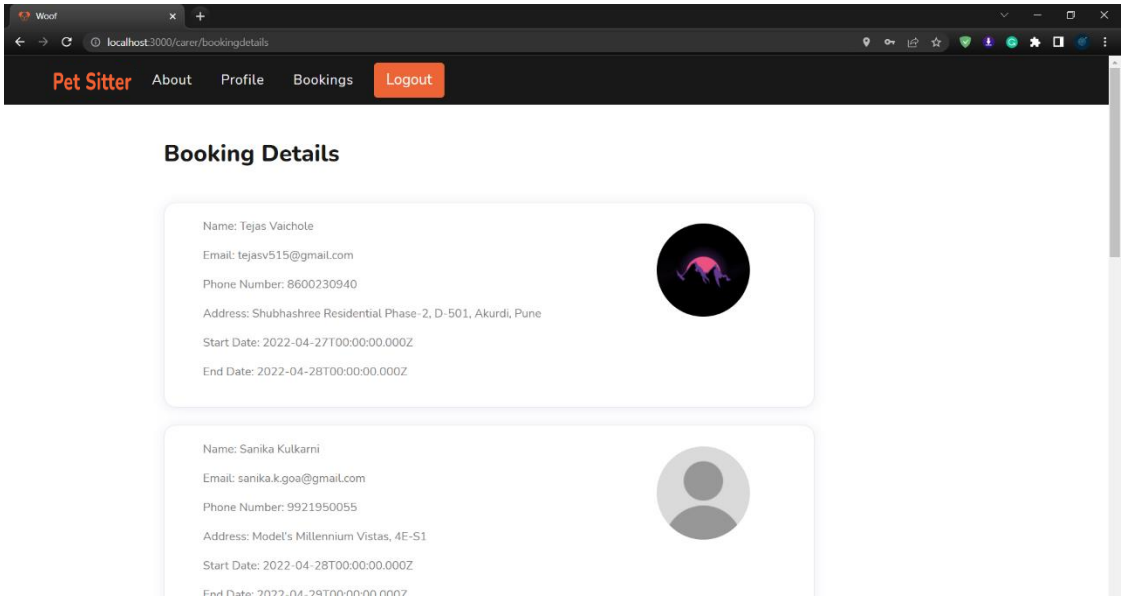
Pet owner booking history:



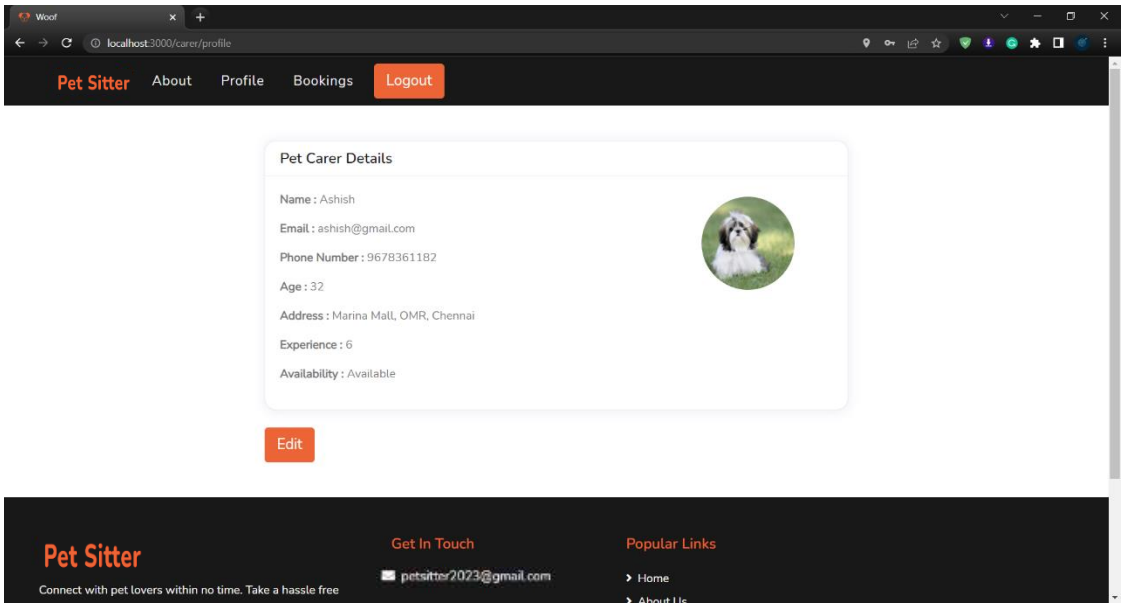
Pet carer SignIn:



Pet carer booking history:



Pet carer profile page:



8. REFERENCES

X. Ge, R. Gu, Y. Lang and Y. Ding, "Design of handheld positioning tracker based on GPS/GSM," 2017 IEEE 3rd Information Technology and Mechatronics Engineering Conference (ITOEC), 2017, pp. 868-871, doi: 10.1109/ITOEC.2017.8122477.

I. Ganchev, Z. Ji and M. O'Droma, "Designing a Low-Cost Location Tracker for Use in IoT Applications," 2020 XXXIIIrd General Assembly and Scientific Symposium of the International Union of Radio Science, 2020, pp. 1-2, doi: 10.23919/URSIGASS49373.2020.9232023.

J. Zhou, H. Yang and K. Anderson, "SNPP ATMS On-Orbit Geolocation Error Evaluation and Correction Algorithm," in IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 6, pp. 3802-3812, June 2019, doi: 10.1109/TGRS.2018.2887407.

L. Jiang, M. Cheng and T. Matsumoto, "A TOA-DOA Hybrid Factor Graph-Based Technique for Multi-Target Geolocation and Tracking," in IEEE Access, vol. 9, pp. 14203-14215, 2021, doi: 10.1109/ACCESS.2021.3052233.

M. Cheng, M. R. K. Aziz and T. Matsumoto, "Integrated Factor Graph Algorithm for DOA-Based Geolocation and Tracking," in IEEE Access, vol. 8, pp. 49989-49998, 2020, doi: 10.1109/ACCESS.2020.2979510.

APPENDIX:

ARDUINO CODE:

```
// TARP PROJECT PET SITTER

#include <TinyGPS++.h>
#include <SoftwareSerial.h>

// Choose two Arduino pins to use for software serial
int RXPin = 2;
int TXPin = 3;

int GPSBaud = 9600;

// Create a TinyGPS++ object
TinyGPSPlus gps;

// Create a software serial port called "gpsSerial"
SoftwareSerial gpsSerial(RXPin, TXPin);

void setup()
{
  // Start the Arduino hardware serial port at 9600 baud
  Serial.begin(9600);

  // Start the software serial port at the GPS's default baud
  gpsSerial.begin(GPSBaud);
}

void loop()
{
  // This sketch displays information every time a new sentence is correctly encoded.
```

```
  while (gpsSerial.available() > 0)
    if (gps.encode(gpsSerial.read()))
      displayInfo();

  // If 5000 milliseconds pass and there are no characters coming in
  // over the software serial port, show a "No GPS detected" error
  if (millis() > 5000 && gps.charsProcessed() < 10)
  {
    Serial.println("No GPS detected");
    while(true);
  }
}

void displayInfo()
{
  if (gps.location.isValid())
  {
    Serial.print("Latitude: ");
    Serial.println(gps.location.lat(), 6);
    Serial.print("Longitude: ");
    Serial.println(gps.location.lng(), 6);
    Serial.print("Altitude: ");
    Serial.println(gps.altitude.meters());
  }
  else
  {
    Serial.println("Location: Not Available");
  }

  Serial.print("Date: ");
  if (gps.date.isValid())
```

```
{
  Serial.print(gps.date.month());
  Serial.print("/");
  Serial.print(gps.date.day());
  Serial.print("/");
  Serial.println(gps.date.year());
}
else
{
  Serial.println("Not Available");
}

Serial.print("Time: ");
if (gps.time.isValid())
{
  if (gps.time.hour() < 10) Serial.print(F("0"));
  Serial.print(gps.time.hour());
  Serial.print(":");
  if (gps.time.minute() < 10) Serial.print(F("0"));
  Serial.print(gps.time.minute());
  Serial.print(":");
  if (gps.time.second() < 10) Serial.print(F("0"));
  Serial.print(gps.time.second());
  Serial.print(".");
  if (gps.time.centisecond() < 10) Serial.print(F("0"));
  Serial.println(gps.time.centisecond());
}
else
{
  Serial.println("Not Available");
}
```

```
  Serial.println();
  Serial.println();
  delay(1000);
}
```

NODE MCU CODE:

```
#include <TinyGPS++.h>
#include <SoftwareSerial.h>
#include <ESP8266WiFi.h>

TinyGPSPlus gps;  // The TinyGPS++ object

SoftwareSerial ss(4, 5); // The serial connection to the GPS device

const char* ssid = "sanjil";
const char* password = "sanjil12345";

float latitude , longitude;
int year , month , date, hour , minute , second;
String date_str , time_str , lat_str , lng_str;
int pm;

WiFiServer server(80);
void setup()
{
  Serial.begin(115200);
  ss.begin(9600);
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);

  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED)
  {
    delay(500);
```

```
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");

  server.begin();
  Serial.println("Server started");

  // Print the IP address
  Serial.println(WiFi.localIP());

}

void loop()
{

  while (ss.available() > 0)
    if (gps.encode(ss.read()))
    {
      if (gps.location.isValid())
      {
        latitude = gps.location.lat();
        lat_str = String(latitude , 6);
        longitude = gps.location.lng();
        lng_str = String(longitude , 6);
      }

      if (gps.date.isValid())
      {
        date_str = "";
        date = gps.date.day();
```

```
  month = gps.date.month();
  year = gps.date.year();

  if (date < 10)
    date_str = '0';
  date_str += String(date);

  date_str += " / ";

  if (month < 10)
    date_str += '0';
  date_str += String(month);

  date_str += " / ";

  if (year < 10)
    date_str += '0';
  date_str += String(year);
}

if (gps.time.isValid())
{
  time_str = "";
  hour = gps.time.hour();
  minute = gps.time.minute();
  second = gps.time.second();

  minute = (minute + 30);
  if (minute > 59)
  {
    minute = minute - 60;
```

```
  hour = hour + 1;
}
hour = (hour + 5) ;
if (hour > 23)
  hour = hour - 24;

if (hour >= 12)
  pm = 1;
else
  pm = 0;

hour = hour % 12;

if (hour < 10)
  time_str = '0';
time_str += String(hour);

time_str += " : ";

if (minute < 10)
  time_str += '0';
time_str += String(minute);

time_str += " : ";

if (second < 10)
  time_str += '0';
time_str += String(second);

if (pm == 1)
  time_str += " PM ";
```

```
      else
        time_str += " AM ";

     }

   }
  // Check if a client has connected
  WiFiClient client = server.available();
  if (!client)
  {
   return;
  }

  // Prepare the response
  String s = "HTTP/1.1 200 OK\r\nContent-Type: text/html\r\n\r\n <!DOCTYPE html>
<html> <head> <title>TARP PROJECT PET SITTER</title> <style>";
  s += "a:link {background-color: YELLOW;text-decoration: none;}";
  s += "table, th, td {border: 1px solid black;} </style> </head> <body> <h1  style=";
  s += "font-size:300%;";
  s += " ALIGN=CENTER>TARP PROJECT PET SITTER</h1>";
  s += "<p ALIGN=CENTER style=""font-size:150%;""";
  s += "> <b>Location Details</b></p> <table ALIGN=CENTER style=";
  s += "width:50%";
  s += "> <tr> <th>Latitude</th>";
  s += "<td ALIGN=CENTER >";
  s += lat_str;
  s += "</td> </tr> <tr> <th>Longitude</th> <td ALIGN=CENTER >";
  s += lng_str;
  s += "</td> </tr> <tr>  <th>Date</th> <td ALIGN=CENTER >";
  s += date_str;
  s += "</td></tr> <tr> <th>Time</th> <td ALIGN=CENTER >";
```

```
  s += time_str;
  s += "</td>  </tr> </table> ";


  if (gps.location.isValid())
  {
    s += "<p align=center><a style=""color:RED;font-size:125%;"" href=""http://maps.google.com/maps?&z=15&mrt=yp&t=k&q=";
    s += lat_str;
    s += "+";
    s += lng_str;
    s += """ target=""_top"">Click here!</a> To check the location in Google maps.</p>";
  }

  s += "</body> </html> \n";

  client.print(s);
  delay(100);

}
```

BYLNK CODE:

```
#define BLYNK_TEMPLATE_ID "TMPLtZldWJVH"
#define BLYNK_TEMPLATE_NAME "Quickstart Template"
#define BLYNK_AUTH_TOKEN "jZnJOnbRLc50uRTN4Z5rBwuH9vxvXRUj"

#include <TinyGPS++.h>
#include <SoftwareSerial.h>
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

static const int RXPin = 4, TXPin = 5;   // GPIO 4=D2(conneect Tx of GPS) and GPIO 5=D1(Connect Rx of GPS
static const uint32_t GPSBaud = 9600; //if Baud rate 9600 didn't work in your case then use 4800

TinyGPSPlus gps;      // The TinyGPS++ object
WidgetMap myMap(V0);  // V0 for virtual pin of Map Widget

SoftwareSerial mygps(RXPin, TXPin);  // The serial connection to the GPS device

BlynkTimer timer;

float latitude;     //Storing the Latitude
float longitude;    //Storing the Longitude
float velocity;     //Variable  to store the velocity
float sats;         //Variable to store no. of satellites response
String bearing;     //Variable to store orientation or direction of GPS

char  auth[] = "jZnJOnbRLc50uRTN4Z5rBwuH9vxvXRUj";                  //Blynk Authentication Token
```

```
char ssid[] = "sanjil";            // WiFi SSID
char pass[] = "sanjil12345";           // WiFi Password

//unsigned int move_index;         // moving index, to be used later
unsigned int move_index = 1;       // fixed location for now



void setup()
{
  Serial.begin(115200);
  Serial.println();
  mygps.begin(GPSBaud);
  Blynk.begin(auth, ssid, pass);
  timer.setInterval(5000L, checkGPS); // every 5s check if GPS is connected, only really
needs to be done once
}

void checkGPS()
{
  if (gps.charsProcessed() < 10)
  {
    Serial.println(F("No GPS detected: check wiring."));
    Blynk.virtualWrite(V3, "GPS ERROR");  // Value Display widget  on V3 if GPS not
detected
  }
}

void loop()
{
  while (mygps.available() > 0)
  {
```

```
    // sketch displays information every time a new sentence is correctly encoded.
    if (gps.encode(mygps.read()))
      displayInfo();
  }
  Blynk.run();
  timer.run();
}


void displayInfo()
{
  if (gps.location.isValid() )
  {
    sats = gps.satellites.value();       //get number of satellites
    latitude = (gps.location.lat());     //Storing the Lat. and Lon.
    longitude = (gps.location.lng());
    velocity = gps.speed.kmph();         //get velocity
    bearing = TinyGPSPlus::cardinal(gps.course.value());     // get the direction

    Serial.print("SATS:  ");
    Serial.println(sats);  // float to x decimal places
    Serial.print("LATITUDE:  ");
    Serial.println(latitude, 6);  // float to x decimal places
    Serial.print("LONGITUDE: ");
    Serial.println(longitude, 6);
    Serial.print("SPEED: ");
    Serial.print(velocity);
    Serial.println("kmph");
    Serial.print("DIRECTION: ");
    Serial.println(bearing);
```

```
    Blynk.virtualWrite(V1, String(latitude, 6));
    Blynk.virtualWrite(V2, String(longitude, 6));
    Blynk.virtualWrite(V3, sats);
    Blynk.virtualWrite(V4, velocity);
    Blynk.virtualWrite(V5, bearing);
    myMap.location(move_index, latitude, longitude, "GPS_Location");
  }
  Serial.println();
}
```

INDEX SAMPLE CODE:

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="utf-8">
    <title>Pet Sitter | Home</title>
    <meta content="width=device-width, initial-scale=1.0" name="viewport">
    <meta content="Free HTML Templates" name="keywords">
    <meta content="Free HTML Templates" name="description">

    <!-- Favicon -->
    <link href="img/favicon.ico" rel="icon">

    <!-- Google Web Fonts -->
    <link
href="https://fonts.googleapis.com/css2?family=Nunito+Sans&family=Nunito:wght@
600;700;800&display=swap" rel="stylesheet">

    <!-- Font Awesome -->
    <link                                href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.10.0/css/all.min.css" rel="stylesheet">

    <!-- Flaticon Font -->
    <link href="lib/flaticon/font/flaticon.css" rel="stylesheet">

    <!-- Libraries Stylesheet -->
    <link href="lib/owlcarousel/assets/owl.carousel.min.css" rel="stylesheet">
    <link              href="lib/tempusdominus/css/tempusdominus-bootstrap-4.min.css"
rel="stylesheet" />
```

```
    <!-- Customized Bootstrap Stylesheet -->
    <link href="css/style.css" rel="stylesheet">
</head>

<body>

    <!-- Navbar Start -->
        <nav class="navbar navbar-expand-lg bg-dark navbar-dark py-3 py-lg-0 px-lg-5">
            <div class="navbar-nav mr-auto py-0">
                <a href="index.html" class="navbar-brand d-none d-sm-block nav-item nav-link">
                    <h2 class="m-0 display-0"><span class="text-primary">Pet Sitter</span></h2>
                </a>
                <a href="service.html" class="nav-item nav-link">Service</a>
                <a href="about.html" class="nav-item nav-link">About</a>
                <a href="contact.html" class="nav-item nav-link">Contact</a>
            </div>


        <div style="display: flex; justify-content: flex-end">
            <a href="login.html" class="btn btn-lg btn-primary px-3 d-none d-lg-block">Login</a>


            <div class="container">

                <button type="button" class="btn btn-lg btn-primary px-3 d-none d-lg-block" data-toggle="modal" data-target="#myModal">Sign Up</button>

                <!-- Modal -->
```

```
<div class="modal fade" id="myModal" role="dialog">
  <div class="modal-dialog">

    <!-- Modal content-->
    <div class="modal-content">
      <div class="modal-header">
        <h4 class="modal-title">SignUp Options</h4>
        <button type="button" class="close" data-dismiss="modal">&times;</button>

      </div>
      <div class="modal-body">
        <a href="signupcare.html" class="btn btn-lg btn-primary px-3 d-none d-lg-block">Sign Up as CareTaker</a>  
        <a href="signuppet.html" class="btn btn-lg btn-primary px-3 d-none d-lg-block">Sign Up as Pet Owner</a>
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-default" data-dismiss="modal">Close</button>
      </div>
    </div>

  </div>
 </div>
</div>
</div>
</nav>
<!-- Navbar End -->
```

```
    <!-- Carousel Start -->
    <div class="container-fluid p-0">
        <div id="header-carousel" class="carousel slide" data-ride="carousel">
            <div class="carousel-inner">

                <div class="carousel-item active">
                    <img class="w-100" src="img/home.jpg" alt="Image">
                    <div class="carousel-caption d-flex flex-column align-items-center justify-content-center">
                        <div class="p-3" style="max-width: 900px;">
                            <h1 class="display-3 text-white mb-3">Keep Your Pet Happy</h1>
                            <h5 class="text-white mb-3 d-none d-sm-block">Take a hassle free journey without worrying about your pets.</h5>
                            <a href="" class="btn btn-lg btn-primary mt-3 mt-md-4 px-4">Create account now</a>

                        </div>
                    </div>
                </div>
            </div>
            <a class="carousel-control-prev" href="#header-carousel" data-slide="prev">
                <div class="btn btn-primary rounded" style="width: 45px; height: 45px;">
                    <span class="carousel-control-prev-icon mb-n2"></span>
                </div>
            </a>
            <a class="carousel-control-next" href="#header-carousel" data-slide="next">
                <div class="btn btn-primary rounded" style="width: 45px; height: 45px;">
                    <span class="carousel-control-next-icon mb-n2"></span>
                </div>
            </a>
        </div>
```

```
    </div>
    <!-- Carousel End -->

    <!-- Footer Start -->
    <div class="container-fluid bg-dark text-white mt-5 py-5 px-sm-3 px-md-5">
        <div class="row pt-5">
            <div class="col-lg-4 col-md-12 mb-5">
                <h1 class="mb-3 display-5 text-capitalize text-white"><span class="text-primary">Pet Sitter</span></h1>
                <p class="m-0">Connect with pet lovers within no time.
                    Take a hassle free journey without worrying about your pets.
                </p>
            </div>
            <div class="col-lg-8 col-md-12">
                <div class="row">
                    <div class="col-md-4 mb-5">
                        <h5 class="text-primary mb-4">Get In Touch</h5>

                        <p><i class="fa fa-envelope mr-2"></i>PetSitter2022@gmail.com</p>
                        <div class="d-flex justify-content-start mt-4">
                            <a class="btn btn-outline-light rounded-circle text-center mr-2 px-0"
                                style="width: 36px; height: 36px;" href="#"><i class="fab fa-twitter"></i></a>
                            <a class="btn btn-outline-light rounded-circle text-center mr-2 px-0"
                                style="width: 36px; height: 36px;" href="#"><i class="fab fa-facebook-f"></i></a>
                            <a class="btn btn-outline-light rounded-circle text-center mr-2 px-0"
                                style="width: 36px; height: 36px;" href="#"><i class="fab fa-linkedin-in"></i></a>
                            <a class="btn btn-outline-light rounded-circle text-center mr-2 px-0"
```

```
                    style="width: 36px; height: 36px;" href="#"><i class="fab fa-instagram"></i></a>
                </div>
            </div>


        </div>
      </div>
    </div>
    <div class="container-fluid text-white py-4 px-sm-3 px-md-5" style="background: #111111;">
      <div class="row">
        <div class="col-md-6 text-center text-md-left mb-3 mb-md-0">
          <p class="m-0 text-white">
            &copy; <a class="text-white font-weight-bold" href="#">Pet Sitterf2022</a>
          </p>
        </div>
      </div>
    </div>
    <!-- Footer End -->


    <!-- Back to Top -->
    <a href="#" class="btn btn-lg btn-primary back-to-top"><i class="fa fa-angle-double-up"></i></a>


    <!-- JavaScript Libraries -->
    <script src="https://code.jquery.com/jquery-3.4.1.min.js"></script>
```

```
    <script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/bootstrap.bundle.min.js"></
script>
    <script src="lib/easing/easing.min.js"></script>
    <script src="lib/owlcarousel/owl.carousel.min.js"></script>
    <script src="lib/tempusdominus/js/moment.min.js"></script>
    <script src="lib/tempusdominus/js/moment-timezone.min.js"></script>
    <script src="lib/tempusdominus/js/tempusdominus-bootstrap-4.min.js"></script>

    <!-- Contact Javascript File -->
    <script src="mail/jqBootstrapValidation.min.js"></script>
    <script src="mail/contact.js"></script>

    <!-- Template Javascript -->
    <script src="js/main.js"></script>
</body>

</html>
```