

GOLD PRICE ALERT

Minor Project Report

**Submitted in partial fulfillment of the requirements
of the degree of**

Bachelor of Technology

By:

19U02005	Mayank Jain
19U02006	Kartikey Singh
19U02048	Shivam Bist
19U02070	Shikhar Verma

**Under the Supervision of:
Dr. Shalini Stalin**



**Department of Computer Science and Engineering
Indian Institute of Information Technology Bhopal
April 2022**

CONTENTS

<u>SNO.</u>	<u>TITLE</u>	<u>PAGE NO.</u>
1.	Introduction of the project	3-4
2.	Background Details and Literature	5-8
3.	Methodology and Frameworks	9-11
4.	Working Process and Algorithm	12-15
5.	Implementation and Result Analysis	16-20
6.	Conclusion and Future Scope	21
7.	References	22

INTRODUCTION OF THE PROJECT

We have acquired decent knowledge about the financial market and have been studying different asset classes and their behaviour in different financial conditions. It is difficult to find an asset class that has greater polarization than gold. There are people who either love it or people who hate it, and more often than not, they remain in the same camp forever. Since gold has very few fundamentals of its own (again a source of high polarization), we will try to find gold price considering the current market scenario using web-scraping techniques along with notification and API tools namely “*BeautifulSoup*”, “*TwilioAPI*”, and “*win10toast*”.

It will be a helpful application as it will give us an alert about the latest prices of gold in the stock market and also, it will alert us whether it is a good time to invest the money or sell the pile of stocks that we already have. As, in India, many people lose their money due to a lack of knowledge about financial market situations i.e., what factors decide the prices of gold, and also due to lack of time and right mentorship. Time is the most precious asset and not only in India but in the whole wide world, many people have less time to give into this, they don't have time to see their children's merit cards so how they will regularly see the prices. So, this will be helpful for these people who have less time or knowledge by sending SMS and/or WhatsApp alert. With this application, people with less knowledge can also earn money without having the headache of constantly checking the value of gold in the market.

We all know how women are fond of gold so this application will most definitely attract the other 50% of the population and keep them engaged and up-to-date with the latest market price of gold, and will of course alert them if prices go down, they can even customize the application to send alerts if and only if prices go down, and if it does they'll be the first one to get the information and purchase accordingly without being late because if the buyer isn't up to the date the demand in the market will rise and we all know when demand increases, to maximize the profits and exploit the demand, the product's price starts increasing in our case it is gold.

In order for our application to run without errors, we must install a few dependencies and libraries which are crucial for the application. The libraries are as follows:

- **BeautifulSoup4**
- **Requests**
- **Winsound**
- **Win10toast**
- **Twilio API**

All the mentioned libraries can be installed using the “*pip install*” command in the terminal.

Coming to the actual implementation of our program we have taken into consideration of error handling if the user enters an invalid city name or website servers are offline or log file being currently accessed or any other kind of issue prone circumstances, we have added try-except blocks so that application would not crash and keep on running, the next big thing we've considered is picking out the most up-to-date website for our scraping of data on gold rates that is (www.fresherslive.com) it also has good uptime, out of several hours of testing we haven't found a single case of the situation when the website was down or unresponsive. The last great consideration we did was giving the ability to the user to change the city accordingly throughout the whole country as it will give an idea which place to sell at a bigger profit and where to buy at low cost. Not to forget that all the information that is retrieved or scraped and displayed on the terminal of the screen is stored in the data.csv file so that users can keep and maintain track of past records and make their decision accordingly.

We've divided our application into various segments for processing the scraped data from a website. Each segment has its own error and exception handling so that if a particular segment fails the program doesn't crash but skips the faulty segment. The segment parts are as follows:

- Extracting city name from city.txt external file
- Scrape data from the site and parses the HTML
- Print the rate on the terminal
- Save the records in the external CSV log file
- Initiate alert system
 - If the change is positive, it'll play 2 beep sounds with 1000Hz frequency for 350ms and send desktop notification about the increase in price along with WhatsApp/SMS alert.
 - If the change is negative, it'll play 1 beep sound with 2000Hz frequency for 1000ms and sends desktop notification about the drop in price along with WhatsApp/SMS alert.
- Loop indefinitely for a given number of seconds

The program will work without issues with only the python (.py) but it's highly recommended to use the gold.ico file along with the city.txt (can be later created by the user) for a better user experience because the gold.ico is an icon file that is used in desktop notification when alerting the user on the desktop if it doesn't exist the default python logo will show up and it won't give the aesthetics of gold investment. Also, the "**city.txt**" file gives the user the liberty of changing the required city to check the gold price, if the file does not exist then by default New Delhi will be selected as it's the capital of our country. So, for a user to have more control and an appealing interface it's highly encouraged to use the additional files along with the main file.

BACKGROUND DETAILS AND LITERATURE

As the name of our project suggests, it is a program that will continuously keep the real-time track of entities like the price of gold, silver, and also the status of the stocks. This program will notify the user automatically regarding the status through SMS and/or WhatsApp messages which will help the user in decision making whether to buy the entities or not, whether to sell or buy stocks. This program is having the functionality of notifying through the beep sound corresponding to the fluctuation in various entities.

At the current time, people are facing a lot of problems financially which can go on to increase in the future as the COVID-19 is in its full swing and there has also been ambience of inflation in the country so it is a perfect time for people who are having a steady stream of income to invest in entities like gold, silver, and the stock market and following are some of the reasons why people should invest: -

- **Low financial activity:** With the lockdown in full swing, most businesses may have to hit the pause button. This will bring down the economic activity of the country, driving the markets down. While it may sound like bad news, a dip in the market means a fall in prices. When we invest when the market is down, there are more chances of profitability in the near future.
- **More time to perform research:** Being at home may not leave us with many options to keep ourselves occupied. With much less travel time, we have more free time than we ever had before. We must take advantage of it to do a thorough analysis of the assets and commodities we wish to invest in. Thorough research equips us with knowledge of the market which assists us to make better investment decisions.

Why this application could be ground-breaking at the time of COVID-19 and hike in prices of various commodities?

Investing at this time will help people in their future, following are the reasons that will justify the importance of this program mainly in the case of investing in gold.

1. Best hedge to counter high inflation in economy

The commodity Gold is one the best indirect solution to the problems like currency debasement and inflation. When the equity markets crashed last year, gold had a bull run for several months- a trend that experts predict will continue in the current state of economic turmoil. The performance of gold depends on several factors like inflation, global trade of the metal, and other geopolitical issues. During these kinds of unpredictable circumstances, it has been recorded that gold prices have remained stable outperforming other asset classes and at the same time, absorbing economic shocks better. Inflation

brings about a significant rise in the price of gold as it erodes the value of money by lowering the purchasing power of the Dollar. Thus, gold becomes a hedge against inflation as, over a longer period, it is seen to offer higher-than-inflation returns. With the Rupee underperforming, the economy is experiencing currency debasement- a time when gold can act as an attractive investment tool.

This program is equipped with information regarding a lot of entities in one place which makes this program unique from other pre-existing programs and the thing which could be the USP of this program is that the user doesn't have to frequently manually go to the interface of the program to check the current status of the entities but he has to just set the program once to keep him updated regarding the status of the entity.

And since in this fast-moving world most of the people who are could be our potential users are the people who are busy in doing their jobs to earn money so it will be very convenient for them to just set their need only once and get the update automatically and frequently without constantly worrying about their investment.

2. Extremely low rate of interest

The Covid-19 pandemic had and will have a massive negative impact on the world economy. And this thing is inevitable. Amid inflation and continued uncertainty regarding the virus and the future of the world, Central banks all around the world have been cutting interest rates. In India as well, most of the monetary policies in 2021-22 are targeted to evolve macroeconomic conditions so that the economy can attain sustainable growth and inflation remains within an acceptable range. With the policy rate cuts from RBI, investors are moving from fixed income to gold in a low-rate environment. With repo rates unchanged as of October 2021 to boost the economic conditions, the yellow shining metal has become a sought-after investment tool and its price has shot up significantly high.

3. Lucrative source for wealth creation considering the global pandemic

Gold has emerged as a major asset class during the current crisis. Its performance was fairly stable during the two waves of the pandemic, which made a lot of new investors hop onto the gold bandwagon. Considering things are still very much uncertain with regard to the coronavirus, and the global economy is still healing, investing in gold is a safe path towards wealth creation. What makes gold investment more lucrative is the transparency that comes with it. Unlike other financial investments like real estate, gold is easy to buy and sell and does not involve any tedious processes. Further, in Indian households, physical gold is also a symbol of family wealth and prosperity and is passed down from one generation to

another. Thus, the ease of investment is another major reason to include gold in one's investment portfolio.

Existing applications

There's not only one solution for one problem, similarly, there already exists a lot of cross-platform stock alert applications made to resolve the problem of keeping track of stocks and/or various other commodities and getting alerts from them for better opportunity to get profit from the investment done so far, for example:

- Stock Alarm
- MoneyPatrol
- Morningstar
- Real-Time Stock

Existing applications limitations

Although the existing applications provide a tremendous amount of support and features there still exist various limitations that can be improved and worked upon. We've already discussed various existing applications and it's time to discuss their limitations and what we have to offer so that it overcomes the existing limitations.

- **Non-domestic Platform:** Existing applications only provide coverage and investment opportunity that is internationally recognized, say we want to invest in local emerging companies of India and keep an eye on them, we won't be able to do that through these applications.
- **Limited to Market Servers:** The fetching of data is done through real-time servers of the commodity market and is hardcoded but our application is based on web-scraping and can fetch information from any website we put in.
- **Wide Range Alerts:** Although they provide a good range of alert providing techniques from e-mails to SMS our application is the best in this case scenario. Our program can send alerts as Desktop notifications, WhatsApp messages, Beep sounds on PC and SMS. So, we just have to sit back and relax.

Challenges faced in making this application

The biggest challenge we faced while making this application was the frequent changing of the website where we were fetching our data from. The sequence of data we gathered depends on the

layout (HTML) of the website if the website were to change its layout, the part of code has to change too in order to accommodate the website changes, the second major issue was API we are using free Twilio API that has a limit on the amount of SMS or WhatsApp messages we can send, so debugging was quite challenging on alerts because we didn't want to run out of credits and exhaust our fixed limited quota on messages. The third and last issue was the lack of testers, we have a small team of 4 people and cannot do exhaustive testing of the application as that would require powerful machines to run big test cases and a lot of man-force to cover the edge cases.

What does this program bring to the table?

This program is equipped with information regarding a lot of entities in one place which makes this program unique from other pre-existing programs and the thing which is the USP of this program is that the user doesn't have to frequently and manually go to the interface of the program to check the current status of the entities but he has to just set the program once to keep him/her updated regarding the status of the entity.

And since in this fast-moving world most of the people who are could be our potential users are the people who are busy in doing their jobs to earn money so it will be very convenient for them to just set their need only once and get the update alerts automatically and frequently.

METHODOLOGY AND FRAMEWORK

Project management methodology offers a clear project roadmap that lists all the steps required to deliver a project successfully. Essentially, a methodology is a collection of **methods, practices, processes, techniques, procedures, and rules**. In project management, methodologies are specific, strict, and usually contain a series of steps and activities for each phase of the project's life cycle.

Now coming to our project, our project is based on checking the latest gold prices and making people alert about it. The main **motive** of choosing this project is to provide better alternative to people who want to get latest updates on gold price. This tool will provide them quick and accurate results, which will save time and efforts of many people.

This program will notify the user automatically regarding the status through SMS and/or WhatsApp messages which will help the user in decision making whether to buy the entities or not, whether to sell or buy stocks.

Tools-

As had used python for this project so, we will also require a suitable environment for our code to run. For that we had used different tools for smooth functioning of our application -:

- *BeautifulSoup*
- *Requests*
- *Winsound*
- *Win10toast*
- *Twilio API*
- *csv*

We had used BeautifulSoup for website scraping, Twilio API for sending SMS or WhatsApp messages to users, so that they can always be up-to-date. Request tool is used to send request to the website while *winsound* is for generating beep sound. We had also used csv files to store the downloaded data.

The first thing that we had done, is to import the necessary dependencies, that we will use in the upcoming part of the program.

Importing Data-

We had used the dataset available on (<https://www.fresherslive.com/gold-rate-today>) website and scraped it using BeautifulSoup module. The site has good uptime, and we haven't found any case of website going down.

```
data = request.get("https://www.fresherslive.com/gold-rate-today/"+city)
```

As the dataset file is downloaded, it gets stored in the form of a CSV file, to read it, we will be needing a method `read.csv()` to read CSV files. We also take data/input from user to increase our accuracy and efficiency; for example, we had given user the ability to change the city accordingly throughout the whole country as it will give an idea which place to sell at a bigger profit and where to buy at low cost.

The scraped data then goes to different segments for processing. Each segment has its own error and exception handling, which prevents the program from getting crash by skipping the faulty segment.

Segments-

1. Firstly, we had extracted city name from **city.txt** external file.
2. We had Scraped data from the site using BeautifulSoup module and parses the HTML to a python list.
3. Next, we had Printed the rate on the terminal.
4. We had used 'alert system' **technique** in which, the application makes beep sounds to alert the users according to the recent change in gold price. For example-:
 - ❖ If the change is positive, it'll play 2 beep sounds for 350ms and also.
 - ❖ If the change is negative, it'll play 1 beep sound for 1000ms.

It also sends desktop notification along with WhatsApp/SMS, about drop/rise in gold values.

The main USP of this program is that the user doesn't have to frequently and manually go to the interface of the program to check the current status of the entities but he has to just set the program once to keep him/her updated regarding the status of the entity.

Frameworks:

A framework, or software framework, is a platform that provides a foundation for developing software applications. Think of it as a template of a working program that can be selectively modified by adding code.

A framework can include support programs, compilers, code libraries, toolsets, and APIs to develop software and create systems. Open-source frameworks are always being updated and improved.

The purpose of a framework is to assist in development, providing standard, low-level functionality so that developers can focus efforts on the elements that make the project unique.

BeautifulSoup – Web Scrapping Framework

Web Scrapping is the process of accessing HTML webpage from websites and extracting valuable information from that data.

To install BeautifulSoup package in windows: `pip install bs4`

It takes the help of the most popular library named *Requests* to make a request to a particular server.

CSV Module

.csv format is the most common import and export format for spreadsheet and databases. The csv module implements classes to read and write tabular data in CSV format. It allows programmers to say, “write this data in the format preferred by Excel,” or “read data from this file which was generated by Excel,” without knowing the precise details of the CSV format used by Excel.

Request Module

The requests module allows you to send HTTP requests using Python. The HTTP request returns a *Response Object* with all the response data (content, encoding, status, etc).

Method in request module: delete(), get(), head(), patch(), post(), put(), request().

Winsound Module

The winsound module provides access to the basic sound-playing machinery provided by Windows platforms. It includes functions and several constants.

Methods in winsound module: Beep(), PlaySound(), MessageBeep(), etc.

Win10toast Module

Win10toast is used to create desktop notification. It is an easy way to get notified when some event occurs.

Some methods are ToastNotifier(), show_toast().

DateTime Module

datetime can be imported to work with the date as well as time. It supplies classes to work with date and time.

DateTime module is categorized into 6 main classes – date, time, datetime, timedelta, tzinfo, timezone.

Twilio's APIs

Twilio's APIs (Application Programming Interfaces) power its platform for communications. Twilio provides many separate REST APIs (Representational State Transfer) for sending text messages, making phone calls, looking up phone numbers, managing your accounts, and a whole lot more.

OS.path Module

This module implements some useful functions on pathnames. To read or write files see open(), and for accessing the filesystem see the os module. The path parameters can be passed as either strings, or bytes.

WORKING PROCESS & ALGORITHM

In this section we will go through the whole working process of our application it will be very intuitive and easy to understand that we can consider it to be in layman's terms, after that we will move on to the algorithm section where we will have a pseudo code written and we will be doing algorithmic analysis on it for better understanding of code along with time and space complexities.

Working Process:

1. Importing the frameworks

The first task is to import all the frameworks and tools that we will be using in our application, they're listed as follow:

- BeautifulSoup (*For website Grabbing*)
- csv (*To Externally Store*)
- requests (*For Sending Requests*)
- winsound (*To generate beep sound*)
- win10toast (*For Desktop Notifications*)
- time (*For loop or sleep*)
- os.path (*To Check if file exists or not*)
- twilio.rest (*For Twilio API*)
- datetime (*To extract date and time*)

2. Initialization

Here we will initialize the Twilio API's variables that will be used as a parameter when we will send API to our mobile phones, they're *account_sid*, *auth_token*, *frm*, *to*, for Twilio ID, token, Twilio number and receiver number respectively.

By default if only numbers are provided the SendAPI() function will send SMS to receiver number, in order to send WhatsApp message we have to add prefix "*whatsapp:*" to the Twilio number and receiver number.

3. SendAPI Function

Using the above initializations as parameters and imported client from Twilio API we can create a function that'll take text and input and send us SMS or WhatsApp message, but we have to make sure our *account_sid* is not shared in public because anyone can exploit it to send illegal messages. The function has following structure:

```
def SENDAPI(text):
    client = Client(account_sid, auth_token)
    message = client.messages.create(to=to, from_=frm, body=text)
```

4. Scraping

With the help of BeautifulSoup module (which we had imported earlier) the data will be scraped from website (<https://www.fresherslive.com/gold-rate-today>) and will be parsed to a Python List and then we will extract the information from the list accordingly. If change is negative the Notification is sent via SMS or WhatsApp and Desktop Notification.

Algorithm (Pseudo Code):

Below is the pseudo code of our application that will give a brief idea about the coding structure and will help us ease the understanding of the implementation part in upcoming section of this project. In the end we have done algorithmic analysis to get the idea of resources used by the application with respect to input.

```
import BeautifulSoup, import csv, import requests, import winsound
import win10toast, import time, import datetime import Twilio
```

```
account_sid← "ACCOUNTSID"
auth_token← "AUTHTOKEN"
frm← "TWILIONUMBER "
receive← "RECEIVERNUMBER"
```

```
SendAPI(text):
    client = Client(account_sid, auth_token)
    client.Message(text, from←frm, to← receive)
```

```
while(True):
    if('city.txt' file exists):
        city←city_file.read()
    else:
        city← "delhi"

    print("Looking for gold price in "+city)
```

Try:

```
data ← Scrape("https://www.fresherslive.com/gold-rate-today/"+city)
HTML.parse(data)

price = soup.find_price_in_website
change = soup.find_change_in_website

print("1g 22k Price in " + city+ ": "+price+" change: "+change)

datenow←time("%m-%d-%Y")
timenow←time ("%H:%M:%S")
header ← ['City','Date', 'Time', 'Price', 'Change']
data ← [city, datenow, timenow, price,change]

Open('data.csv') as f:
    writer = csv.writer(f)
if(data.csv does NOT exists):
writeTheDataRow(header)

writeTheDataRow(data)

if(change is NEGATIVE):
winsound.Beep(2000Hz, 1000s)
if(gold.ico exists):
DesktopNotification("Gold Price INCREASED", gold.ico, time←5 seconds)
    else:
DesktopNotification("Gold Price INCREASED", defaultIcon, time←5 seconds)

    try:
SendAPI("Gold Price DROPPED")
    except:
print("API Alert Failed!")

if(change is POSITIVE):
winsound.Beep(1000Hz, 350s)
winsound.Beep(1000Hz, 350s)
```

if(gold.ico exists):

DesktopNotification("Gold Price INCREASED", gold.ico, time=5 seconds)

else:

DesktopNotification("Gold Price INCREASED", defaultIcon, time=5 seconds)

try:

SendAPI("Gold Price INCREASED")

except:

print("API Alert Failed!")

except:

print("City Not Found!")

time.sleep(5)

IMPLEMENTATION AND COMPARATIVE RESULT ANALYSIS

In this section we'll walk through the whole implementation process of our application its actual code is very well written with use of appropriate variable name for better understanding along with multiple comments so that new user can also get the idea what a particular section of code is doing.

Implementation:

```
from bs4 import BeautifulSoup as BS      #For Website Grabbing
import csv                               #To Externally Store
import requests                          #For Sending Requests
import winsound                           #For Beeping Sounds (pip install)
import win10toast                         #For Desktop Notifications
import time                               #For loop or Sleep
import os.path                           #To Check if file exists or not
from twilio.rest import Client            #For Twilio API
from datetime import datetime             #For date and time
```

```
#Twilio API Information
```

```
#To send on whatsapp instead of sms(by default)
```

```
#use "whatsapp:+917985414735" and from should be a sandbox whatsapp
```

```
account_sid = ""
```

```
auth_token = ""
```

```
frm="whatsapp:+1"
```

```
to="whatsapp:+91"
```

```
#API Function
```

```
def SENDAPI(text):
```

```
    client = Client(account_sid, auth_token)
```

```
    message = client.messages.create(to=to, from_=frm,body=text)
```

```
while(1):
```

```
    #City Name from city.txt if fails default is delhi
```

```
    if(os.path.exists('city.txt')):
```



```

city_file=open("city.txt","r")
    city=city_file.readline().lower()
city_file.close()
else:
    city="delhi"

print("Looking for gold price in "+city.capitalize()+"...")

#Try and Except for Error Handling
try:

    #Grabs data from the site and parses the HTML
    data = requests.get("https://www.fresherslive.com/gold-rate-today/"+city.lower())
    soup = BS(data.text, 'html.parser')

    #To locate the data from site
    price = soup.findAll("td",class_="center-text")[1].text
    change = soup.find_all('b')[5].text

    #Statement to print
    final="1g 22k Price in " + city.upper().capitalize()+ ": "+price+" change: "+change
    print(final)

    #For external file
    dataExistance = os.path.exists('data.csv')
    now=datetime.now()
    datenow=now.strftime("%m-%d-%Y")
    timenow=now.strftime("%H:%M:%S")
    header = ['City','Date', 'Time', 'Price', 'Change']
    data = [city.upper().capitalize(), datenow, timenow, "Rs."+price.split('₹')[1], change]

    with open('data.csv', 'a', encoding='UTF8', newline="") as f:
        writer = csv.writer(f)

```

```

    if(dataExistance==False):
        # write the header only if file doesn't exist
writer.writerow(header)

    # write the data
writer.writerow(data)


#For sound effect, desktop notification and WhatsApp/SMS Alert
if(change[0]=="-"):
winsound.Beep(2000, 1000)
    if(os.path.exists('gold.ico')):
        toaster = win10toast.ToastNotifier().show_toast("Gold Price DROPPED!",final , duration=5,
icon_path="gold.ico")
    else:
        toaster = win10toast.ToastNotifier().show_toast("Gold Price DROPPED!",final , duration=5)

    try:
        #API FUNCTION CALL DISABLE THIS TO DISABLE API
        #SENDAPI("Gold Price DROPPED!\n"+final)
print()
    except:
print("\nWhatsApp Alert Failed!")


    if(change[0]=="+"):
winsound.Beep(1000, 350)
winsound.Beep(1000, 350)
    if(os.path.exists('gold.ico')):
        toaster = win10toast.ToastNotifier().show_toast("Gold Price INCREASED!",final ,
duration=5, icon_path="gold.ico")
    else:
        toaster = win10toast.ToastNotifier().show_toast("Gold Price INCREASED!",final ,
duration=5)

```

```

try:
    #API FUNCTION CALL DISABLE THIS TO DISABLE API
    #SENDAPI("Gold Price INCREASED!\n"+final)
print()
except:
print("\nWhatsApp Alert Failed!")

except Exception as e:
print("City Not Found!")

time.sleep(5)    #Loops over every given seconds

```

Result Analysis:

The output alert will be either send via SMS or WhatsApp depending upon the choice of the user, if the price of gold has dropped the price drop alert will be send along with the current rate and change, change is calculated by current rate – yesterday rate. If the price of gold has increased the increase price alert will be send along with other information.

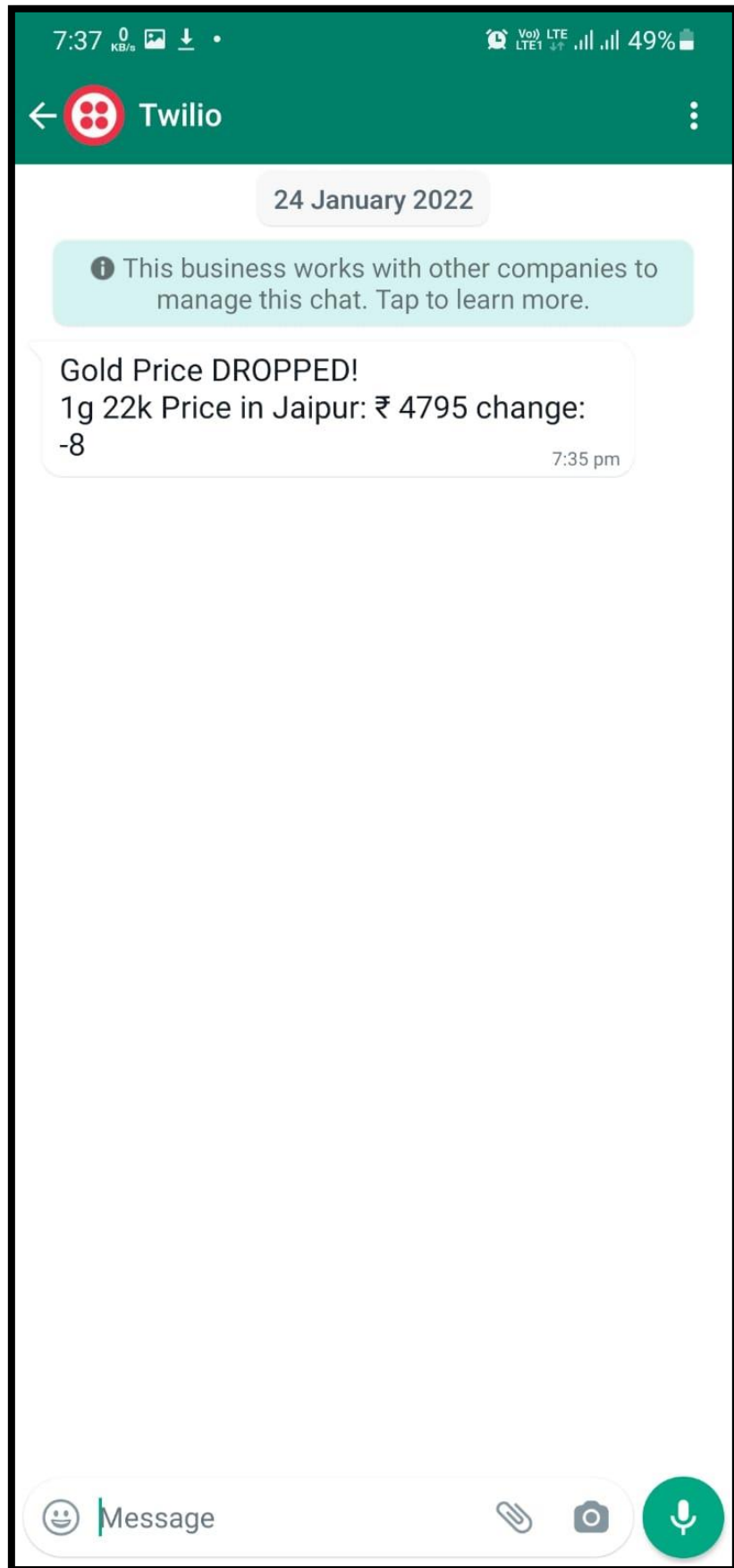
With the received information user can invest at dropped price and sell at higher price.

Output:

Following is the output screenshot of WhatsApp when market value of gold dropped in the given city of Jaipur.



Alert notification on Windows



Alert notification on WhatsApp

CONCLUSION AND FUTURE SCOPE

To conclude, this project is like “All in one combo pack”, as it provides multiple functionalities to users. User just has to set the program once and he will get updated accordingly. This application is good as it provides various stock related services to its users and also overcomes the problems like constantly checking the gold rates and many other limitations that use has to go through.

It provides:

- User-friendly environment
- Efficient results
- Save time and efforts of user
- Results of different cities
- Notification on WhatsApp/SMS on mobile phone
- Alerting sound system
- Stock guidance whether to sell or buy
- Notification on PC
- Keep records of old data for better insights

Future Scope:

The project has a very vast scope in future as we are living in a fast-growing world, we do not have enough time to look constantly for stock updates and put great efforts in a single source of income. The project can provide us alerts on your mobile phones and on our PC, we just have to do what the application recommends us to do which is literally a no-brainer.

Among all the investable assets, the gold market has always been the most mature one. Gold is one of the very few assets that are prone to high volatility or price swings. Gold, as in investment, has shown consistent value growth. The value of gold has been observed to be increasing day by day. That's the reason gold has always been viewed as a valuable asset for thousands of years.

Project can be updated and expanded in near future to accommodate various other investing opportunities like stock market, cryptocurrency, etc. and provide up-to-date information, investment alerts, storage of previous data for better insights and most importantly saving time and efforts of the user.

REFERENCES

1. BeautifulSoup Documentation:

<https://beautiful-soup-4.readthedocs.io/en/latest/>

2. Twilio Rest API:

<https://www.twilio.com/>

3. BeautifulSoup Scraping Tutorial:

<https://www.youtube.com/watch?v=87Gx3U0BDlo>

4. Win10toast Documentation:

<https://pypi.org/project/win10toast/>

5. WhatsApp Business API with Twilio:

<https://www.twilio.com/docs/whatsapp>

6. GitHub Repository:

https://github.com/kartikeyasingh6/kartik_python/tree/master/GoldPriceChecker

7. Working with CSV files in Python:

<https://www.geeksforgeeks.org/working-csv-files-python/>

8. Sound-playing interface for Windows:

<https://docs.python.org/3/library/winsound.html>

9. Python Datetime:

https://www.w3schools.com/python/python_datetime.asp

10. Scraping Website:

<https://www.fresherslive.com/gold-rate-today/delhi>