Summer Internship Project Report



Internship Title: Predicting Life Expectancy using Machine Learning –

SB51612

Project Title: Predicting Life Expectancy using Machine LearningProject Duration: 21/05/2020-18/06/2020  
Submitted by: Shubham

Email: mohitbit22@gmail.com



**Preface**This report documents is the work done during the summer internship at **SmartBridge Educational Services PVT Ltd**, for the prediction of life- expectancy under the supervision of **mentors of SmartBridge and smart practice school platform**. The report will give an overview of the tasks completed during the period of internship with technical details. Then the results obtained are discussed and analyzed. I have tried my best to keep report simple yet technically correct. I hope I succeed in my attempt.

Shubham

**Index  
1 INTRODUCTION**

1.1 Overview

1.2 Purpose  
**2 LITERATURE SURVEY**

2.1 Existing problem

2.2 Proposed solution  
**3 THEORITICAL ANALYSIS**

3.1 Block diagram

3.2 Hardware / Software designing  
**4 EXPERIMENTAL INVESTIGATIONS  
5 FLOWCHART  
6 RESULT**

**7 ADVANTAGES AND DISADVANTAGES  
8 APPLICATIONS  
9 CONCLUSION `**

**10 FUTURE SCOPE  
11 BIBILOGRAPHY  
APPENDIX**A. Source code

**1. INTRODUCTION**  
**1.1. Overview**  
Life Expectancy is one of the most important factor in end-of-life decision making. Life expectancy is a statistical measure of the average time a human being is expected to live, Life expectancy depends on various factors: Regional variations, Economic Circumstances, Sex Differences, Mental Illnesses, Physical Illnesses, Education, Year of their birth and other demographic factors. It is very important to predict average life expectancy of a country to analyse further requirements to increase its rate of growth or stabilise the rate of growth in that country. So this is a typical Regression Machine Learning project that leverages historical data to predict insights into the future.

The end product will be a webpage where you need to give all the required inputs and then submit it .  
Afterwards it will predict the life expectancy value based on your regression technique.  
**Project Requirements**: Python, IBM Cloud, IBM Watson  
**Functional Requirements**: IBM cloud  
**Technical Requirements**: ML, WATSON Studio, Python, Node-Red  
**Software Requirements**: Watson Studio, Node-Red  
**Project Deliverables**: Smartinternz Internship  
**Project Team**: Shubham  
**Project Duration**: 23.5 Days

**1.2. Purpose**Predicting Life Expectancy using Machine Learning A typical Regression Machine Learning project leverages historical data to predict insights into the future. This problem statement is aimed at predicting Life Expectancy rate of a country given various features. The purpose of project is help to country to know their rate of life expectancy, which factors affects more to decrease life expectancy so that they can take appropriate decision to increase life expectancy of human being in their country.

**2. LITERATURE SURVEY**

**2.1. Existing Problem :**Country must know about Life Expectancy of their country. If country know about  
their life expectancy they can understand which factors affects more to decrease life expectancy of their country, so that they can take appropriate decision to increase life  
expectancy of human being in their country.

**2.2. Proposed Solution**

Predicting life expectancy is not a new concept. Experts do this at a population level by classifying people into groups, often based on region or ethnicity. Also, tools such as deep learning and artificial intelligence can be used to consider complex variables, such as biomedical data, to predict someone’s biological age. Biological age refers to how “old” their body is, rather than when they were born. A  
30-year-old who smokes heavily may have a biological age closer to 40. Calculating a life expectancy reliably would require a sophisticated system that considers a breadth of environmental, geographic, genetic and lifestyle factors – all of which have influence.

Predicting life Expectancy using Machine Learning project will help country to know their life expectancy.so that they can understand which factors affects more to decrease life expectancy of their country, so that they can take appropriate decision to increase life expectancy of human being in their country.

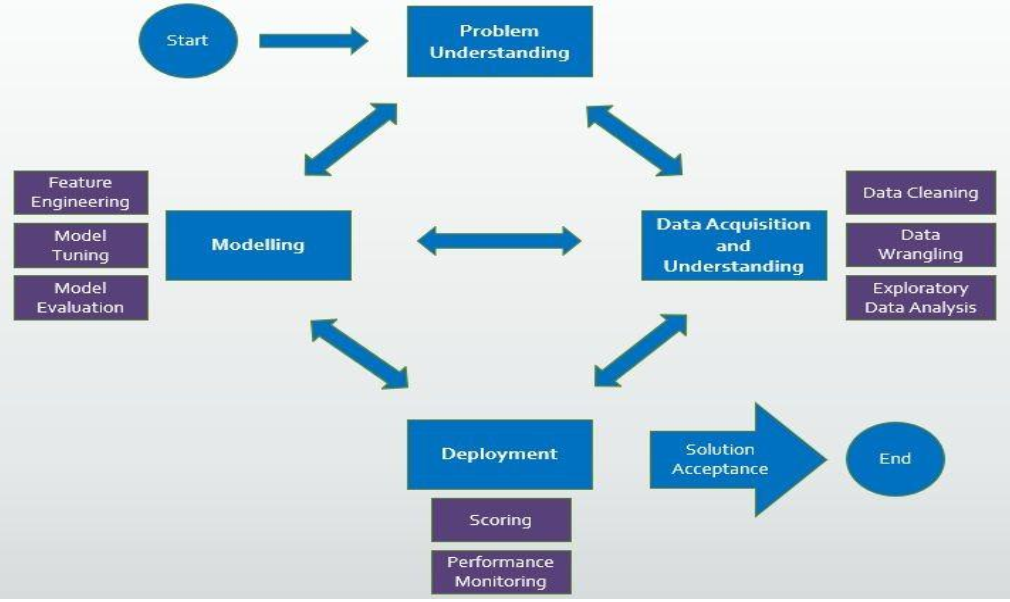
With machine learning and artificial intelligence, it’s becoming feasible to analyse larger quantities of data. The use of deep learning and cognitive computing, such as with IBM Watson, helps doctors make more accurate diagnoses than using human judgement alone.

This project take following aspects (features) as input:  
1. Country  
2. Status  
3. Life Expectancy  
4. Adult Mortality  
5. Alcohol  
6. percentage expenditure  
7. Hepatitis B  
8. Measles  
9. BMI  
10. under-five deaths  
11. Polio  
12. Total expenditure  
13. Diphtheria  
14. HIV/AIDS  
15. GDP  
16. Population  
17. thinness 1-19 years  
18. thinness 5-9 years  
19. Income composition of resources  
20. Schooling

Target is Life Expectancy, measured in number of years.  
The assumptions are:

1.These are country level average  
2.There is no distinction between male and female.

**3. THEORITICAL ANALYSIS  
 3.1. Block diagram**



**3.2. Hardware / Software designing**

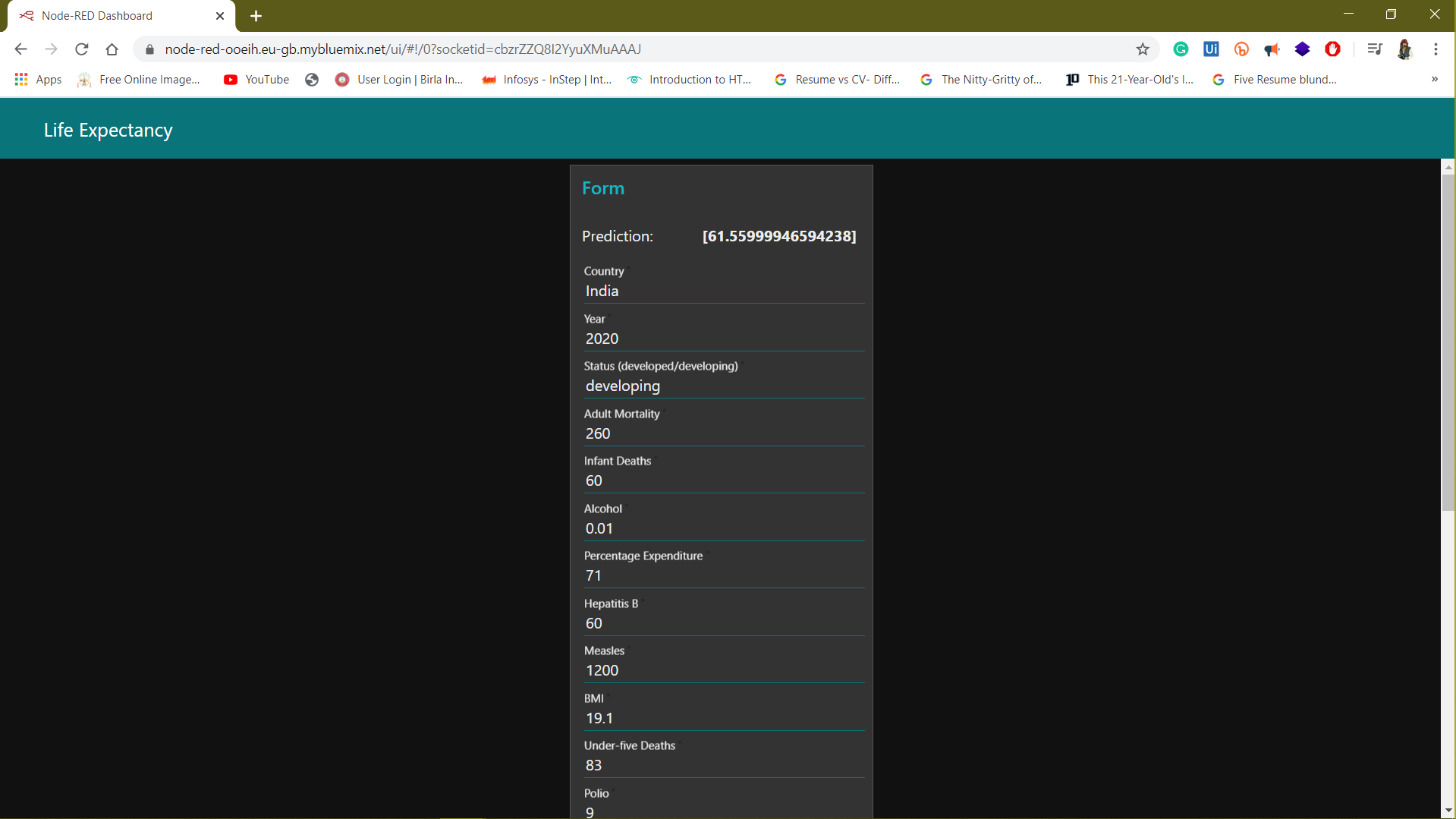
**Hardware :**  
• Internal hardware devices include Motherboard,Hard-Drives,RAM.  
• External hardware devices include Monitor,keyboard, mouse.  
**Software:**  
• IBM cloud Software :

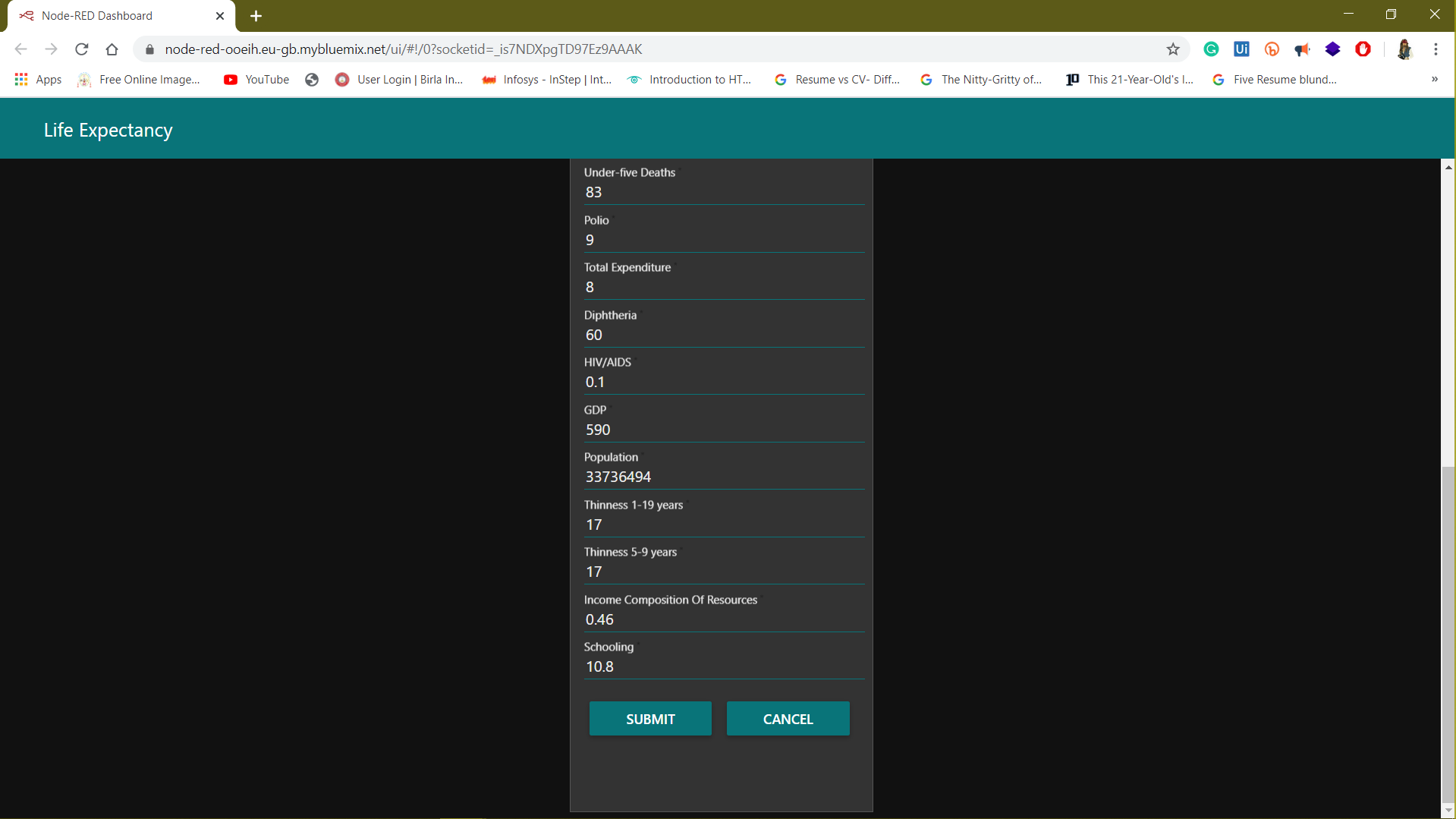
-Node-Red App[Webpage].  
• Programming Environment :

-Default Python 3.6 XS

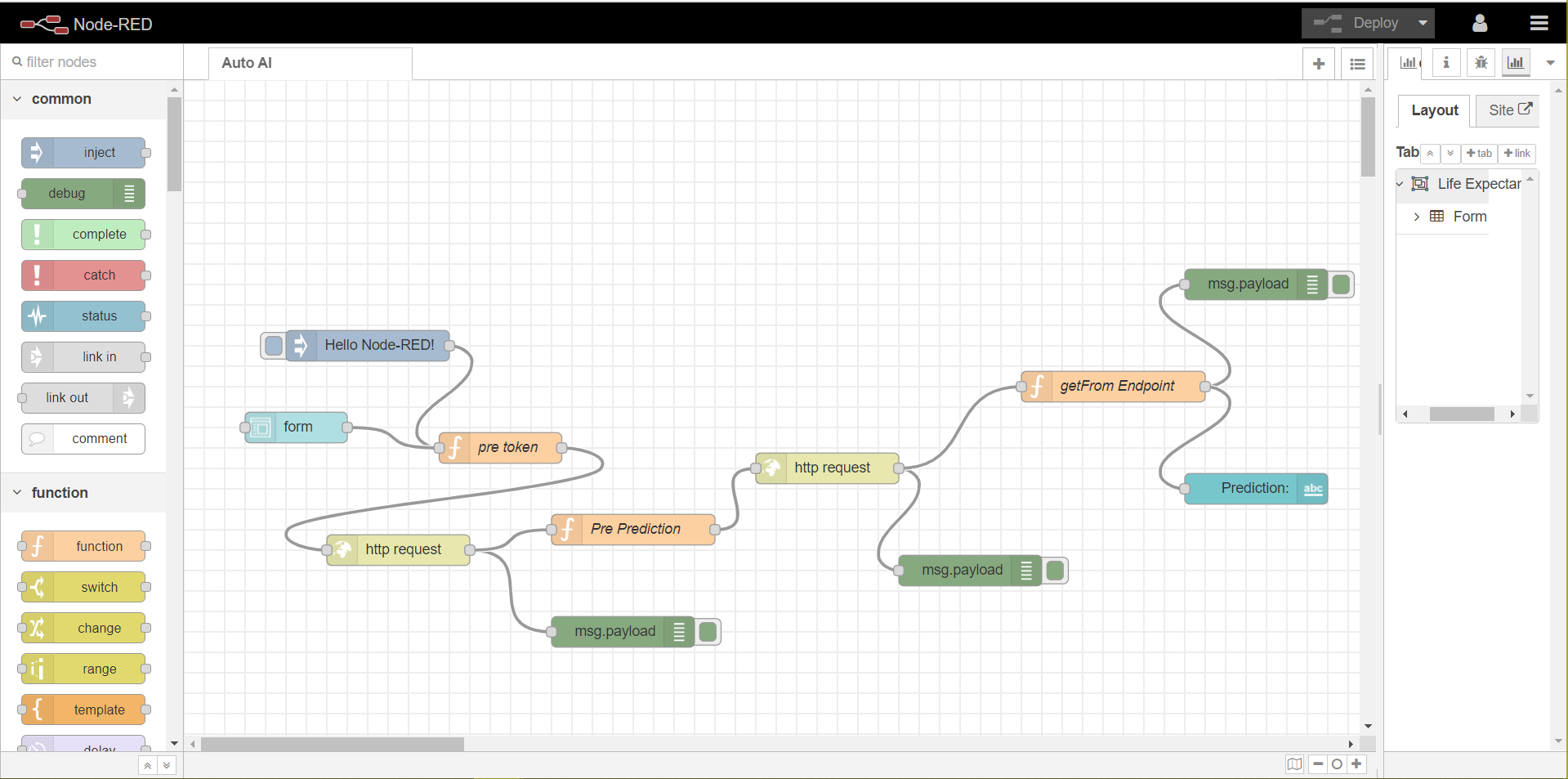
A node red flow is made for the prediction of life expectancy. Which is given below.

<https://node-red-ooeih.eu-gb.mybluemix.net/ui/#!/0?socketid=cbzrZZQ8I2YyuXMuAAAJ>

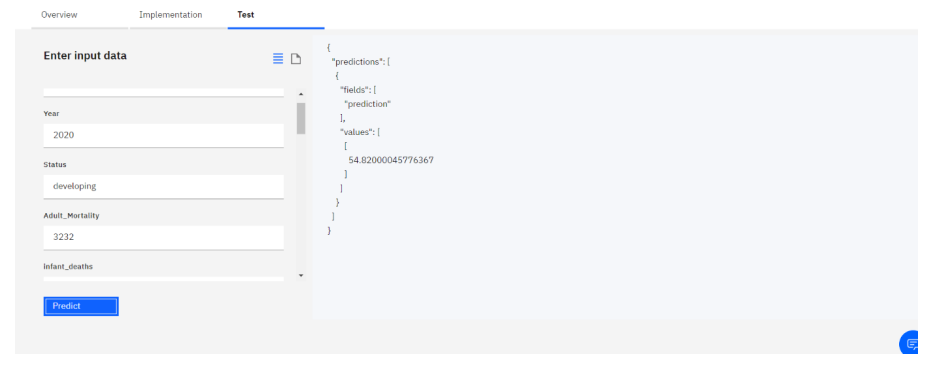
****



Demonstration of node red flow

****

Auto AI model



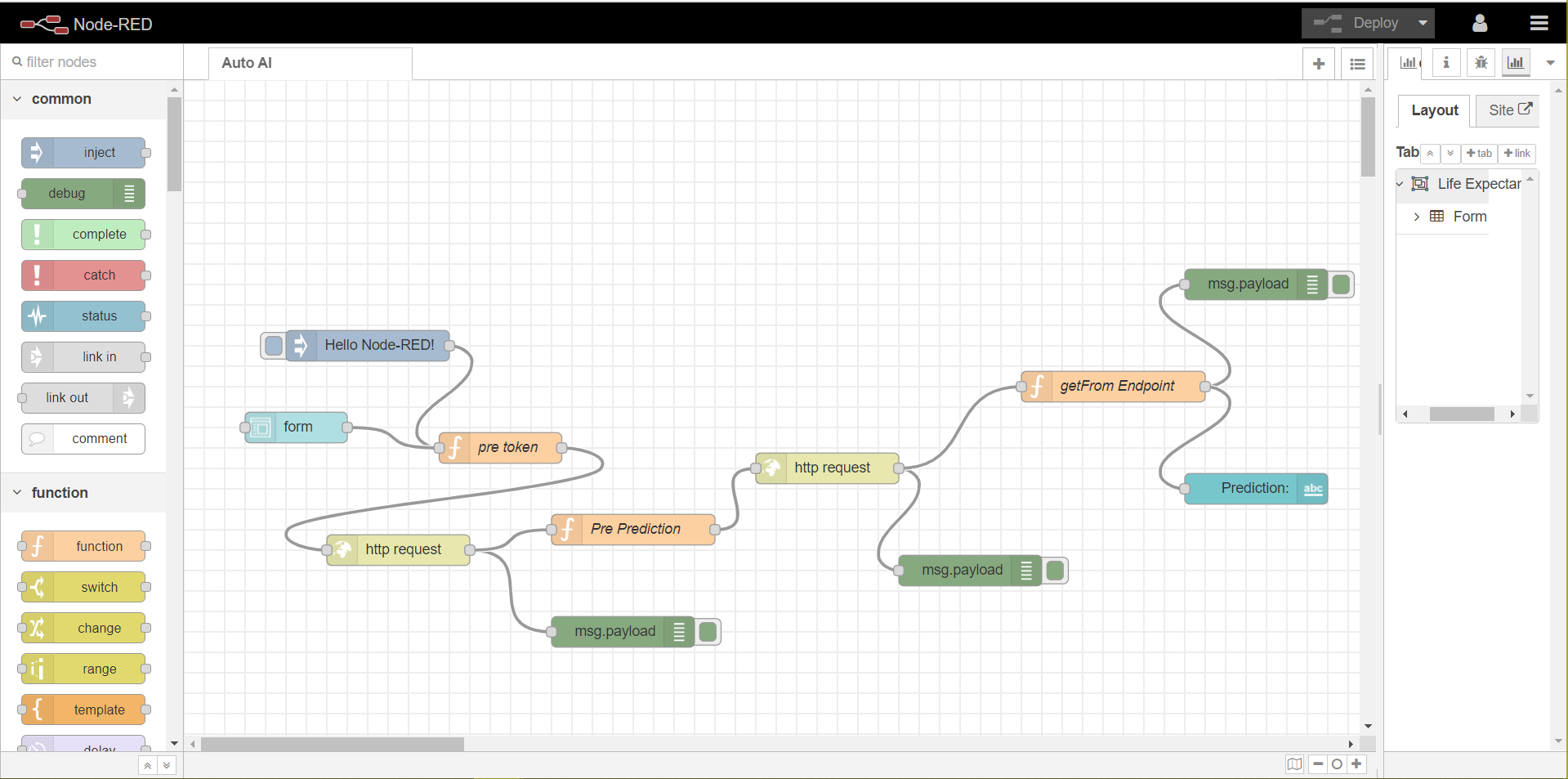
**4. EXPERIMENTAL INVESTIGATIONS**

• Collection of data set from Kaggle.  
https://www.kaggle.com/c/predicting-life-expectancy/data

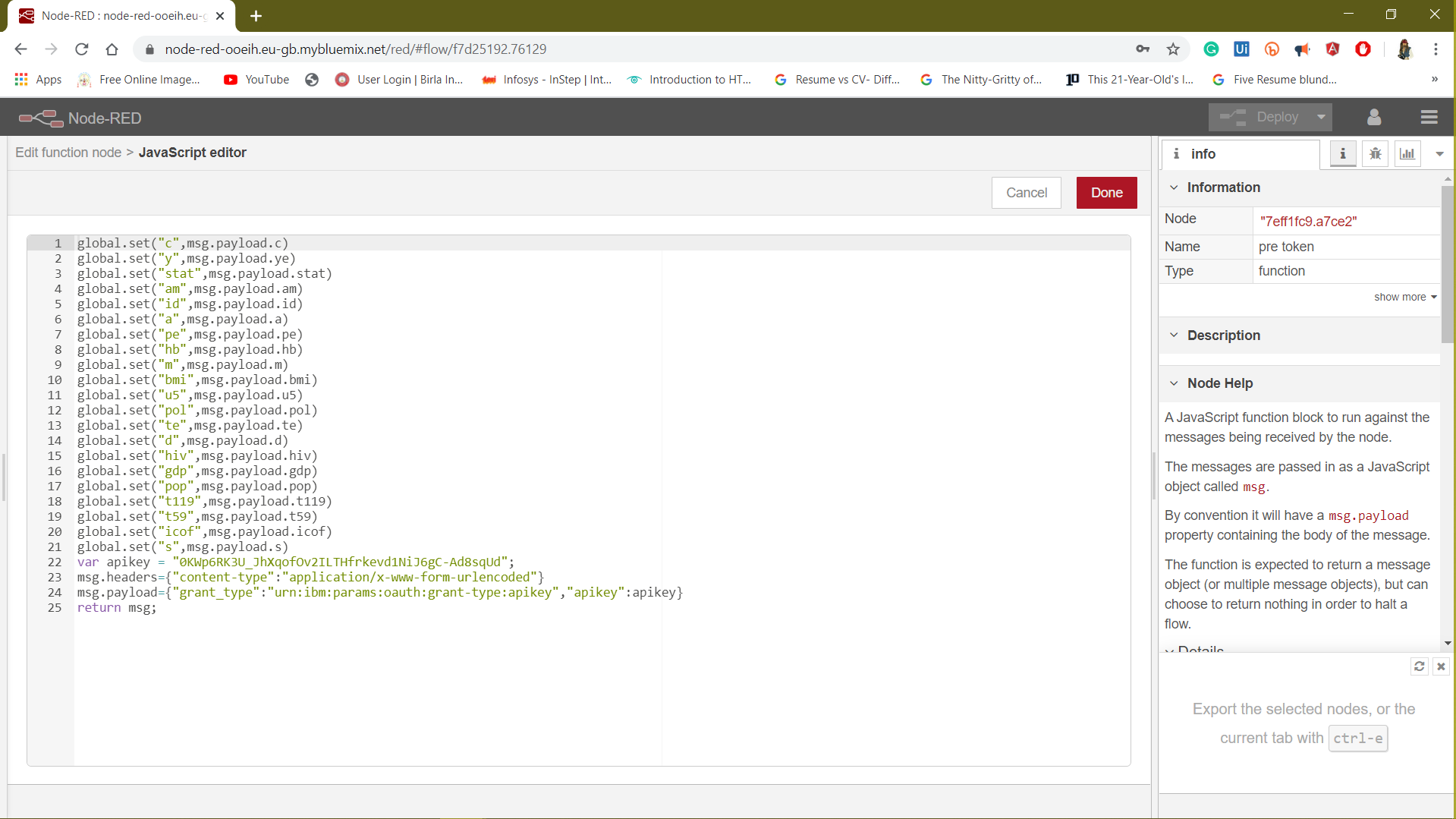
* On IBM Watson studio machine learning using auto ai build a model to predict  
  life expectancy.  
  ➢ To do so first create account on IBM Watson studio.  
  ➢ Using Add to project choose auto ai.  
  ➢ Then upload data set  
  ➢ Choose best way to predict.  
  ➢ Save as a model which is on the top  
  ➢ Deploy the model.  
  ➢ Test the model.  
  ➢ Create service credential

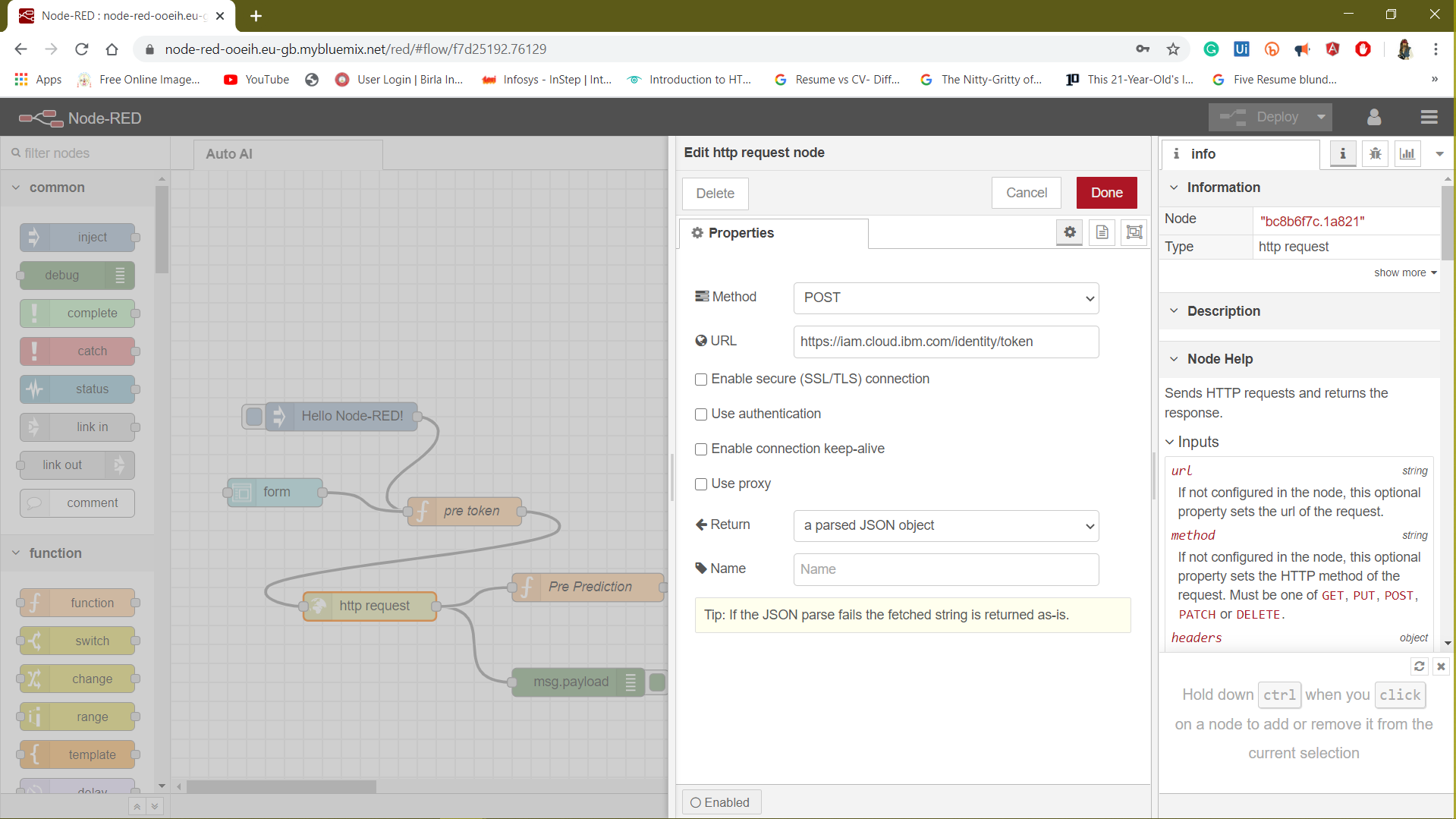
• Create cloud foundry app

<https://node-red-ooeih.eu-gb.mybluemix.net/red/#flow/f7d25192.76129>  
• Make node red flow

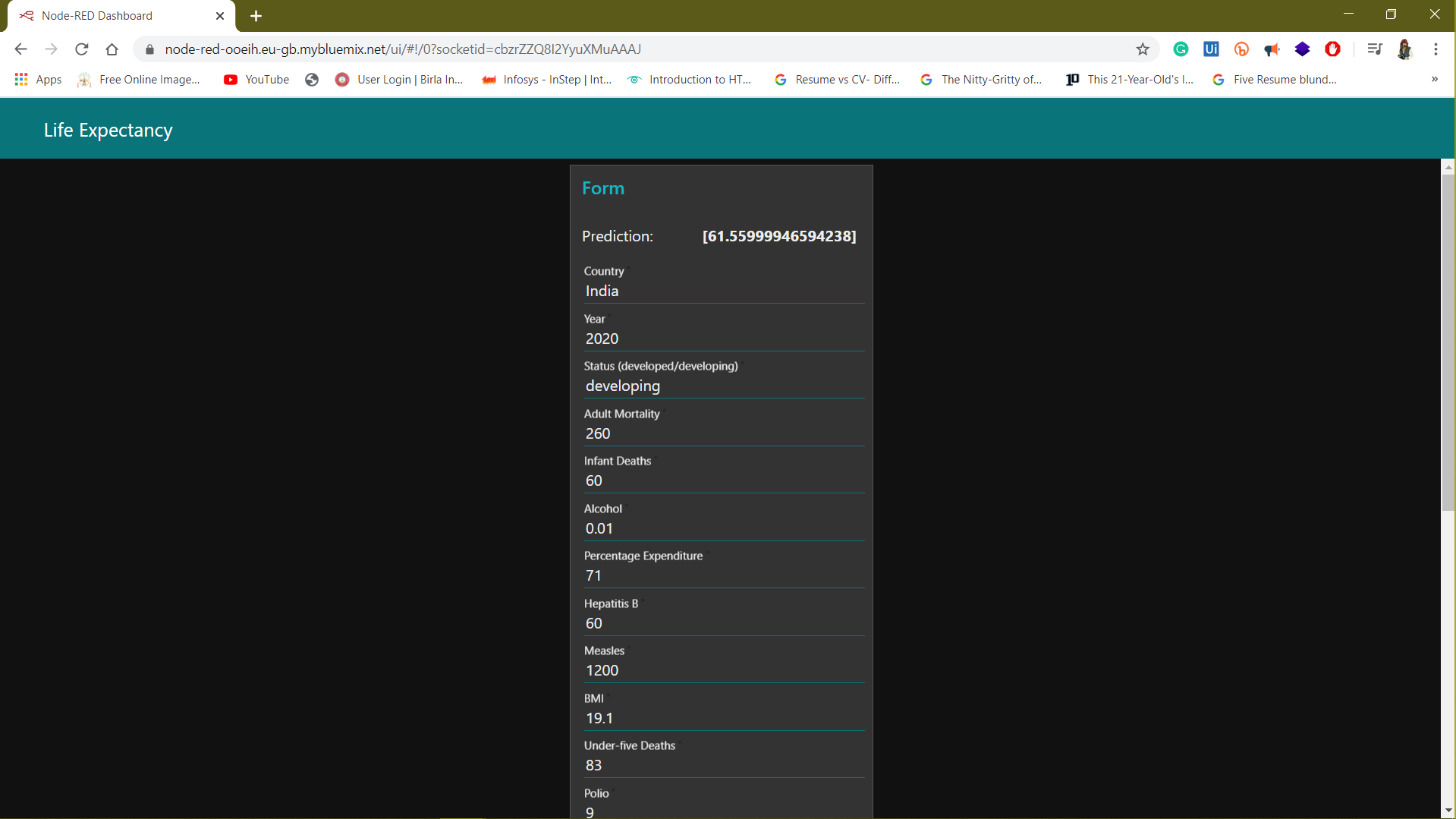
****

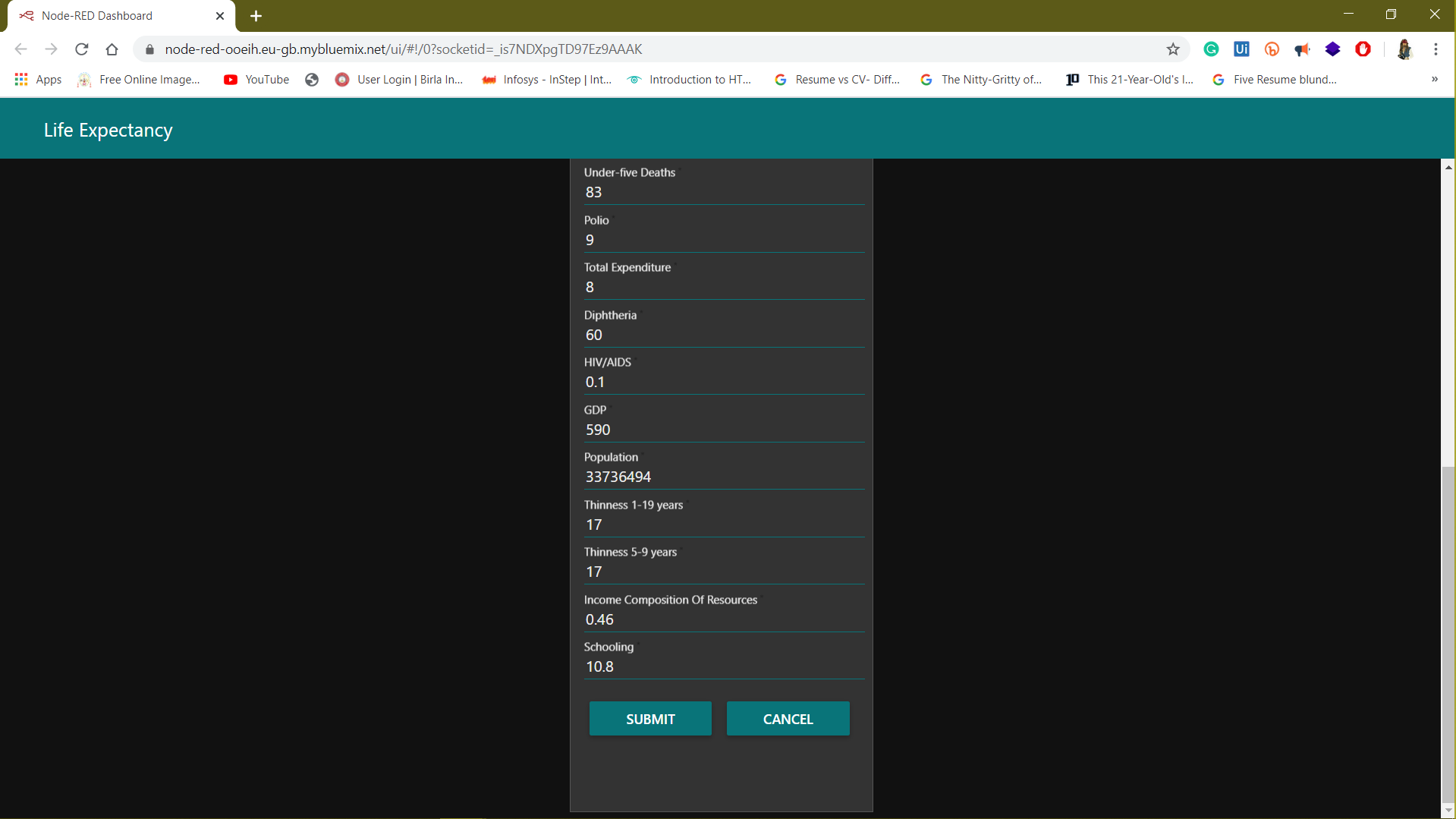
• Then add API key, instance Id and url.



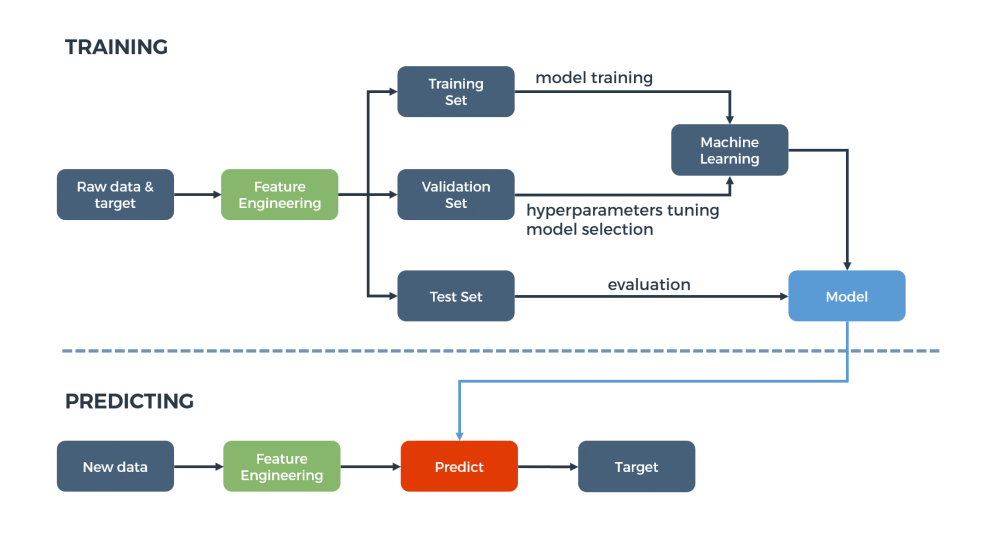


• After deploying the model from dashboard UI can be seen.

****

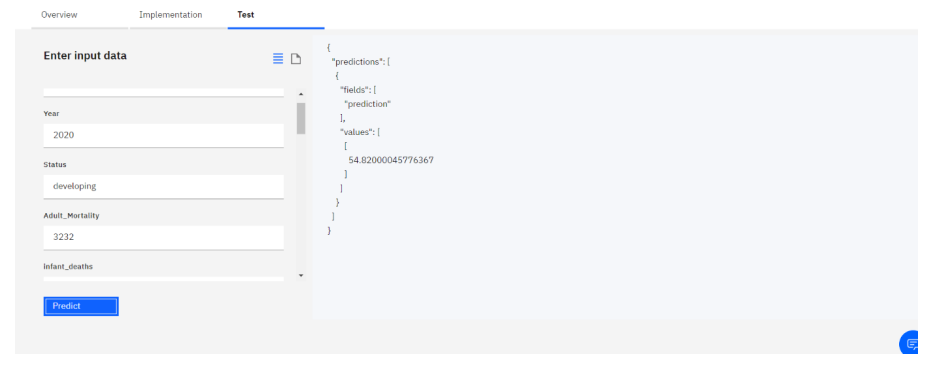


**5. FLOWCHART**

****

**6. RESULT**

**Prediction** of life expectancy based on country, year, status, adult molarity, GDP and population etc.



Based on the given data, the auto ai understands the data and cross reference the  
data to watch what are the factors that are affecting the results we require i.e life  
expectancy.  
Then when we give any input, it has already run algorithm to get the output based  
on previously given data. So the results we get are approximations, they are not  
definitely true, but it works in maximum number of cases, except for some exceptional  
ones.

**7. ADVANTAGES AND DISADVANTAGES  
ADVANTAGES:**a) Health Inequalities: Life expectancy has been used nationally to monitor health inequalities of a country.  
b) Reduced Costs: This is a simple webpage and can be accessed by any citizen of a country to calculate life expectancy of their country and doesnot required any kind of payment neither for designing nor for using.  
c) User Friendly Interface: This interface requires no background knowledge of how to use it. It’s a simple interface and only ask for required values and predict the output.  
**DISADVANTAGES:**  
a) Wrong Prediction: As it depends completely on user, so if user provides some wrong values then it will predict wrong value.  
b) Average Prediction: The model predicts average or approximate value with 97.07% accuracy but not accurate value.

**8. APPLICATIONS**

● Predicting life expectancy will help to country to know their average rate of life  
expectancy.  
● Country can analyse what factors affects more to increase life expectancy.  
● Country can also analyse which factors affects more to decrease life expectancy so  
that they can take appropriate decision to increase life expectancy of human being  
in their country.

**9. CONCLUSION**Some interesting correlations here:  
• There is a strong positive correlation between ‘Schooling’ and ‘Life Expectancy’. This may be because education is more established and prevalent in wealthier countries. This means countries with less corruption, infrastructure, healthcare, welfare, and so forth.  
• Similarly to the point above, there is a moderate positive correlation between ‘GDP’ and ‘Life Expectancy’, most likely due to the same reason.  
• Surprisingly there’s a moderate positive correlation between ‘Alcohol’ and ‘Life Expectancy’. I’m guessing that this is due to the fact that only wealthier countries  
can afford alcohol or the consumption of alcohol is more prevalent among wealthier populations.

Similarly more result can be abstract.

Predicting Life Expectancy using Machine Learning project will help country to know their life expectancy.

**10. FUTURE SCOPE**

The problem of processing datasets such as electronic medical records(EMR) and their integration with genomics, environmental factors, socioeconomic factor and patient behavior variations have posed a problem for researchers the health industry. Due to rapid innovations in machine learning field such as big data, analytics, visualization, deep learning, health workers now have improved way of processing, and developing meaningful information from huge datasets that have been accumulated over many years.  
Big data and machine learning can benefit public health researchers with analyzing thousands of variables to obtain data regarding life expectancy. We can use demographics of selected regional areas and multiple behavioral health disorders across regions to find correlation between individual behavior indicators and behavioral health outcomes.

Future Scope of the Model can be:

a) Feature Reduction  
It requires much more data about 21 columns to be known prior for predicting life expectancy  
which can be again difficult for a normal user to gather such datas so I have decided to do  
some kind of feature reduction or replacement of some features as individuals or groups to  
make it more user friendly.

b) Attractive UI  
It is a simple webpage only asking inputs and predict output. In future I have decided to make  
it more user friendly by providing some useful information about the country in the webpage  
itself so that user does not need to do any kind of prior research for the values.

c) Integrating with services such as speech recognition

**11. BIBLIOGRAPHY**

▪ https://cloud.ibm.com/docs/overview?topic=overview-whatis-platform  
▪ https://developer.ibm.com/tutorials/how-to-create-a-node-red-starter-application/  
▪ https://nodered.org/  
▪ https://github.com/watson-developer-cloud/node-red-labs  
▪ https://www.youtube.com/embed/r7E1TJ1HtM0  
▪ https://bookdown.org/caoying4work/watsonstudio-workshop/jn.html  
▪ https://www.kaggle.com/kumarajarshi/life-expectancy-who  
▪ https://www.youtube.com/watch?v=Jtej3Y6uUng  
▪ https://bookdown.org/caoying4work/watsonstudio-workshop/jn.html#deploy-model-as-webservice  
▪ https://machinelearningmastery.com/columntransformer-for-numerical-and-categorical-data/

**APPENDIX:**

**Source Code**

1)Notebook  
**#import librarires**import pandas as pd  
import numpy as np  
import os

import matplotlib.pyplot as plt  
import seaborn as sns  
import warnings  
from sklearn.pipeline import Pipeline  
from sklearn.preprocessing import OneHotEncoder  
from sklearn.impute import SimpleImputer  
from sklearn.preprocessing import StandardScaler  
from sklearn.compose import ColumnTransformer  
from sklearn.model\_selection import train\_test\_split  
from sklearn.ensemble import ExtraTreesRegressor  
from sklearn.metrics import mean\_squared\_error, r2\_score  
from watson\_machine\_learning\_client import WatsonMachineLearningAPIClient  
**#import dataset  
#Data Preprocessing**df=df.rename(columns={‘old name:new name’})  
df=df.fillna(df.mean())  
**#Exploratory data Analysis**df\_kor=df.corr()  
plt.figure(figsize=(10,10))  
#heatmap  
sns.heatmap(df\_kor,vmin=-1,vmax=1,annot=True,linewidth=0.1)  
#pairplot  
sns.pairplot(df)  
**#Train&Test**Y=df['Life expectancy']  
X=df[df.columns.difference(['Life expectancy'])]  
categorical\_features = ['Country', 'Status']  
categorical\_feature\_mask = X.dtypes==object  
categorical\_features = X.columns[categorical\_feature\_mask].tolist()  
categorical\_transformer = Pipeline(steps=[  
('onehot', OneHotEncoder(handle\_unknown='ignore')),  
])  
numeric\_features = ['Year','Adult Mortality','infant deaths','Alcohol','percentage expenditure', 'Hepatitis B',  
'Measles', 'BMI', 'under-five deaths ', 'Polio', 'Total expenditure','Diphtheria', 'HIV/AIDS', 'GDP',  
'Population',  
'thinness 1-19 years', 'thinness 5-9 years','Income composition of resources', 'Schooling']  
numeric\_feature\_mask = X.dtypes!=object

numeric\_features = X.columns[numeric\_feature\_mask].tolist()  
numeric\_transformer = Pipeline(steps=[  
('imputer', SimpleImputer(strategy='median')),  
('scaler', StandardScaler()),  
])  
preprocessor = ColumnTransformer(  
transformers=[  
('num', numeric\_transformer, numeric\_features),  
('cat', categorical\_transformer, categorical\_features)  
]  
)  
ExtraTreeRegressor = Pipeline([  
('preprocessor', preprocessor),  
('ExtraTreeRegressor', ExtraTreesRegressor(n\_estimators=100, random\_state=0))  
])  
reg = ExtraTreeRegressor.fit(X\_train, Y\_train)  
test\_pred=reg.predict(X\_test)  
**#Model Building and Deployment**wml\_credentials={  
"apikey": "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*",  
"instance\_id": "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*",  
"url": "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"  
}  
client = WatsonMachineLearningAPIClient(wml\_credentials)

model\_props = {client.repository.ModelMetaNames.AUTHOR\_NAME: "\*\*\*\*",  
client.repository.ModelMetaNames.AUTHOR\_EMAIL: "\*\*\*\*\*\*\*\*\*\*",  
client.repository.ModelMetaNames.NAME: "LifeExpectancy"}  
model\_artifact=client.repository.store\_model(ExtraTreeRegressor, meta\_props=model\_props)  
model\_uid = client.repository.get\_model\_uid(model\_artifact)  
create\_deployment = client.deployments.create(model\_uid, name="LifeExpectancyPrediction")  
scoring\_endpoint = client.deployments.get\_scoring\_url(create\_deployment)  
print(scoring\_endpoint)  
2) Flow.json  
[{"id":" }]