A System for Fair Rickshaw Fare Measurement in Bangladesh

by

Prioty Anika Tahsin
19101435
Mehjabeen Afroz Tanni
19101489
Mohaiminul Islam Meshal
16201043
Md. Mostaque Ahmed Ejaz
19101003
Md. Aosaf Ibad Chowdhury
19101492

A Project Submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
School of Data and Sciences
Brac University
March 24, 2023

Declaration

It is hereby declared that

1. The project submitted is my/our own original work while completing degree at Brac University.
2. The project does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The project does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Prioty Anika Tahsin
19101435

Mehjabeen Afroz Tanni
19101489

Mohaiminul Islam Meshal
16201043

Md. Mostaque Ahmed Ejaz
19101003

Md. Aosaf Ibad Chowdhury
19101492

Approval

The project titled "A System for Fair Rickshaw Fare Measurement in Bangladesh" submitted by

1. Prioty Anika Tahsin(19101435)
2. Mehjabeen Afroz Tanni(19101489)
3. Mohaiminul Islam Meshal(16201043)
4. Md. Mostaque Ahmed Ejaz(19101003)
5. Md. Aosaf Ibad Chowdhury(19101492)

Of Spring, 2023 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on March 24 2023.

Examining Committee:

Supervisor: (Member)

Md.Imran Bin Azad
Senior Lecturer
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Md. Golam Rabiul Alam
Program Coordinator
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

A rickshaw is a three-wheeled, environment-friendly, low-cost mode of transportation. They are one of the most efficient modes of transportation for moving people or goods across short to medium distances. The majority of people in Bangladesh utilize rickshaws as their primary mode of transportation for commuting as well as social and recreational trips. Since rickshaws don't have a specific fare system, the concept of an automated system for their usage is proposed. The goal is to develop a fare measurement system that will be advantageous to both rickshaw drivers and passengers. It is a microcontroller project based on Arduino that solves the problem of figuring out a fair rickshaw fare on a daily basis. The proposed model makes use of the IR sensor, accelerometer, and humidity sensor to measure variables like the rickshaw's distance and time of travel, the state of the road, and the condition of the weather. The system will calculate the fare considering these factors and show it on the LCD display, which will be recorded on a SD card in order to maintain a fair rate. In a test-based environment, this prototype is implemented with satisfactory results. Apart from calculating fares, this system will also serve as a method to study the dynamics of this physical labor.

Keywords: Rickshaw, fare, Arduino Uno, IR, automation, prototype, sensor, humidity, SD card, accelerometer, model, fair rate.

Acknowledgement

First and foremost, we want to express our gratitude to Allah, by whose help we have been able to successfully finish our project.
Secondly, we want to convey our gratitude to our supervisor, Mr. Imran Bin Azad for guiding us throughout the project. He was very helpful and cooperative all the time. His invaluable advice and support has helped us to reach our goal.
Last but not the least, thanks to all the friends and family for their unwavering support and motivation.

Table of Contents

List of Figures

Chapter 1

Introduction

In Bangladesh, rickshaws are a common mode of transportation, particularly in urban areas like Dhaka. They are widely used by people of all social classes and are considered a cost-effective and convenient way to get around the city. In the late 1800s, three Japanese men named Izumi Yosuke, Suzuki Tokujiro, and Takayama Kosuke created the first rickshaws in Japan. In a few tourist areas of Japan, rickshaws are still available for travelers. In our neighboring countries like India, Nepal, and Sri Lanka, there are mainly electric rickshaws. In Calcutta, there are hand-pulled rickshaws, especially in the tourist areas. However, cycle rickshaws are now most common in Bangladesh, where they hold social, economic, and cultural significance. Rickshaws are now a symbol that represents Bangladesh.

The actual number of rickshaws in Dhaka is estimated at 1.1 million, and 60 % of the population uses them for transportation. In 1941, the city had 37 rickshaws, 181 rickshaws in 1947, 112,572 rickshaws in 1998, 280,000 rickshaws in 2000, and 500,00 (Banglapedia) rickshaws in 2005. A rickshaw puller's monthly salary in Bangladesh is BDT 6300 on average and BDT 14000 at the highest level [1].

Rickshaw pulling can be considered an arduous job. In hot and crowded conditions, pulling a rickshaw can be physically demanding. Furthermore, as the fare increases daily due to the lack of a fixed rate, the suffering of the lower and middle classes has increased. The situation gets worse when rickshaw drivers won't haggle and demand a very high fee.

To solve this problem, we have developed a prototype that would determine the fare based on different factors. The prototype will give an approximate value, which will do justice to rickshaw pullers as well as passengers. The system will calculate the fare according to factors such as distance, steepness, time and weather.

Chapter 2

Problem Statement

In Bangladesh, rickshaws are among the most widely used modes of transportation, but they lack a fare measurement system, which causes problems like constant haggling between drivers and passengers which causes a great deal of significant time waste, problems with trust, and frequent injustice toward the drivers. However, it's also not uncommon to see passengers being charged unfairly. This is a domestic issue and has not been addressed yet. The problems would be addressed and resolved by the implementation of a fare measurement system for the rickshaws of Bangladesh.

2.1 Motivation:

The motivation behind creating this system is to address the day-to-day problem that is occurring when it comes to determining the fair rickshaw fare. The problem occurs between the rickshaw puller and the passenger. Bangladesh's economy has suffered greatly as a result of the covid-19 pandemic in the most recent years (2020–2022). The people that have been affected most by this are those in the middle and lower classes. In addition, the most common users of rickshaws are middle-class people, and the rickshaw pullers are of the lower class. Considering the severity of the economic crisis, almost everyone in these two classes has been earning less than they used to. As a result, the rickshaw pullers will look to earn more by asking for extra fares than usual. On the other hand, passengers will try to find a cost-effective way by bargaining for a lower fare. Most of the time, this bargaining leads to unpleasant conversations and behaviors among the drivers and passengers. Due to this the valuable time of both rickshaw pullers and passengers get wasted.

Following the issue, sometimes the hard labor done by the rickshaw pullers goes unnoticed. Weather conditions and road conditions also come into play. Unlike taxis and other vehicles, rickshaws do not have any fare measurements for weather or road conditions. Most of the time, they ask for a random fare for this, which might affect the regular fare and the passengers' condition as well. Furthermore, a big part of the transportation system and the economy depend on this.

To support this environmentally friendly vehicular system, our main goal is to provide a fair and reasonable fare for both parties. We are aiming to develop an algorithm that will determine an appropriate and reasonable fare based on some important factors. The features are total traveled distance, steepness of the road, weather condition, and duration of travel. All the data for these features will be collected using different sensors. The implementation of this kind of system and feature has not been done before, which makes our research work more challenging and exciting.

2.2 Project Objectives:

- Supporting an environmentally friendly vehicular system in our country.
- Addressing the intense physical labor done by the rickshaw pullers and developing a proper fare calculation for Rickshaws.
- Reducing day-to-day chaos among passengers and drivers.
- A new field of research and implementation in the country.

Chapter 3

Detailed Literature Review

Rickshaws are the most common vehicle in our country for public transportation. It has been a signature vehicle for our country for ages, as it has always carried a traditional and cultural value. Even though this vehicle bears cultural significance, we still couldn't find a systematic fare measurement system for this vehicle. There is some research that has been done theoretically but has yet to be implemented practically.

In the publication of Akter, S and Jui, Tahmina T and Athaya, T and Zaman, A and Rafi, S named A proposed system for fare measurement for Rickshaws of Bangladesh [2] where there was a proposal for an automated system that will measure various factors and calculate a proper fare that will eradicate bargaining and reduce hassle for both rickshaw pullers and passengers. In the proposed system, the distance will be determined by the rotation of the wheels, which has been developed as a prototype for the rickshaw fare measurement system. This prototype will also calculate the steepness of the road based on the gradients of the road. Other important factors like travel time, carrying weight, and relative humidity will also be measured to set a fair fare. They proposed using sensors like accelerometer, humidity sensor, and load sensor to get values from those different factors. They also implemented a feedback system where the opinions of the rickshaw pullers will be stored. The main drawbacks of this system are that the hardware products they used are not good enough for the present world. We intend to use the system they have suggested as a starting point to develop a practical fare measurement system for rickshaws in order to address the issues already associated with the absence of a suitable fare measurement system.

Another paper named Smart Fare System for CNG was done by Alok, S and Haque, Md E and Junayed, Rashidul H and Tasnim, R. proposed an automated system that will measure distance with the help of a GPS tracker and calculate a fair fare for CNGs in Bangladesh [4] to eradicate a mutual problem with ours: bargaining and overcharging. In their proposed system, they also used an RFID card to start and finish the process, and also to make the payment. After swiping the card to its reader, it will start counting the distance through a GPS tracker and upon receiving this data, Google map API will show the exact location. They used the GSM900A module to transfer the data to the server from where they got all the necessary information. Swiping the RFID card for the second time will stop the counting and provide the calculated distance. Based on that and their fare rate the passenger will be charged automatically from their account. Now, a system like this won't be applicable for rickshaws, and even if it is applicable, it'll be overkill as rickshaw rides can be really short in most cases. The requirement of an RFID card is another obstacle, as it is not mainstream in our country. Making people carry around an extra card linked to an account just for the payment of rickshaw rides is irrational. Moreover, this system does not consider any other factors other than distance for calculating the fare like weather, steepness of the road, etc. These factors are really important in the case of rickshaws, so our proposed system will consider these factors for calculating a fair fare.

In a patent document, D. Pomerantz named the Taxi trip meter system with indication of fare and distance violations [3] showed a system for the fare calculation of taxis by measuring distance and duration. It is a meter-based system where the driver will set the destination when there is a passenger in the taxi. Then the meter will calculate the shortest route to reach the destination. There will be a location sensor attached to the meter, which will be used to navigate the shortest path. It is very important that the driver follows this shortest path. Otherwise, there will be a violation message against the driver and the commuter can file a report against the driver if he/she wants. This system will also count the duration of the total time of traveling. In this calculation process, the average speed of the taxi will also be counted. Then, after reaching the destination, the actual fare will be calculated. Now, this system is not as applicable for rickshaws as it is for taxis. Moreover, this system only counts distance and duration, whereas the system we are trying to implement will cover various important factors like steepness, temperature, humidity, time, as well as the distance. So, it is a much more sophisticated version in terms of calculating fare. Besides, we are not trying to implement the shortest possible route finding system, as rickshaws are mainly used to travel short distances.

In a nutshell, all of these previous works were mostly done to find a proper method to calculate fares for public vehicles like CNG and taxis. One of the researches was done to calculate Rickshaw fare considering various factors into account.However, as was already stated, the hardware and cost of their suggested system are inappropriate for the project. Our research primarily uses the theories they have proposed and practically implements the system, bringing some improvements to that system with some new factors and hardware updates to eliminate daily haggling and establish a fair fare measurement system. Our goals are to use the hardware that is currently most appropriate for the project while also ensuring that the system is cost effective and hence affordable. The simplicity of our system will make it easy for the rickshaw puller and the passengers to use it.

Chapter 4

System Analysis and Requirements

We are aiming to build a cost-effective fare measurement system for manual rickshaws. However, it can also be modified for auto rickshaws if wanted. We are focusing on manually driven rickshaws, as they require a lot of physical strength and hence should have a proper digital measurement system. As previously said, rickshaws are the most popular mode of short-distance transportation, but the lack of a suitable measurement method creates issues like bargaining and unfair payment. Therefore, in order to end the misery of both passengers and rickshaw drivers, we aim to digitize the process of measuring the fare. After a successful launch of manual rickshaws, we also wish to use our technology on e-rickshaws.

In our microcontroller-based system, we will use an Arduino UNO R3 equipped with an IR sensor, an accelerometer, and a humidity sensor to measure various factors such as time, distance, steepness, and the temperature and humidity of the weather. We also wish to add wind resistance as a new factor to our system. The calculation of the wind against the direction of the rickshaw is another aspect of determining the fare that we believe is crucial. To detect the wind speed and direction, we will use a sensor called an anemometer. Sometimes it is very hard to pull a rickshaw when the rickshaw puller is pulling it against the direction of the wind. That's why we also considered this factor as well as others. So, we will calculate the wind speed that is flowing against the direction of the wind. After the completion of the ride, our system will calculate a fair fare considering all of these factors.
First of all, we will use the Arduino Uno R3 as the microcontroller. The sensors that we will use will be connected to the Arduino board. All the computation will be done there, and then the results will be forwarded to the LCD screen

4.1 Measuring Distance with IR Obstacle Sensor:

The fare is mostly dependent on the distance traveled by the rickshaw. It is the key factor in determining the fare. We will use the IR obstacle sensor on the wheel of the rickshaw to measure the distance. The device will turn on upon pressing the power button. Then, the rotations of the wheels will be calculated in a loop. With every rotation of the wheel, the rickshaw crosses a distance and the number of rotations gets counted. With each rotation, we have programmed an Unit Price value, which will be multiplied with each rotation using the rotation counter. We can see the following equation,

$$\text{Distance\_Fare} = \text{R} * \text{RP} \tag{4.1}$$

Here,
R= number of rotations,
RP= Unit Price for Rotation

4.2 Time of Travel:

The time of travel also starts being calculated upon pressing the start button on the LCD display. The elapsed time during travel will be displayed on the screen. The elapsed time is calculated and shown using the millis function of the built-in library of Arduino. Upon completion of the ride, the total time will also be shown.

4.3 Measuring the Steepness of the Roads with Accelerometer:

Then, the condition of the road will also be taken into consideration. The steepness of the road determines the amount of physical strength that has to be exerted. If the road is damaged or has holes, pulling a rickshaw becomes very challenging. It is also very challenging to pull a rickshaw up a steep. For this reason, measuring the road's condition is crucial. Extra fare will be added if the road is steeper than usual. We will use a 6 DOF accelerometer Gyroscope to calculate the steepness. The slope of the road determines the steepness. The higher the gradient, the harder it will be to pull the rickshaw. The ride also becomes quite uncomfortable on bumpy roads. According to a report, the slope of the roads in Bangladesh is between (0%-7%) [5]. A positive slope indicates going upward or uphill. It is hard to pull a rickshaw with passengers in it when the slope is positive, that is the steepness of the road. For this reason, when the Gyroscope finds the slope towards pitching up or down, the charge will increase through adding the Unit Price for steepness which can be expressed through the following equation,

$$\text{Steepness\_Fare} = (\text{S} * \text{SP}) \tag{4.2}$$

Here,
S= Value of Steepness,
SP = Unit Price for Steepness

The Gyroscope sensor sends the change in the x-axis values constantly to the Arduino. The changes of these values then gets averaged to a single value. Only the positive values will be used to calculate the average and the negative values will automatically get discarded. Then the average value gets multiplied by an arbitrary value set by us, which also can be changed if needed.

4.4 Calculating Relative Humidity with Humidity Sensor Module DHT11:

Moreover, the fare will also vary depending on the weather conditions. For instance, if it's raining or extremely sunny outside, the fare will be higher. The percentage of humidity will therefore be measured using the relative humidity sensor. Humans can live comfortably in a humidity range of 30–60% [6]. Hot environments tend to be more humid. We also perceive heat as being higher than it actually is when there is high humidity. Likewise, we feel cooler than it actually is when the humidity is very low. Relative humidity is 65.8% on average each year in Bangladesh. The monsoon season, which lasts from June to October, is when the humidity is at its highest [7]. The condition in Dhaka is the worst, with an average annual percentage of humidity of 74.0% [8]. The weather in Bangladesh has become one of the most difficult problems to handle as a result of pollution. With such living conditions, the weather must be taken into consideration while calculating the rickshaw fare. The equation mentioned below can be used in this regard [2]

$$\text{Weather\_Fare} = (\text{tmp} * \text{TP}) + (\text{hdt} * \text{HP}) \tag{4.3}$$

Here,
tmp = Average Temperature Value,
TP = Unit Price for Temperature,
hdt = Average Humidity Value,
HP = Unit Price for Humidity

$$\text{Total\_Fare} = \text{Distance\_Fare} + \text{Weather\_Fare} + \text{Weather\_Fare} \tag{4.4}$$

Another approach can be a ranged system. E.g. For a temperature value of 15-25°C no extra fare will be added. For, 10 - 14°C : +2 BDT. For, <10 : +4 BDT. For 26 - 35°C : +2 BDT and for >35°C : +4 BDT

4.5 Calculating Resistance Against Wind using Anemometer:

In addition to the factors mentioned above, we also wanted to calculate and add resistance against wind as one of the factors in calculating the fare. Due to limitations it could not be implemented. However, the findings and approaches taken for implementation of wind resistance calculation are being discussed for the betterment of future research on the topic. When the wind is blowing strongly, pulling a rickshaw against the wind is physically challenging. In this case, we can use an anemometer to measure the speed of the wind. It is possible to measure the direction and speed of the wind using an anemometer. Up to 70 m/s of wind speed can be measured by the Adafruit anemometer [9]. This anemometer can be connected to an LCD screen and an Arduino board. The sensor will measure the wind speed in m/s, and the value will be used to determine the fare, which will then be displayed on the LCD

COMPONENT NAME	MODEL	PRICE (BDT)	QUANTITY
LCD	LCD Keypad Shield for Arduino	550	1
Arduino	Arduino Uno R3	840	1
6 DOF Accelerometer Gyroscope	MPU-6050	247	1
IR Obstacle Sensor	Lm 393	178	1
Humidity and Temperature Sensor	DHT11	180	1
Breadboard	MB102	129	1
SD card and Arduino SD card Module		400	1
Jumper wires		75	25
		TOTAL=2599	

Figure 4.1: System Cost

screen.

$$windspeed = (n * 2\pi r)/t \tag{4.5}$$

Here,
n = number of rotations
r = distance from center to IR sensor,
t= time of travel

4.6 LCD Screen to Display the fare:

Finally, an LCD screen is used to display the calculated fare. The LCD Keypad Shield LCD Display has a user-friendly interface that allows users to display what they want and make selections using buttons as well.

We have chosen and collected all the required components, keeping in mind that the system has to be cost-effective. The system cost table contains the list of components used along with their price and quantity.

Chapter 5

System Design

5.1 Arduino Uno

Arduino uno is a standard microcontroller board based on a microchip technology known as ATmega328P. It contains a 16 MHz ceramic resonator, 6 analog pin inputs, 14 digital input/output pins (of which 6 can be used as PWM outputs), a USB port, a power jack, an ICSP (In-Circuit Serial Programming) header, and a reset button [11]. Its purpose is to design various kinds of electrical circuits by fusing a physical microcontroller-based programmable circuit board with computer code via a USB connection. Arduino Uno uses a simplified version of the C++ programming language, which makes it simple to replace hundreds of wires with codes. It is an open-source electronics platform with simple hardware and software that can run on both online and offline platforms.
In our system, we are using the Arduino UNO R3. It has all the components required to support the microcontroller. It can be powered by an AC-DC battery or by connecting to a computer by USB cable. Its operating voltage is 5 volts.

Reason of using Arduino UNO:
Arduino UNO is easy to use and to program as well which makes it ideal for beginners. The cost is lower compared to other development boards. It has a large and active community of users and resources. It's suitable for projects with power constraints due to low power consumption. It's also compatible with a wide range of sensors, actuators, and other components.

5.2 IR Obstacle Sensor

An IR sensor is an electronic device that emits infrared radiation with a view to perceiving certain aspects of its environment. Infrared radiation is produced by anything that radiates heat, and by using it, this sensor can detect the motion of any object. In addition to that, it can measure an object's heat. The module has five key components: TX, RX, an operational amplifier, a trimmer pot (a variable resistor), and an output LED. The IR sensor module has three pins: VCC for power supply input, GND for ground, and OUT for digital output. The main specifications and characteristics of the IR sensor are that the working voltage is 5 VDC and the input/output pins are 3.3 or 5 V. There is a mounting hole ranging up to

20 centimeters, and the supply current is 20 mA. Moreover, it has an ambient light sensor with an adaptable and fixed range of sensing. .

In our system, first of all, we needed to establish a connection between the IR sensor and the Arduino Uno, which is connected with a LCD keypad shield. Two jumper wires were respectively connected to the 5V and GND ports of the Arduino from the VCC and GND pins of the IR sensor through a breadboard. For getting the output, a wire was connected between the digital 2 port of the Arduino and the out pin of the IR sensor.

5.3 Gyroscope

A gyroscope is a device for evaluating or maintaining orientation and angular velocity. It is a revolving wheel or disc that can spin in any direction due to the flexibility of the spin axis. According to the principle of angular momentum conservation, the orientation of this axis is constant while the mounting is rotating or tilting. We have used a 6 DOF Accelerometer Gyroscope GY-521 in our prototype. For interfacing the gyroscope, there were four wires connected between the Arduino and the sensor itself. Like the IR sensor, the VCC and GND ports of the gyroscope were connected to the 5V and GND ports of the Arduino Uno. The SCL and SDA ports of the gyroscope were connected to the analog A5 and A4 of the Arduino Uno. Finally, we can see the output on the LCD keypad shield that is mounted on the Arduino Uno.

5.4 DHT11 Sensor Module for Arduino

The DHT11 is a fundamental digital temperature and humidity sensor that is quite inexpensive. It generates a digital signal on the data pin and uses a thermistor and a capacitive humidity sensor to measure the air's humidity. It does not require analog input pins. Compared to an analog sensor, it produces a temperature reading that is far more accurate. The digital signal from the DHT-11's output can be read using the digital I/O pins on an Arduino board. The sensor features an 8-bit microcontroller for serial data output of temperature and humidity information as well as a dedicated NTC (Negative Temperature Coefficient) for temperature measurement. The sensor also makes it simple to integrate with other microcontrollers due to its being factory calibrated. The sensor can measure temperature from 0°C to 50°C and humidity from 20% to 90% with an accuracy of ±1°C and ±1% [14].
The sensor module has 3 pins (Ground, DATA and VCC), as illustrated in fig. 5.5: DHT11 Sensor, however, the sensor is packaged in a 4-pin configuration, only three of which can be used. The key difference between the sensor and module is that the latter includes an internal pull-up resistor and filtering capacitor, while the former requires us to use them externally as needed. [14]. These facts are the reason the DHT11 sensor module is being used in our project.

- **Sensing Humidity:** The resistive component of the DHT-11 consists of two electrodes and a substrate that traps moisture in between them for the purpose of sensing humidity. The substrate discharges ions when it draws in water vapor, which increases the conductivity between the electrodes. Higher relative humidity leads to a drop in the resistance between the electrodes, whilst lower relative humidity results in an increase in that resistance. This variation in resistance is proportional to the relative humidity [15].

- **Sensing Temperature:** The DHT-11 uses a thermal resistance called an NTC Thermistor to measure temperature. The thermistor found inside the DHT11 has a negative temperature coefficient, which causes its resistance to drop as the temperature rises [15].

Direct Arduino interfacing is possible with the DHT-11 sensor module. For measuring humidity and temperature, we used DHT-11. There are two variants of this sensor. One with 4 pins and the other with 3 pins. We used the 3-pin variant. The VCC and GND pins of this sensor were connected to the Arduino's 5V and GND ports as usual. To get the output we connected a wire between the digital 4 port of the Arduino and the data pin of the DHT11.

5.5 LCD Keypad Shield

An LCD Keypad Shield is a shield or expansion board for the Arduino microcontroller board that adds a 16x2 LCD display and a 6-button keypad. The shield allows the user to easily display text and characters on the LCD screen and input data using the keypad. It can display up to 16 characters per row and 2 rows, and the user can read in any lighting condition as the display is typically backlit. Those buttons on the keypad are arranged in a matrix configuration. In our keypad shield, the buttons are labeled as reset, right, up, down, left, and select. It communicates with the Arduino using the I2C protocol. Multiples wires are used to connect the Arduino with LCD in order to display the output on it. the wires are plugged in the digital 4-9 ports of the Arduino Uno to interface the LCD keypad shield. To power up the LCD, the VCC and GRND. To read the six push buttons we connected a wire with the analog 0 port of the Arduiono Uno. This implies that the LCD and keypad may be connected to the Arduino board using just two of the digital I/O pins, enabling the other pins for other uses. There is also a potentiometer attached to the LCD keypad shield, which can control the brightness of the display. Users can modify the brightness by turning the knob of the potentiometer to regulate the signal. To operate the keypad shield, there is an Arduino library, which includes a set of functions such as printing text, controlling the cursor, and adjusting the contrast of the display. Overall, the LCD keypad shield is very convenient when the user will need a display and an input device in any Arduino project.

5.6 Anemometer (DIY)

An anemometer is a device that is mainly used to measure wind speed. It includes various cups on a vertical shaft. When the wind blows, these cups begin to rotate anticlockwise; the faster the wind blows, the faster the rotation. We are using this device because wind has a considerable impact on the rickshaw puller.
In our initial development of the prototype we tried to make an anemometer. We have made a DIY anemometer to measure the wind speed. We have used four cups, a PVC pipe, a bearing, some small sticks, and glue to build the physical device. Additionally, we have attached an obstacle in front of an IR sensor to measure the

cups' rotation. The pipe is attached to the bearing in which the cups are rotating. Every time the cups rotate around the small obstacle, the IR sensor detects that. When it is rotating at a higher speed, that means there's a strong wind blowing in the surrounding area. This is how this device will measure the wind speed. However, the system we built is very basic and also in a primary stage of development.

5.7 Power Supply

We have used two power banks to supply power. One of the convenient features of the Arduino UNO is that it can be connected to any external power source through a USB. So, we have securely mounted the power bank in the vehicle around the prototype. However, lithium ion batteries or other types of rechargeable batteries could be used to power the micro-controllers when the prototype is mass-produced.

5.8 Memory Module

We have used a memory card in the fare meter system in order to keep record of the previous ride fares. The fares can be compared and reviewed and this serves as a basis for maintaining a fair calculation of fare. The SD card module has been interfaced with the Arduino. Utilizing the built-in 3.3V regulator, an SD Card Reading and Writing Module for Arduino was used. MOSI, MISO, CS and SCK are all outputs on the SD SPI pins. The SD card has been installed and programmed to keep track of the fares. Using the up and down buttons on the LCD screen, the previous fares can be viewed. Diagram 5.13 shows the connection detail of SD card module with Arduino Uno. The pin numbers are: MOSI - pin11, MISO - pin 12, CLK - Digital pin 13 and CS - Digital 10.

5.9 Chassis

The entire system that is the is the micro controller board along with the sensors are placed inside a chassis box to make the system secure. The figures 6.6 and 6.7 shows the final outlook of the system in a chassis. The LCD display is placed in such a way in the box that can be clearly viewed and the buttons can be used from the top. This meter in a chassis can easily be mounted on a rickshaw using clamps or any other suitable way.

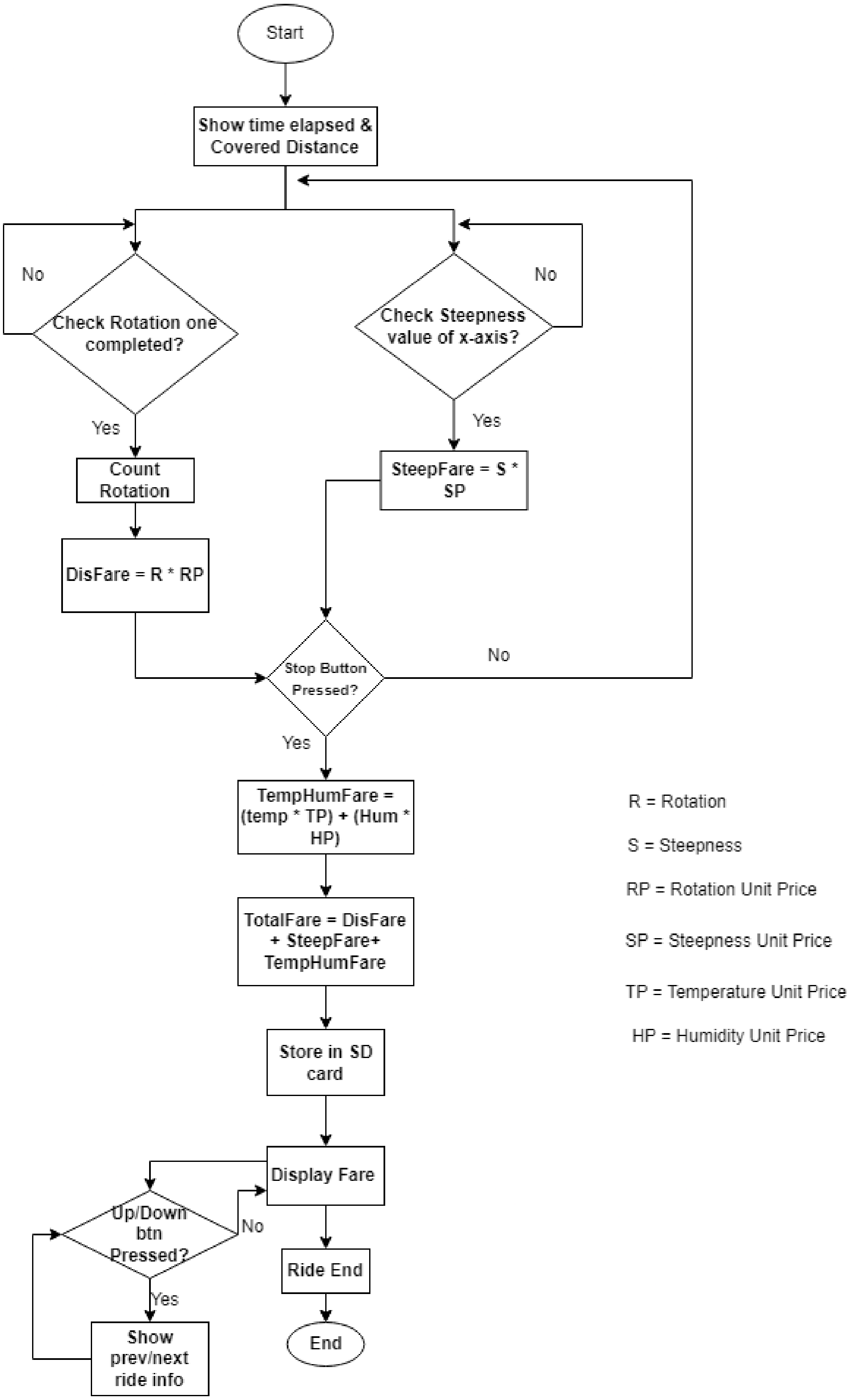

Figure 5.1: Flowchart

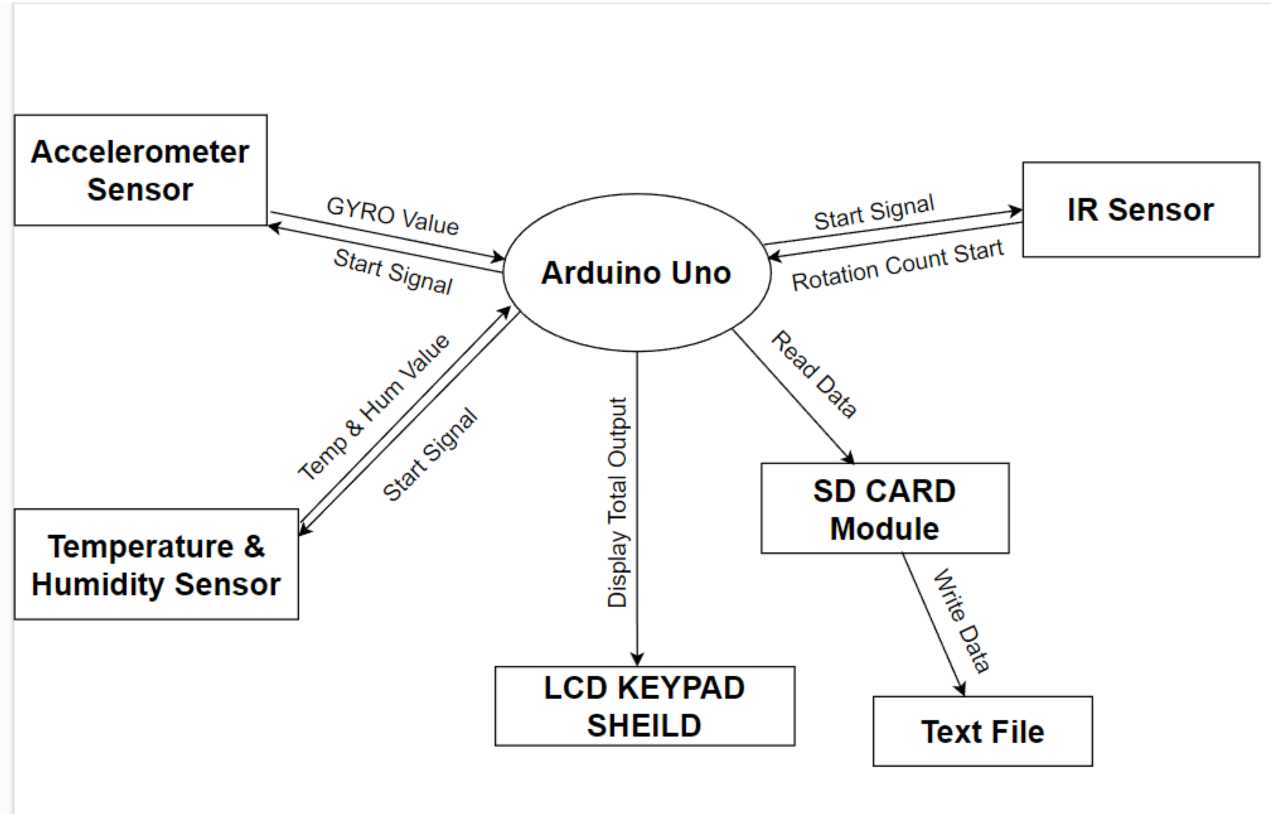

Figure 5.2: System Design

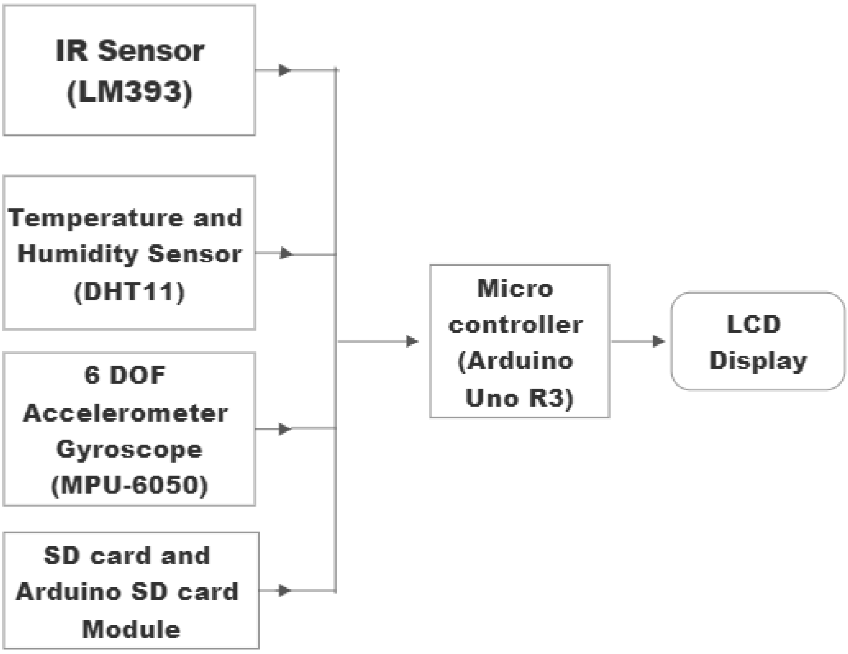

Figure 5.3: Block Diagram

Board	Price (BDT)	Digital I/O Pins	Analog Inputs	Ease of Use	Other Features
Arduino Uno	840 TK	14	6	Beginner friendly, Easiest among the 4	16 MHz clock, power jack, reset button, LED indicator
Arduino Mega	1995 TK	54	16	Slightly more complex	16 MHz clock, power jack, reset button, LED indicator
Arduino Nano	565 TK	14	8	Similar to Arduino Uno	Small size, breadboard-friendly, USB connector
Raspberry Pi	15499 TK	26	none	More complex than the Arduino boards due to its more powerful processor and additional features	1.4 GHz clock, quad-core CPU, GPIO pins, camera connector, microSD card slot

Figure 5.4: Comparison between different boards

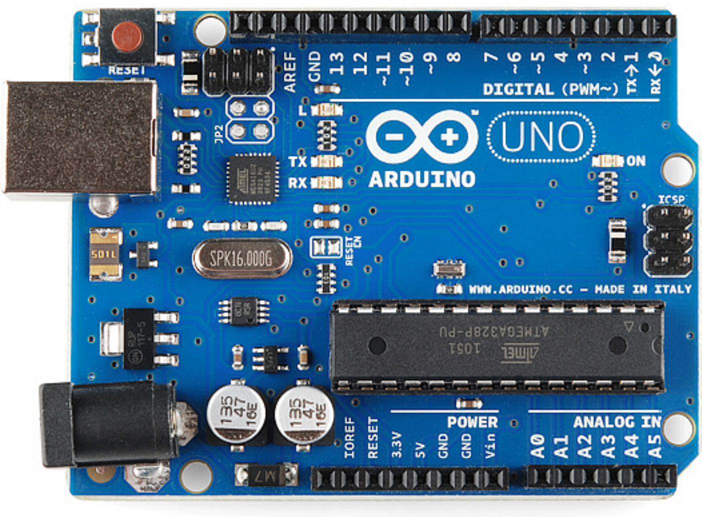

Figure 5.5: Arduino UNO R3

Figure 5.6: IR Obstacle Sensor

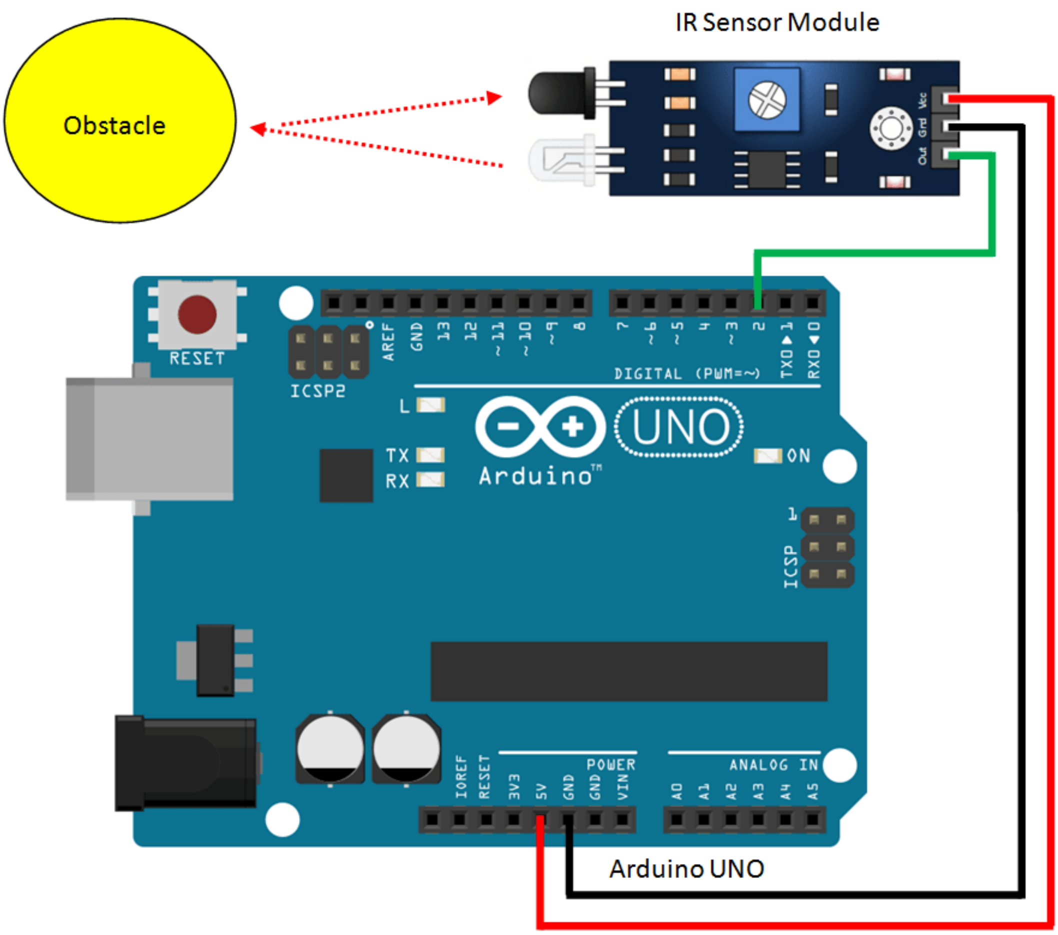

Figure 5.7: Interfacing IR sensor with Arduino

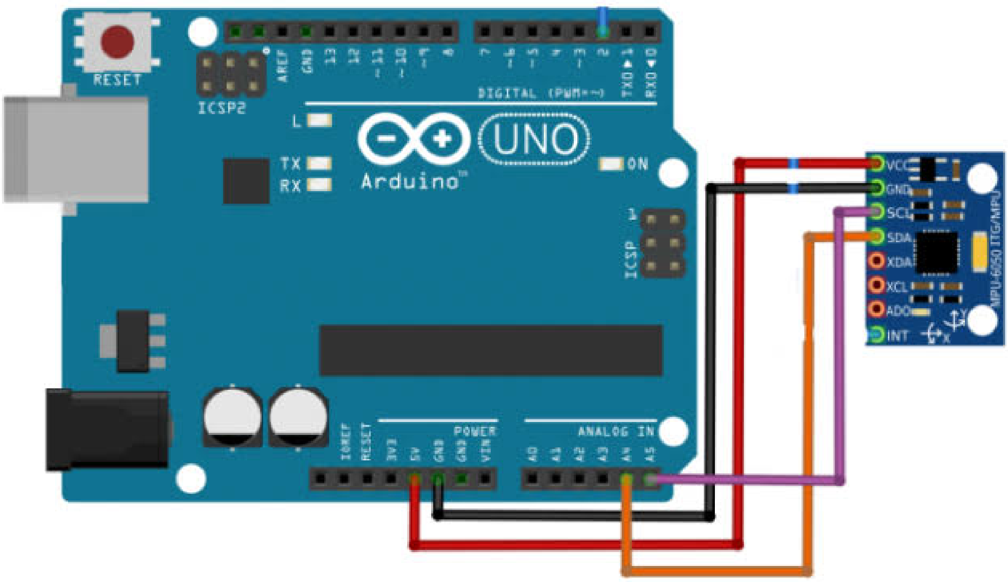

Figure 5.8: 6 DOF Accelerometer Gyroscope MPU-6050 with Arduino

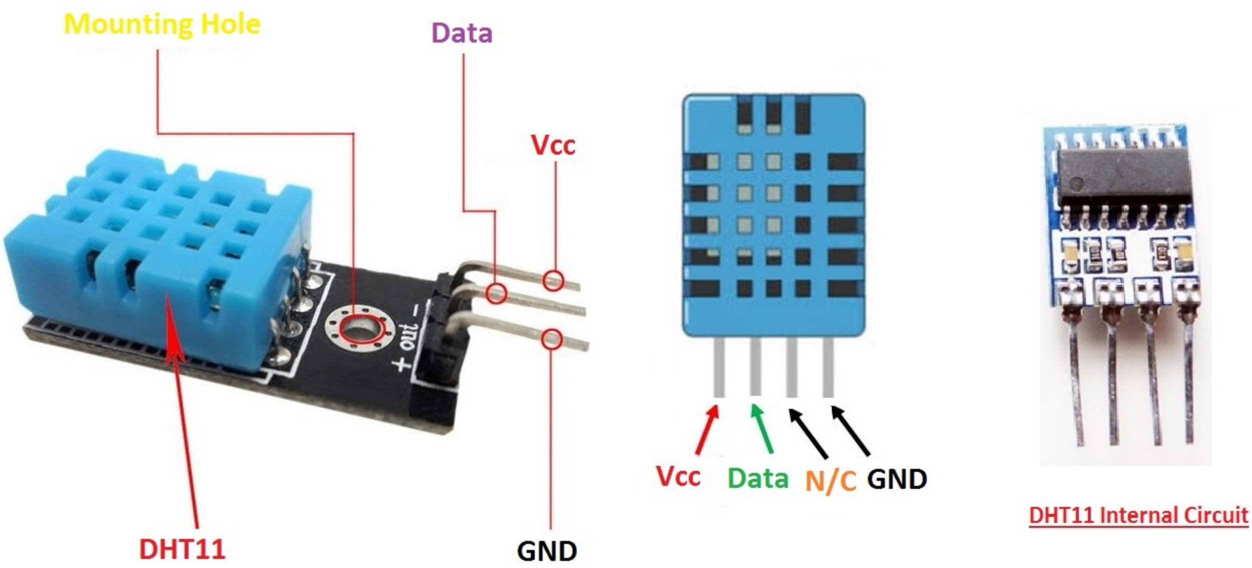

Figure 5.9: DHT11 Sensor

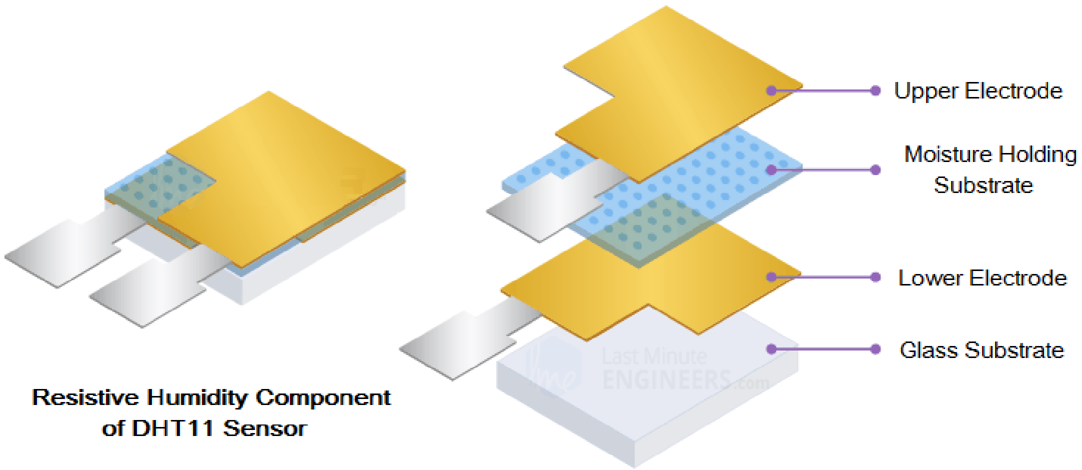

Figure 5.10: Resistive Humidity Component of DHT11 Sensor

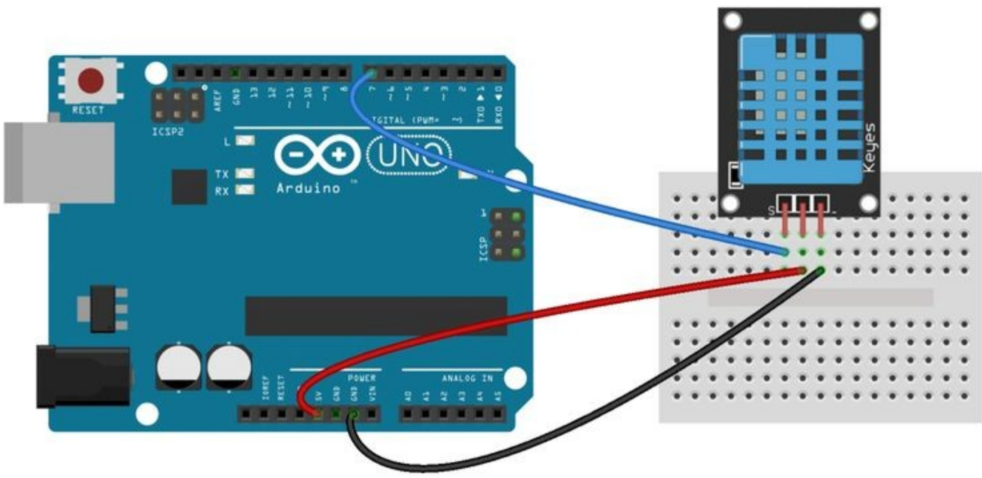

Figure 5.11: DHT11 with Arduino

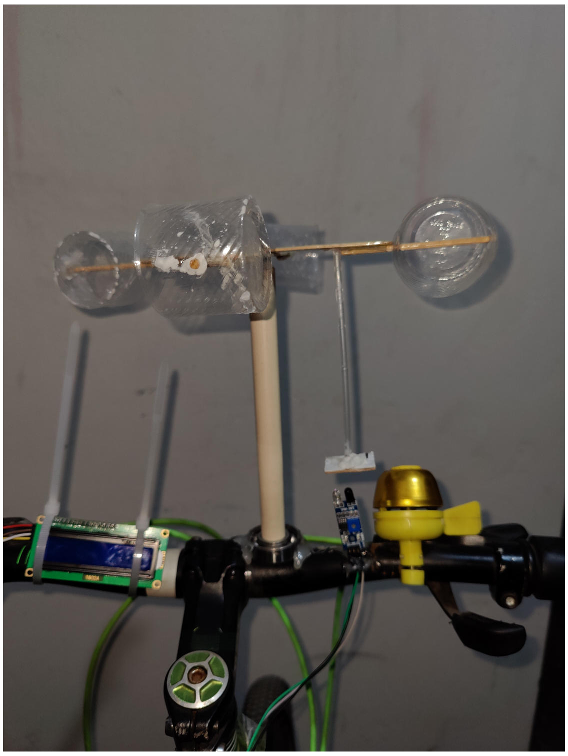

Figure 5.12: Phase 1- DIY Anemometer

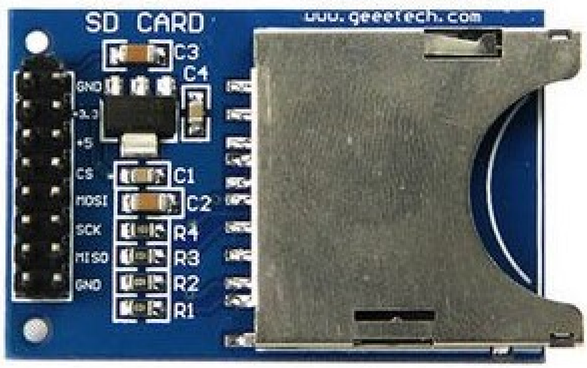

Figure 5.13: SD Card Module

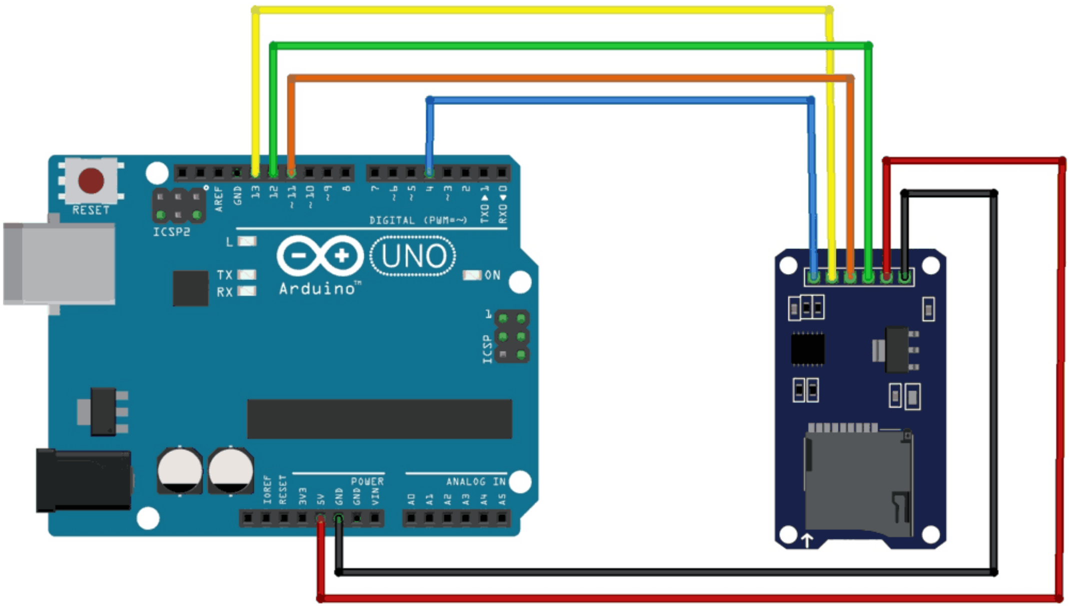

Figure 5.14: SD Card Interfacing with Arduino Uno

Chapter 6

Project Working Procedure

This project is about developing a system that will fairly measure the rickshaw fare by considering various factors like distance, road conditions, the rotation of the wheel, temperature, humidity etc. We need various types of sensors to develop this prototype. First of all, our project is a microcontroller-based system. So, we used the Arduino Uno to connect every sensor to it through a breadboard. Then we used an IR sensor that would detect the wheel rotation to measure the distance passed by the rickshaw. We mounted the IR sensor around the wheel of the vehicle, and it is measuring the whole distance by detecting a small piece of obstacle every time the wheel is rotating. The fare will increase according to the distance. Then there is a sensor named DHT11 to measure the humidity and temperature of the location. It is basically a capacitive humidity sensor that measures the surrounding air and provides a digital signal as an output. Then, we used a gyroscope, which is mainly for estimating the steepness of the road as it can measure orientation and angular velocity. Additionally, we stored these sensors reading in an 8gb sd card which was formatted to FAT32. To implement this, we interfaced a micro sd card module with the arduino uno where the sd card was inserted. This module was reading all the log data provided by those sensors and writing those readings in a text file stored in the memory card. Finally, to display a value from calculating these three factors, we used an LCD Keypad Shield. We plugged this keypad shield onto the Arduino board and then connected all the sensors to it.
Then in the beginning of the ride when the rickshaw puller will press the start button, Elapsed time and covered distance will be displayed in the LCD keypad shield. Then when the stop button is pressed, Ride number and total fare will be shown in the lcd keypad shield.
After setting up the whole prototype, we made a chassis box and put all components in it. This chassis will be mounted in the vehicle. After that, we installed some libraries to run and compile the code. To run the code for measuring temperature and humidity, we installed a library named DHT.h. The MPU6050_tockn.h library was installed to compile the code of the gyroscope to measure the steepness of the road. To communicate with the I2C device we installed a library called Wire.h. In addition to that, we will not get any value displayed on the LCD if we don't install the Liquid Crystal library. So we imported this library as well. Lastly, to communicate with the sd card via the SPI interface, we used two libraries named SD.h and SPI.h. Finally, We connected our Arduino to a PC with a USB cable to supply power. However, when we mounted it on a vehicle in real life, we used a power

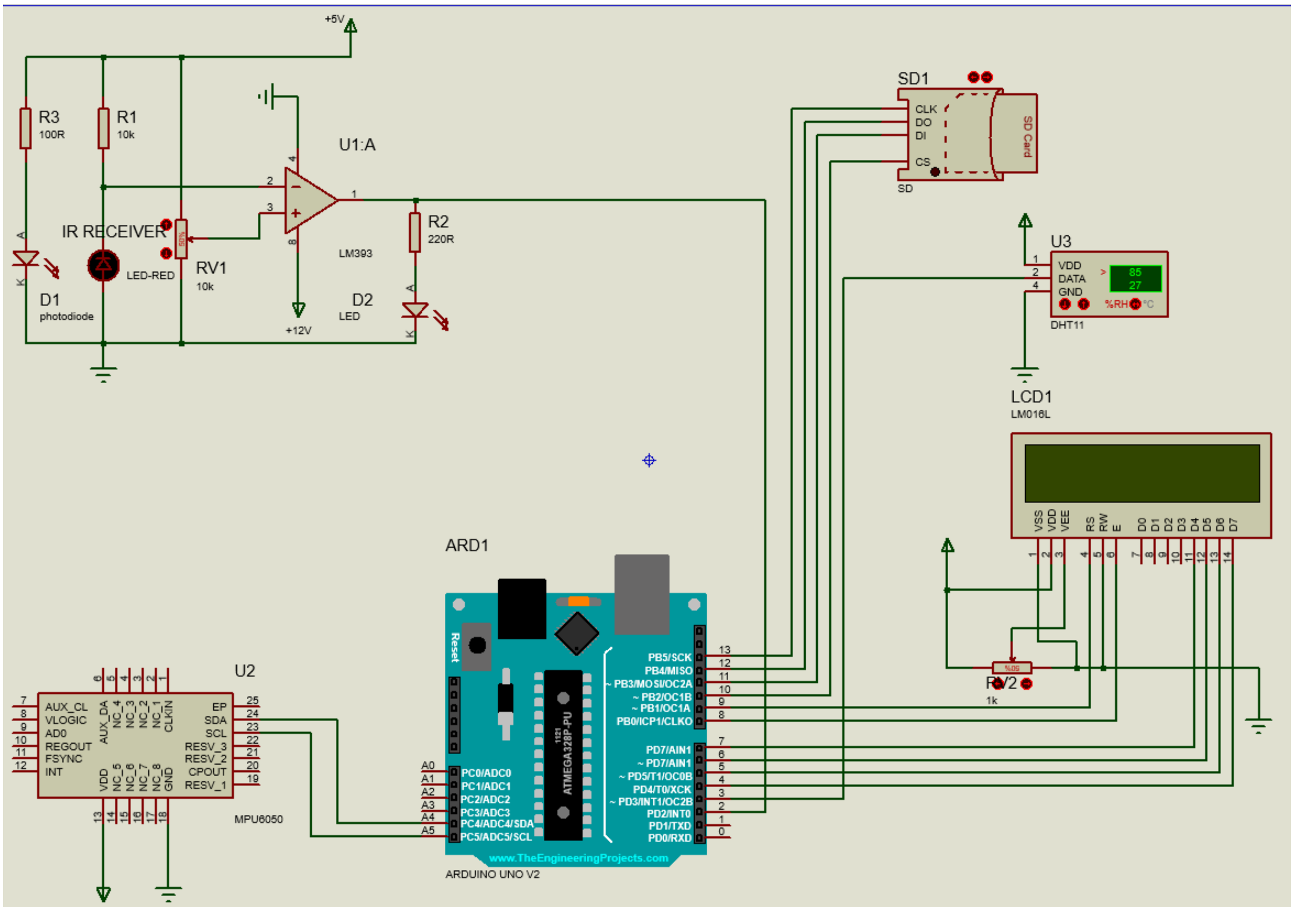

Figure 6.1: Schematic Diagram

bank to supply the power. Each sensor was checked separately before merging them together to get the actual fare. So this was our working procedure for the prototype.

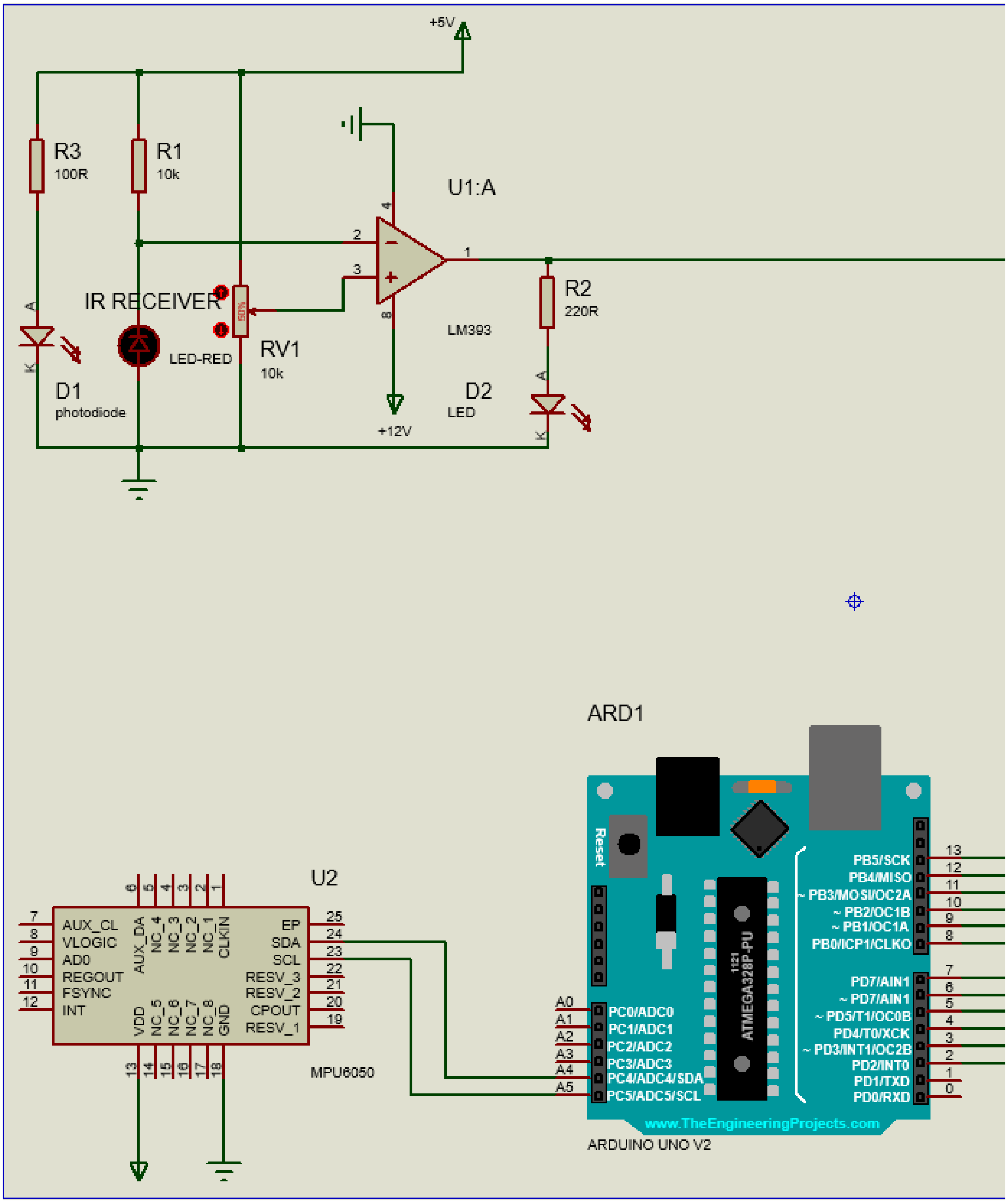

Figure 6.2: MPU6050 Schematic Diagram

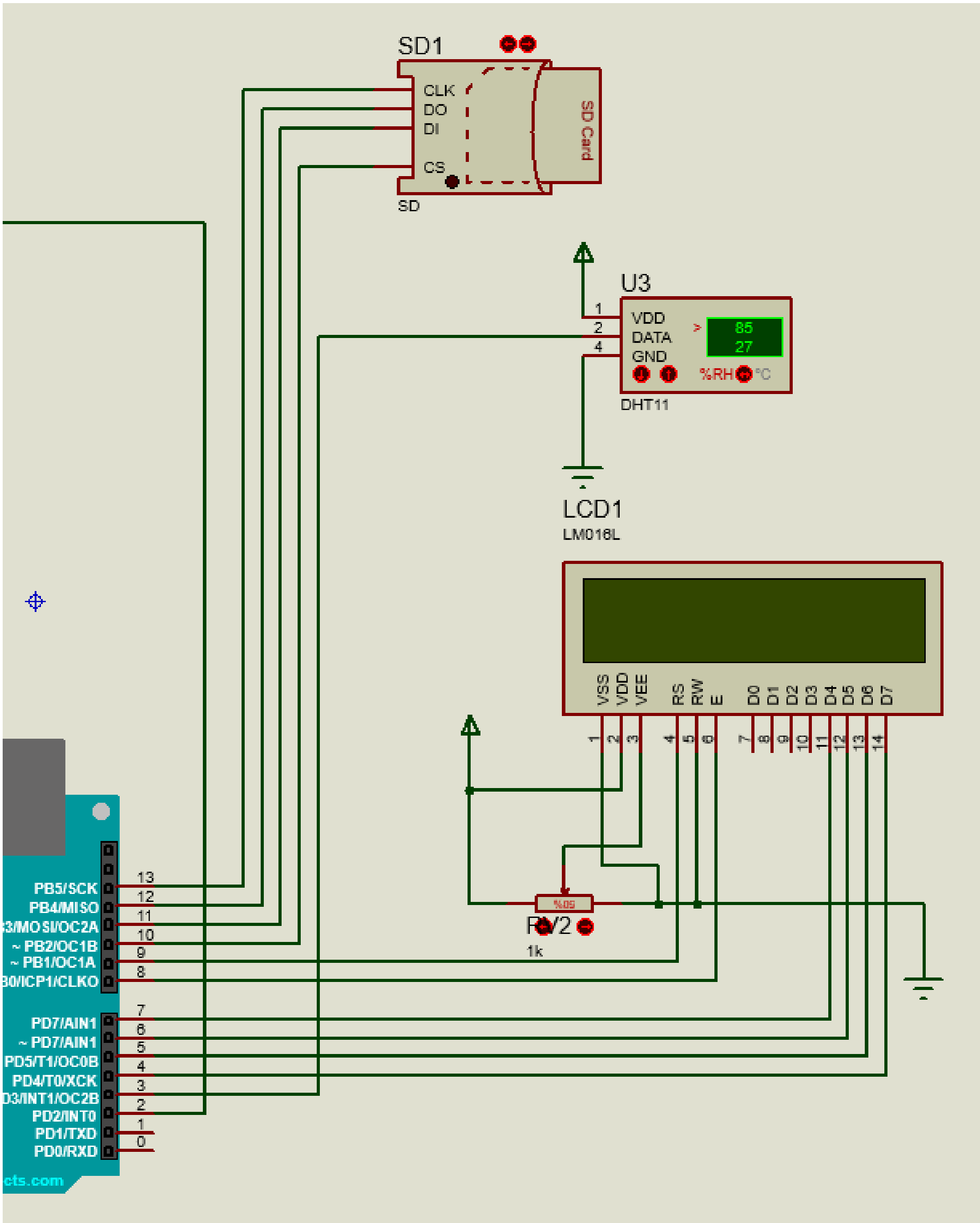

Figure 6.3: Schematic Diagram

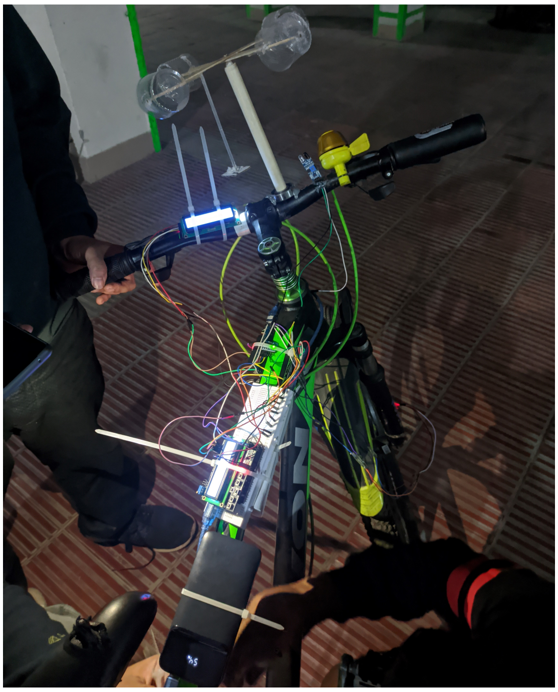

Figure 6.4: Initial Phase - Initial project development

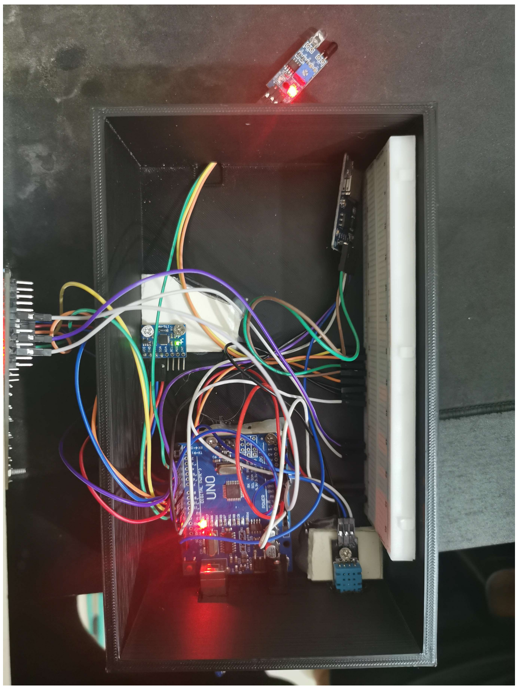

Figure 6.5: Final Phase - Securing the System in a Chassis Cover

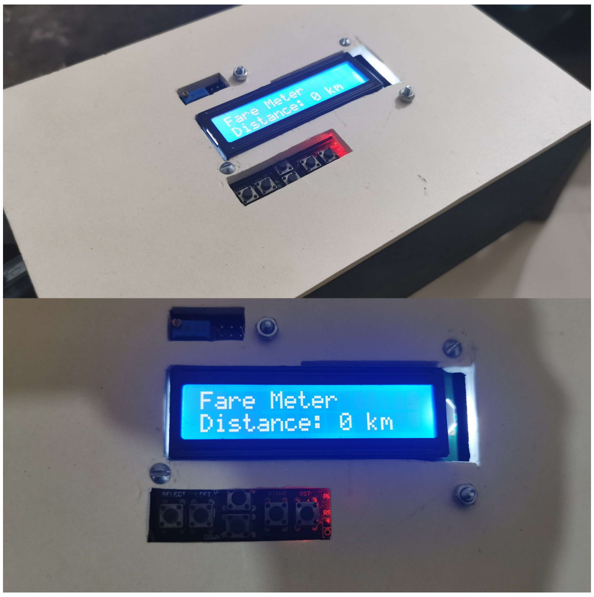

Figure 6.6: System in a Chassis ready to be mounted on Rickshaw

Figure 6.7: System in a Chassis mounted on a cycle

Chapter 7

Result and Discussion

7.1 Dataset

Distance (km)	Distance (m)	Fare				rotation
		dis	temp	gyro	total (BDT)	
1.5	1500	29.82	1.23	0.7	31.75	710
0.7	700	13.902	1.28	0.056	15.238	331
0.5	500	9.912	1.07	0.018	11	236
0.6	600	11.928	1.33	0.02	13.278	284
1.2	1200	23.856	1.07	0.4	25.326	568
0.35	350	6.93	1.07	0.011	8.011	165
0.25	250	4.956	1.42	0.01	6.386	118
0.5	500	9.912	1.33	1.2	12.442	236
0.8	800	15.918	1.07	0.26	17.248	379
0.9	900	17.892	1.07	0.7	19.662	426
1.1	1100	21.882	1.07	1.1	24.052	521
1.3	1300	25.872	1.23	1	28.102	616
1.7	1700	33.81	1.28	0.6	35.69	805
0.96	960	19.068	1.23	0.9	21.198	454
1.2	1200	23.856	1.07	0.8	25.726	568
1.4	1400	27.846	1.07	1.4	30.316	663
0.86	860	17.094	1.23	1.2	19.524	407
0.67	670	13.314	1.28	0.07	14.664	317
0.55	550	10.92	1.42	0.04	12.38	260
0.85	850	16.884	1.23	0.3	18.414	402
1.32	1320	26.25	1.07	0.6	27.92	625
1.11	1110	22.092	1.07	0.23	23.392	526
0.88	880	17.514	1.42	0.3	19.234	417
1.9	1900	37.8	1.28	1.3	40.38	900
1.86	1860	37.002	1.33	1.2	39.532	881
0.95	950	18.9	1.42	0.8	21.12	450
2	2000	39.774	1.07	1.1	41.944	947
2.1	2100	41.79	1.23	1.3	44.32	995

Figure 7.1: Test run Output

7.2 Discussion

The test run produced satisfactory results. We took readings in different weather conditions as well as on different areas, including Gulshan, Banani, Dhanmondi, Hatirjheel, and Mohakhali. We spoke with rickshaw drivers and passengers about the cost of a ride in Gulshan 2. Both parties agreed that the outcome was satisfactory. For the distance cost, we have given the unit price as 0.042 TK per rotation, which covers 2.11 m. The unit price for temperature is 0.02 TK and for humidity, it is 0.01 TK. For steepness, the unit price is 0.007 TK. The table clearly shows that as the distance traveled increases, so does the fare charged by the transportation service. This is consistent with standard pricing models for transportation services. The table also includes data from various sensors, such as temperature and gyroscope readings. Then a price for these variables gets added to the distance fare.

7.3 Limitation

- **Anemometer:** We faced a difficult task in measuring wind velocity. Despite the availability of various anemometers, they tend to be costly and only able to measure wind speed and direction from a stationary position. Thus, it would not make sense to purchase them, as we need to measure wind speed for a moving object or vehicle. This presents a genuine challenge when measuring wind speed under such circumstances. As the cost of an anemometer was prohibitively expensive, we attempted to create a DIY version using easily obtainable components. Unfortunately, we encountered inconsistent and unusable results and found that a second IR sensor was necessary to track the number of rotations accurately. While mass production could bring down the costs, identifying a cheaper alternative to the anemometer remains a crucial goal for future studies. One possible approach would be to explore alternative methods for determining wind speed, which could prove invaluable to our project. In addition, we have conducted research into potential techniques for measuring both wind direction and speed.

7.4 Future Work

- **GPS:** GPS can be used on this prototype in the future. Though it can be a little bit expensive to add a GPS in the system right now, however in the future when there will be cheaper alternatives, we can certainly add GPS to locate the vehicle and track the distance.

- **Anemometer:** One possible solution we considered involves using a potentiometer to measure rotation and a wind vane to determine wind direction. However, we encountered a challenge as the wind vane would be mounted on a moving rickshaw and could move even in the absence of wind, making it difficult to distinguish between static and moving situations. Another approach we explored was to remotely obtain wind speed information from an online server, but this would require a significant overhaul of the project.

7.5 Observation and Challenges

As mentioned earlier, this is a domestic issue and no work has been done on it yet. Therefore, we could not get much benefit regarding the project from previous studies. Moreover, as it is a hardware dependant project we faced lots of issues in handling with various components. Repeated hardware malfunction and damage increased the complexity of our work and consumed a lot of time. Moreover, the components are not easily available in the market which created some difficulties.

7.6 Conclusion

To conclude, as this is a domestic issue rather than an international one, we did not find much work on this topic. And the ones we came across didn't offer a satisfying solution to our problem. Some of the proposed solutions failed to account for crucial external elements such as weather, route steepness, and passenger numbers. We need to refine some of these solutions and add some new ones to solve a fraction of this problem. If we can implement our plan for a system for fair rickshaw fare, it'll eradicate a huge hassle from our daily life. Both drivers and passengers will benefit from it. Along with this, we can also look into the statistics of this massive sector of human labor. We couldn't find any research on the subject either. In this manner, our research paper will be of great assistance to society.

Bibliography

[1] Md. Karim and K. Salam, "Organizing the Informal Economy Workers: A Study of Rickshaw Pullers in Dhaka City," March 2019.

http://bilsbd.org/wp-content/uploads/2019/06/A-Study-of-Rickshaw-Pullers-in-Dhaka-City.pdf

[2] S. Akter, T. T. Jui, T. Athaya, A. Zaman and S. Rafi "A Proposed System for Fare Measurement of Bangladesh" in Conf. 5th International Conference on Advances in Electrical Engineering (ICAEE), Dhaka, Bangladesh, 26-28 September, 2019
shorturl.at/aEZ56

[3] D. Pomerantz, "Taxi trip meter system with indication of fare and distance violations," US Patent 6, 122, 591, 2000.

[4] S. Alok, Md. Haque, R. H. Junayed and R. Tasnim, "Smart Fare System for CNG," BSc dissertation, Dept. CSE, Brac University, Dhaka, 2018.

http://dspace.bracu.ac.bd/xmlui/handle/10361/10191

[5] M. of Communications. (2000). Geometric Design Standards for Roads High-ways Department. Road Geometric Design Manual.pdf. [Online] Available:

http://www.rhd.gov.bd/Documents/ConvDocs/Road%20Geometric

[6] W. contributors. (2022, May) Relative Humidity. [Online] Available: Humidity - Wikipedia

[7] Climatemaps. (2009-2017). Relative Humidity in Dhaka, Bangladesh. [Online] Available:

https://www.dhaka.climatemps.com/humidity.php

[8] Weather Climate. (2022). Average Humdity in Dhaka [Online]. Available:

https://weather-and-climate.com/average-monthly-Humidity-perc,Dhaka,Bangladesh

[9] HowtoElectronics. (2021, Aug 3) How to Measure Wind Speed using Anemometer Arduino [Online] Available: How to Measure Wind Speed using Anemometer Arduino (how2electronics.com)

[10] W. contributors. (2022, May 6) Microcontroller. [Online] Available: https://en.wikipedia.org/wiki/Microcontroller

[11] "UNO R3," Arduino.cc. [Online]. Available: https://docs.arduino.cc/hardware/uno-rev3

[12] W.Electronics. (2022, Sep 18) IR Sensor : Circuit, Types, Working Principle its applications. [Online] Available: https://www.watelectronics.com/ir-sensor/

[13] W. contributors. (2022, Sep 12) Gyroscope. [Online] Available: https://en.wikipedia.org/w/index.php?title=Gyroscopeoldid=1109958476.

[14] Components101. (2022, Sep 18) DHT11–temperature and humidity sensor. [Online] Available: https://components101.com/sensors/dht11-temperature-sensor

[15] N. Agnihotri, "Arduino compatible coding 15: Reading sensor data from DHT-11 without using a library," Engineers Garage, 6 March, 2022

https://www.engineersgarage.com/articles-arduino-dht11-humidity-temperature-sensor-interfacing/

[16] Katranji.com. [Online] Available: https://www.katranji.com/tocimages/files/464885-312477.pdf. [Accessed: 18-Sep-2022].

[17] Circuitjournal.com. (2022) 50kg load cells with HX711 and arduino. 4x, 2x, 1x diagrams [Online]. Available: https://circuitjournal.com/50kg-load-cells-with-HX711. [Accessed: 18-Sep-2022]

[18] Energiazero.org. HX711 Load Cell Amplifier Interface with Arduino. [Online] Available: shorturl.at/afgu1

[19] RoboticsBD. (2022, Sep 18) 16x2 serial LCD module display for arduino assembled. [Online] Available: https://store.roboticsbd.com/arduino-shield-module/1349-16x2-serial-lcd-module-display-for-arduino-assembled-robotics-bangladesh.html

[20] A.ProjectHub. (2022, Sep 18) Interface I2C 16x2 LCD with Arduino Uno (Just 4 wires). [Online] Available: https://create.arduino.cc/projecthub/akshayjoseph666/interface-i2c-16x2-lcd-with-arduino-uno-just-4-wires-273b24. [Accessed: 18-Sep-2022].