# EARLY DETECTION OF CHRONIC KIDNEY DISEASE USING MACHINE LEARNING

# PROJECT REPORT

## TEAM ID: PNT2022TMID46343


## ABSTRACT

ChronicKidney Disease (CKD) is a major public health concern with rising prevalence.Chronic kidney disease is a fatal illness of kidney which can be prevented with early correct predictions and proper precautions.Data mining and machine learning play a vitalrole in health care and also medical information and detection, Chronic kidney disease (CKD), otherwise referred to as renal disease. So, it is required to be concerned about kidney disease to our very primary stage. We take a few attributes to measure our analysis about chronic kidney disease and this attribute is one of the major occurrences of chronic kidney disease. Innumerable feature selection methods have been presented in state-of-arts literature to tackle the problems of high dimensional data.Many evolutionary and swarm intelligence algorithms find solutions based on algorithm-specific control parameters. However, it is a challenging task to identify the optimal feature subset using a feature selection algorithm that is not dependent on the controlling parameters of an algorithm that is specific to a particular problem in hand. However, specific techniques must be executed to accomplish better consequence. This paper presents a machine-learning-based approach using to detect chronickidney disease; The new approach delivers results highly competitive with existing approaches

# 1.INTRODUCTION

Currently predictive analytics is the most significant area for analysis. In multitudinous field, totally varied applied mathematics and predictive analytics algorithms area unit enforced. Within the medici nal area, several conditions may be identified by device or expected by device by the applying the predictive analytics algorithms. Chronic conditions be get a veritably important trouble to the worldwide healthy problem of the twenty first centenary. The rising frequence of habitual condition like habitual uropathy has important problems for health and profitable affair in developing nations. The faster rise in usual threat factors, particularly between the poor, like polygenic disease, high blood pressure, and blubber, would lead to even larger and huge burdens that developing society don't sees be ready to handle with. There had been a failure of exposure to habitual conditions, uropathy above all, primarily thanks to the main target of the worldwide health community on contagious conditions, and lack of knowledge. The significant got focus in it and also the assumption of a lot of comprehensive, cost-efficient, and prevent illness ways by developed nations. The area unit several clinics that keep the data of chronic uropathy case in there info. By analyse the data, varied design may be find which can be useful for making decision. Victimization information processing methodology on that information, it is attainable for finding several styles of facts and makes use this information to find the illness. There is a huge quantity individuals World Health Organization area unit littered with chronic Uropathy. Within the raised technical power, discovery of assorted system and storage brooding again range clinical information may be collect and keep recently offered to medicinal analysis organizations for more analyses

## 1.2 PROJECT OVERVIEW

Chronic Kidney Disease (CKD) is a major public health concern with rising prevalence. Chronic kidney disease is a fatal illness of kidney which can be prevented with early correct predictions and proper precautions. Data mining and machine learning play a vital role in health care and also medical information and detection, chronic kidney disease (CKD), otherwise referred to as renal disease. So, it is required to be concerned about kidney disease to our very primary stage. We take a few attributes to measure our analysis about chronic kidney disease and this attribute is one of the major occurrences of chronic kidney disease. Innumerable feature selection methods have been presented in state-of-arts literature to tackle the problems of high dimensional data. Many evolutionary and swarm intelligence algorithms find solutions based on algorithm-specific control parameters. However, it is a challenging task to identify the optimal feature subset using a feature selection algorithm that is not dependent on the controlling parameters of an algorithm that is specific to a particular problem in hand. However, specific techniques must be executed to accomplish better consequence. This paper presents a machine-learning-based approach using to detect chronic kidney disease; the new approach delivers results highly competitive with existing approaches.

## 1.2 PURPOSE

In these project renal disease dataset is collected. Data is pre-processed after collection of various records. Raw data is converted into refined data using data pre-processing. The dataset is partitioned based on the decision tree diagram which i got from the rattle. I applied classification algorithm for the splitted dataset. The best classifier algorithm is selected based on the highest accuracy. Entropy, it will partition the dataset. Then hybridisation is done. The Hybrid machine learning model combine or integrate(ensemble learning) different classifier algorithms which gave highest accuracy. With this the performance is optimized by classifier features into overall accuracy.

## 2. LITERATURE SURVEY

| SNO | TITLE | AUTHOR | DESCRIPTION | YEAR OF PUBLICATION |
|---|---|---|---|---|
| 1 | Optimal Feature Selection for Chronic Kidney Disease Classification using Deep Learning Classifier | K.Shankar, P. Manickam, G. Devika,M. Ilayaraja | **ALGORITHM USED:** Deep Neural Network (DNN). **ACCURACY:** Greatest accuracy around 98% of DNN. | 2018 |
| 2 | Performance Evaluation of an Ensemble Method for Diagnosis of Chronic Kidney Diseasewith Feature Selection Technique | Olayinka Ayodele Jongbo, Toluwase Ayobami OlowooAdebayo Olusola Adetunmbikere. | **ALGORITHM USED:** Radom forest algorithm ,Naïve Bayes, kNearest Neighbor, and Decision Tree. **ACCURACY:** We gain the highest accuracy from the Random Forest(RF) and it is 98.3%. | 2020 |
| 3 | XGBoost Model for Chronic Kidney Disease Diagnosis | Adeola Ogunleye and Qing-Guo Wang | **ALGORITHM USED:** Extreme Gradient Boosting (XGBoost). **ACCURACY:** Acucuracy gained 89% | 2018 |
| 4 | Performance Analysis of Chronic Kidney Disease through Machine Learning Approaches | Minhaz Uddin Emon, Al Mahmud Imran, Rakibul Islam. | **ALGORITHM USED:** Logistic Regression(LG), Naive Bayes(NB), Multilayer Perceptron(MLP), Stochastic Gradient Descent(SGD), Adaptive Boosting(Adaboost), Bagging, Decision Tree(DT), Random Forest(RF) classifier are used. | 2021 |

| | | | | |
|---|---|---|---|---|
| | | | **ACCURACY:**<br><br>We gain the highest accuracy from the Random Forest(RF) and it is 99 | |
| 5 | Preemptive Diagnosis of Chronic Kidney Disease Using Machine Learning Techniques | ReemA. Alassaf , Khawla A. Alsulaim ,Noura Y. Alroomi , Nouf S. Alsharif, Mishael F. Aljubeir, Sunday O. Olatunji, Alaa Y. Alahmadi, Mohammed Imran, Rahma A. Alzahrani, Nora S. Alturayeif. | **ALGORITHM USED:**<br><br>ANN, SVM, Naïve Bayes .<br><br>**ACCURACY:**<br><br>ANN, SVM, Naïve Bayes achieved a testing accuracy of 98.0% while k-NN has achieved an accuracy of 93.9%. | 2018 |
| 6 | Featue selection effects on kidney disease analysis. | Zeinab Sedighi, Hossein Ebrahimpour-Komleh, Seyed Jalaleddin Mousavirad . | **ALGORITHM USED:**<br><br>AdaBoost,Adaptive Boosting,Naïve Bayes classifier.<br><br>**ACCURACY:**<br><br>92% of accuracy gain. | 2015 |
| 7 | Analysis of Chronic Kidney Disease Dataset by Applying Machine Learning Methods. | Yedilkhan Amirgaliyev, Shahriar Shamiluulu, Azamat Serek. | **ALGORITHM USED:**<br><br>support vector machines<br><br>**ACCURACY:**<br><br>Experimental results showed over 93% of success | 2019 |
| 8 | Predictive Analytics for Chronic Kidney Disease Using Machine Learning Techniques. | Anusorn Charleonnan, Thipwan Fufaung, Tippawan Niyomwong . | **ALGORITHM USED:**<br><br>K-nearest neighbors (KNN), support vector machine (SVM), logistic regression (LR), and decision tree classifiers.<br><br>**ACCURACY:**<br><br>96% of accuracy. | 2016 |
| 9 | Chronic Kidney Disease for Collaborative | V.Shanmuga | **ALGORITHM USED:** | 2021 |

| | | | | |
|---|---|---|---|---|
| | Healthcare Data Analytics using Random Forest Classification Algorithms. | rajeshwari , M. Ilayaraja. | Random forest, SVM and ANN algorithms.<br><br>**ACCURACY:**<br><br>87% of accuracy. | |
| 10 | Diagnosis of Chronic Kidney Disease using effective classification and feature selection technique | Nusrat Tazin, Shahed Anzarus Sabab,<br><br>Muhammed Tawfiq Chowdhury. | **ALGORITHM USED:**<br><br>Support Vector Machine, Decision tree, Naïve Bayes and K-Nearest Neighbor.<br><br>**ACCURACY:**<br><br>93% of accuracy. | 2016 |

## 2.1 EXISTING PROBLEM

- They applied algorithms to dataset and finalize which is best algorithm based on highest accuracy. And they concluded the project.
- They did not use that much methods to improve accuracy. They used algorithms which is already existing.

## 2.2 REFERENCE

1. https://www.researchgate.net/publication/335698017_Detection_of_Chronic_Kid ney_Disease_using_Machine_Learning_Algorithms_with_Least_Number_of_Predict ors

2. https://pubmed.ncbi.nlm.nih.gov/34211680/

3. https://www.primescholars.com/articles/early-prediction-of-chronic-kidney-diseaseby-using-machine-learning-techniques-92643.html

4. https://www.webology.org/abstract.php?id=3038

5. https://www.ijert.org/chronic-kidney-disease-prediction-using-machine-learning

6. https://pubmed.ncbi.nlm.nih.gov/34372817/

7.https://www.researchgate.net/publication/330637326_Machine_LearningBased_Prediction_System_For_ Chronic_Kidney_Disease_Using_Associative_Classif ication_Technique

8. https://ijcsmc.com/docs/papers/July2014/V3I7201499a42.pdf

9. https://www.researchgate.net/publication/344335234_Chronic_Disease_Detection_M odel_Using_Machine_Learning_Techniques

10. 10. https://www.researchgate.net/publication/359618037_An_Augmented_Artificial_Intel ligence_Approach_for_Chronic_Diseases_Prediction

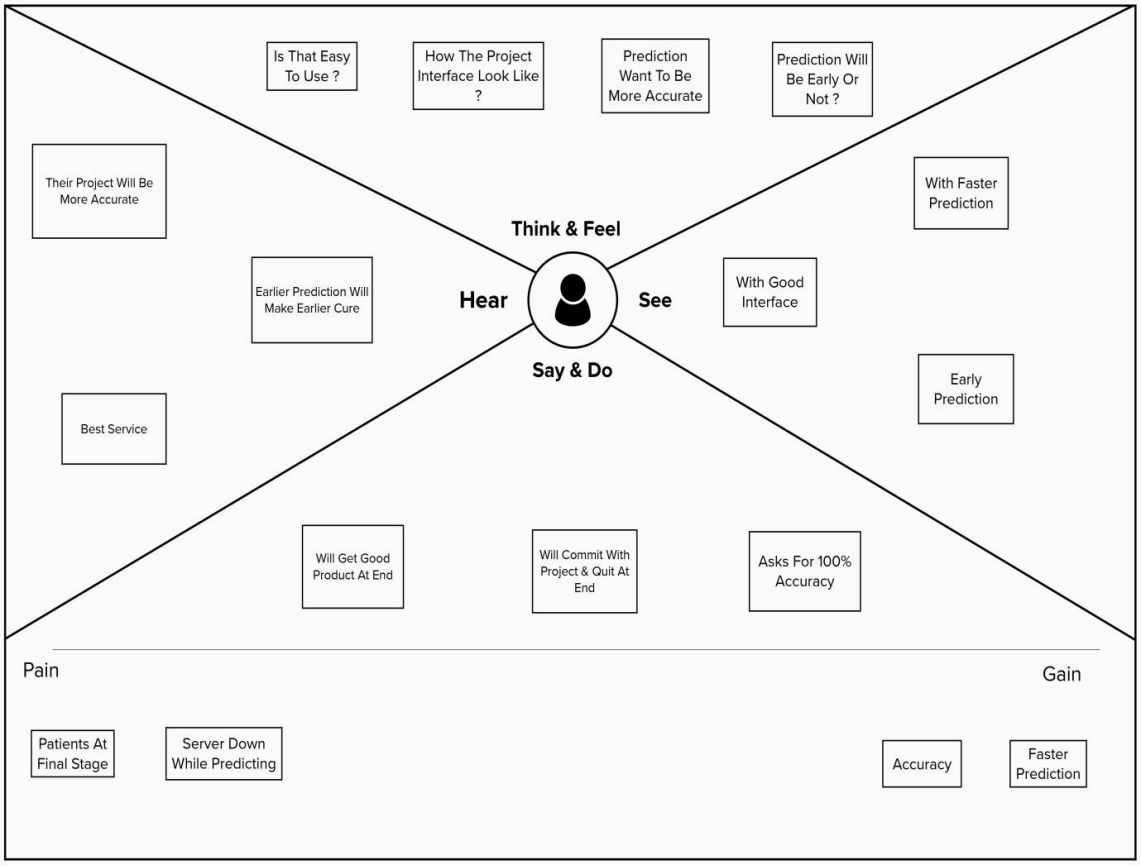
## 2.3 PROBLEM STATEMENT DEFINITION

Kidney diseases avert the normal function of the kidney. Due to the large amount of alcohol consumption kidney disease arises. Early prediction of kidney disease using classification and regression algorithms are an efficacious task that can help the doctors to diagnose the disease within a short duration of time. Discovering the existence of kidney disease at an early stage is complex task for the doctors. The main objective of this project is to analyses the parameters of various classification algorithms and compare their predictive accuracy, to find out the best classifier for determining the kidney disease.

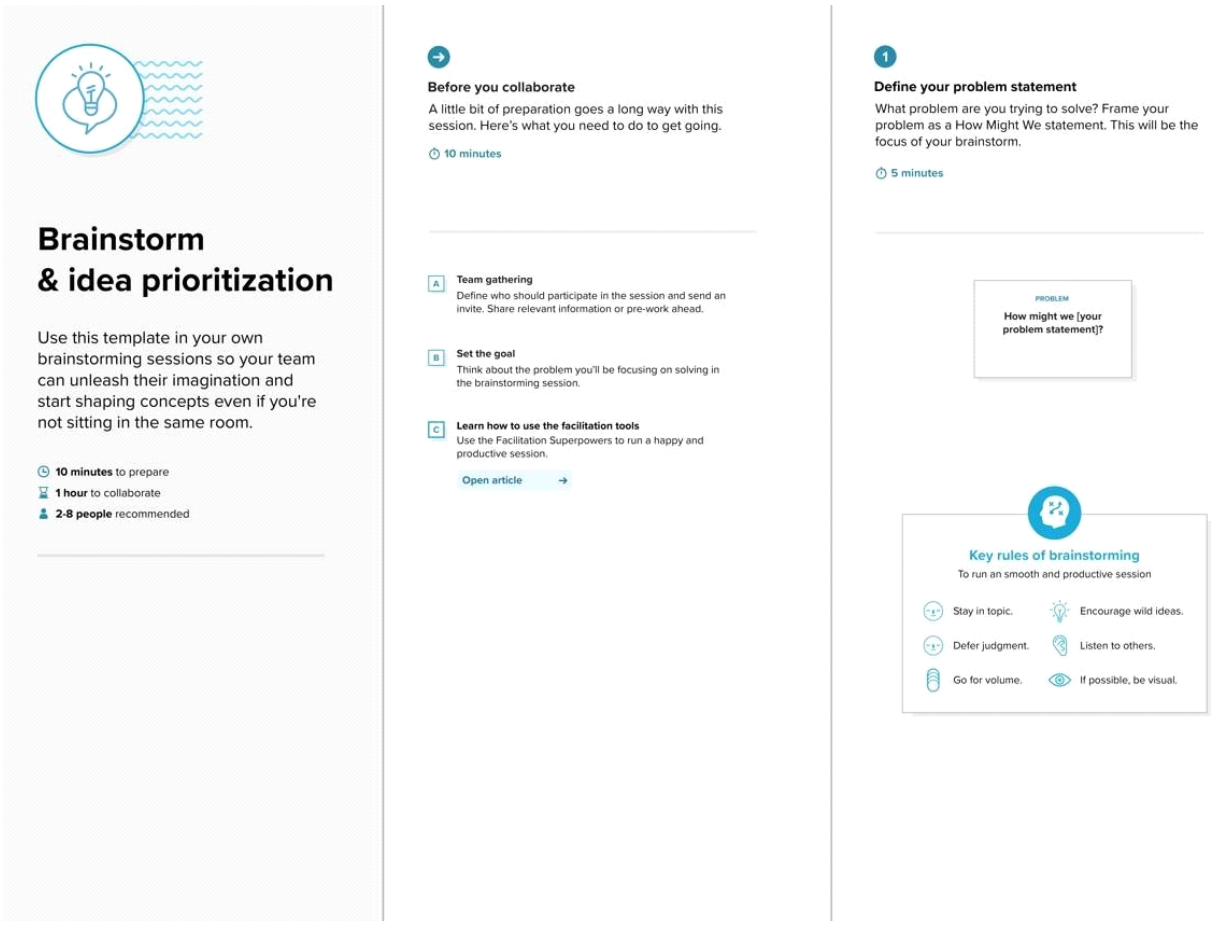
# 3. IDEATION AND PROPOSED SOLUTION

- We had splitted dataset based on decision tree diagram which I got from rattle and algorithms are applied. We calculated error rate then we found best algorithm based on low error rate.
- To get best accuracy,I had done one more step(i.e., hybridisation). The Hybrid machine learning model combine or integrate(ensemble learning) different classifier algorithms which gave highest accuracy. The combination of best algorithms we will get improved accuracy.

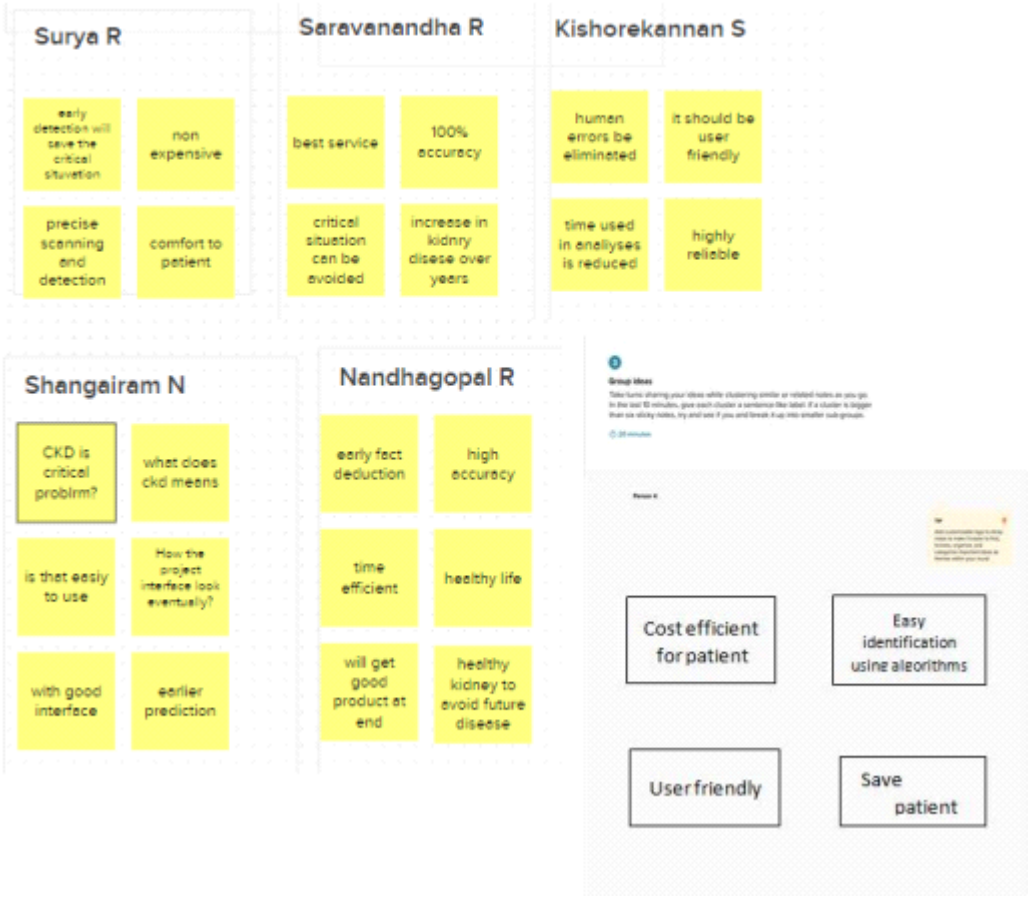
## 3.1 EMPATHY MAP CANVAS



## 3.2 IDEATION & BRAINSTORMING



## Step-2: Brainstorm, Idea Listing and Grouping

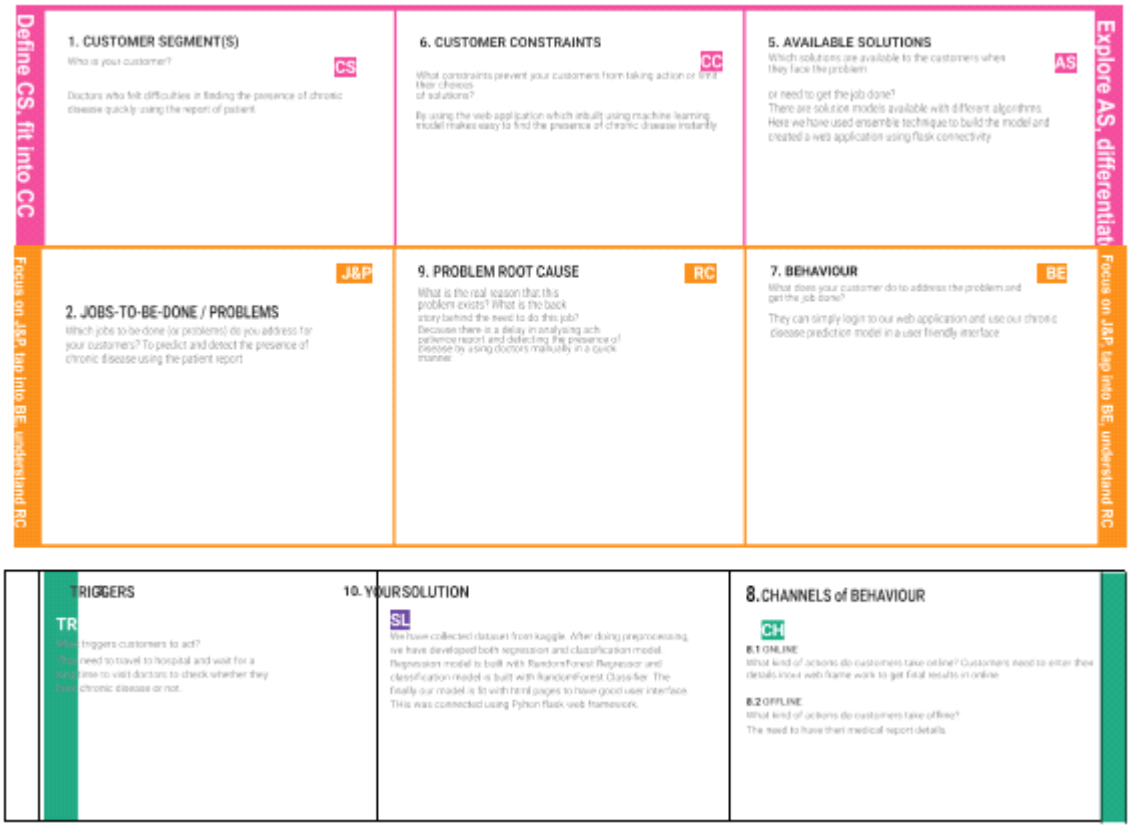


## Step-3: Idea Prioritization

**3.3 PROPOSED SOLUTION**

| S.NO. | PARAMETER | DESCRIPTION |
|---|---|---|
| 1. | Problem Statement (Problem to be solved) | Early detection of chronic Kidney diseases avert the normal function of the kidney. Early prediction of kidney disease using both classification and regression algorithms are an effective task that can help the doctors todiagnose the disease within a short duration of time. |
| 2. | Idea / Solution description | It is One of the Easiest solutions to predict the kidney disease using Machine Learning techniques. |
| 3. | Novelty / Uniqueness | For predicting the kidney disease. This project provides the best accuracy |
| 4. | Social Impact / Customer Satisfaction | It helps to identify the kidney disease in effective way and it is user friendly. |

## 3.4 PROBLEM SOLUTION FIT



# 4. REQUIREMENT ANALYSIS

## 4.1 FUNTIONAL REQUIREMENT

Functional requirements are product features or functions that developers must implement to enable users to accomplish their tasks.

| FR.NO | FUNCTIONAL REQUIREMENT(EPIC) | SUB-REQUIREMENT(STORY/SUB-TASK) |
|---|---|---|
| **FR-1** | Home page | ● Chronic Kidney disease description<br>● Information about Test Vitals required for prediction<br>● If new User, REGISTER<br>● If already exist, SIGN IN |
| **FR-2** | **User Registration** | ● Enters Mail ID and other personal details required for Registering. |
| **FR-3** | **User login** | ● Uses Mail ID and Password for login |
| **FR-4** | **Test Vitals** | ● Test Vitals should be entered for prediction |
| **FR-5** | **Result** | ● If Positive – Test Result along with the Information about what is to be done next will be displayed.<br>● If Negative – Test result along with preventive measures to prevent themselves from getting chronic kidney disease will be displayed.. |

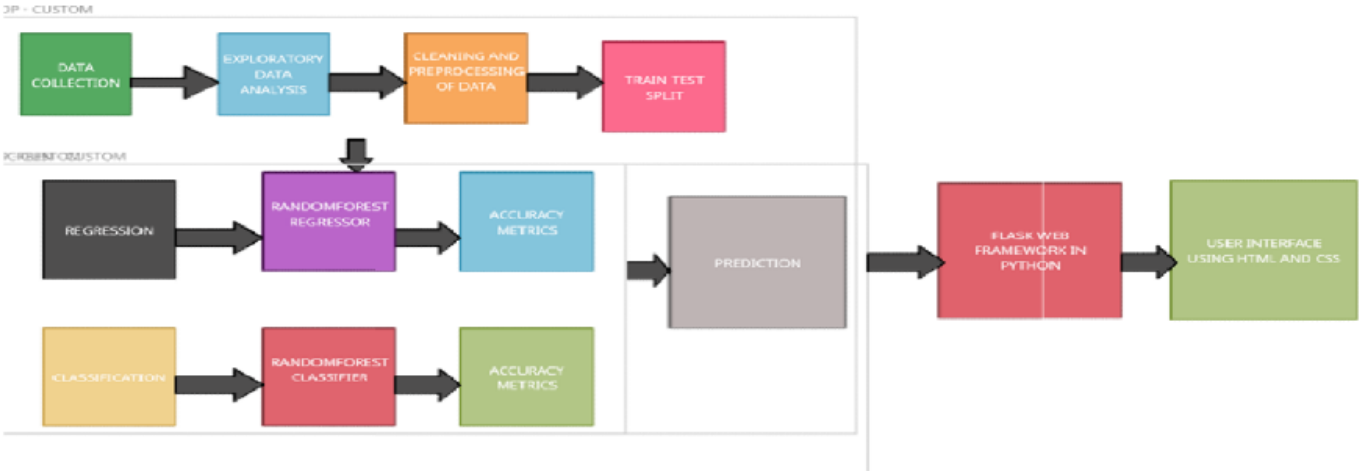
## 4.2 Non Functional Requirement

Nonfunctional Requirements (NFRs) define system attributes such as security, reliability, performance, maintainability, scalability, and usability.

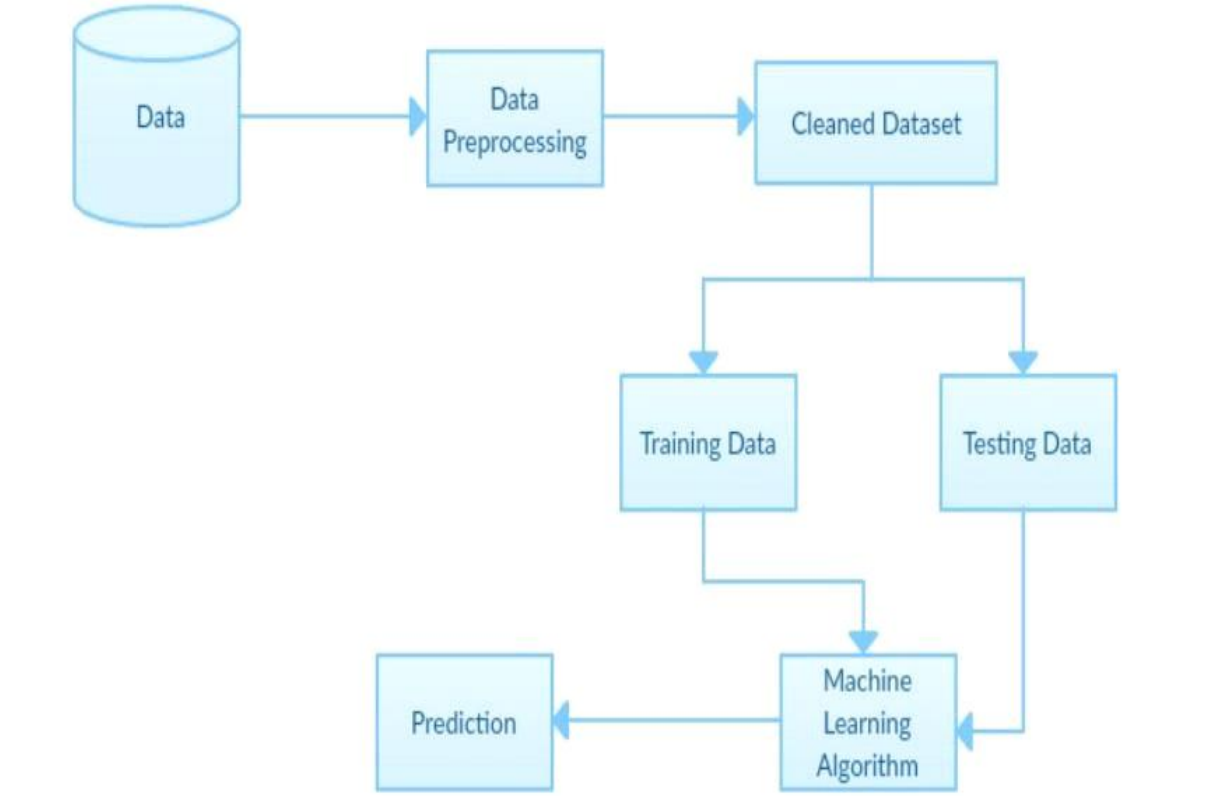
| FR.NO | NON-FUNTIONAL REQUIREMENT | DESCRIPTION |
|---|---|---|
| NFR-1 | USABILITY | Even Illiterates and people with no understanding of computer/mobile should be able to use the product. |
| NFR-2 | SECURIYY | Access permission for particular system information may be changed by systems data administration. |
| NFR-3 | RELIABILITY | The database update process must roll back all related updates when any updates fails. |
| NFR-4 | PERFORMANCE | The Home-page load time must be no more than 2 seconds for users that access the website using an LTE mobile connection. |
| NFR-5 | AVAILABILITY | New Model Deployment must not impact home page, test page and result page availability and must not take longer than one hour. |
| NFR-6 | SCALABILITY | It support 2000,000 users |

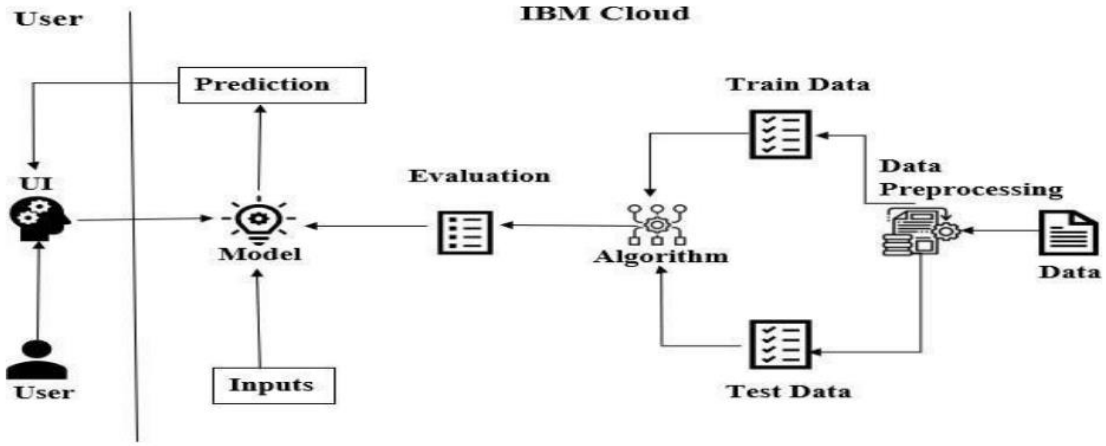
# 5. PROJECT DESIGN

## 5.1 DATA FLOW DIAGRAMS



## 5.2 SOLUTION & TECHNICAL ARCHITECTURE



## 5.3 USER STORIES

| User Type | Functional Requirement (Epic) | User Story Number | User Story / Task | Acceptance criteria | Priority | Release |
|---|---|---|---|---|---|---|
| Customer (Web-user) | Registration | USN-1 | As a user, I can register for the application by entering my email, password, and confirming my password. | I can access my account / dashboard | High | Sprint-1 |
| | Verification | USN-2 | As a user, I will receive confirmation email once I have registered for the application | I can receive confirmation email & click confirm | High | Sprint-1 |
| | Login | USN-3 | As a user, I can log into the application by entering email and password | Check whether password and email is correct | High | sprint-1 |

| | | | | | | |
|---|---|---|---|---|---|---|
| Customer (Web-user) | Dashboard | USN-4 | If the email id and password is correct, the user can log in to application otherwise it shows 'incorrect password or Id'. | View the dashboard of user who is login | High | Sprint-1 |
| Customer Care Executive | Help | USN-5 | If the user faces any issues, he/she an report it to our mail id. | Report option will be available in web app | High | Sprint-2 |
| Administrator | Verification | USN-6 | Administrator also has unique Id and password to login. He has additional users to organize the users of this web app | Check whether password and email is correct | High | Sprint-3 |

# 6.PROJECT PLANNING AND SCHEDULING

## 6.1 SPRINT PLANNING AND ESTIMATION

| Sprint | Functional Requirement (Epic) | User Story Number | User Story / Task | Story Points | Priority | Team Members |
|---|---|---|---|---|---|---|
| Sprint-1 | User Registration | USN-1 | As a user, I can register for the application by entering my name, mobile number, email, password, and confirming my password. | 10 | High | R.Nandhagopal<br>R.Surya<br>R.Saravana nandha<br>S.Kishorekannan<br>N.Shangairam |
| Sprint-2 | | USN-2 | As a user, I can register for the application through Gmail | 5 | Medium | R.Nandhagopal<br>R.Surya<br>R.Saravana nandha<br>S.Kishorekannan<br>N.Shangairam |
| Sprint-1 | User Confirmation | USN-3 | As a user, I will receive confirmation email once I have registered for the application | 10 | High | R.Nandhagopal<br>R.Surya<br>R.Saravana nandha<br>S.Kishorekannan<br>N.Shangairam |
| Sprint-2 | | USN-4 | As a user, I will receive confirmation OTP to verify the identity. | 5 | High | R.Nandhagopal<br>R.Surya<br>R.Saravana nandha<br>S.Kishorekannan<br>N.Shangairam |

| | | | | | | |
|---|---|---|---|---|---|---|
| Sprint-2 | User Confirmation | USN-5 | As a user, I will enter the input data for disease prediction in the form | 10 | High | R.Nandhagopal<br>R.Surya<br>R.Saravanananandha<br>S.Kishorekannan<br>N.Shangairam |
| Sprint-3 | Provide output to the user | USN-6 | As a user, I will get the result of disease prediction in the dashboard. | 10 | High | R.Nandhagopal<br>R.Surya<br>R.Saravanananandha<br>S.Kishorekannan<br>N.Shangairam |
| Sprint-3 | Data Analysis | USN-7 | As the admin, I will develop modules to pre-process and store the data. | 10 | High | R.Nandhagopal<br>R.Surya<br>R.Saravanananandha<br>S.Kishorekannan<br>N.Shangairam |
| Sprint-4 | Prediction of disease | USN-8 | As the admin, I will build a Machine Learning model to predict the disease | 10 | High | R.Nandhagopal<br>R.Surya<br>R.Saravanananandha<br>S.Kishorekannan<br>N.Shangairam |
| Sprint-4 | Final Delivery | USN-9 | Deploy the application in IBM cloud and make it available for use. | 10 | High | R.Nandhagopal<br>R.Surya<br>R.Saravanananandha<br>S.Kishorekannan<br>N.Shangairam |

**6.2 SPRINT DELIVERY SCHEDULE**

## Product Backlog,Sprint Schedule and Estimate

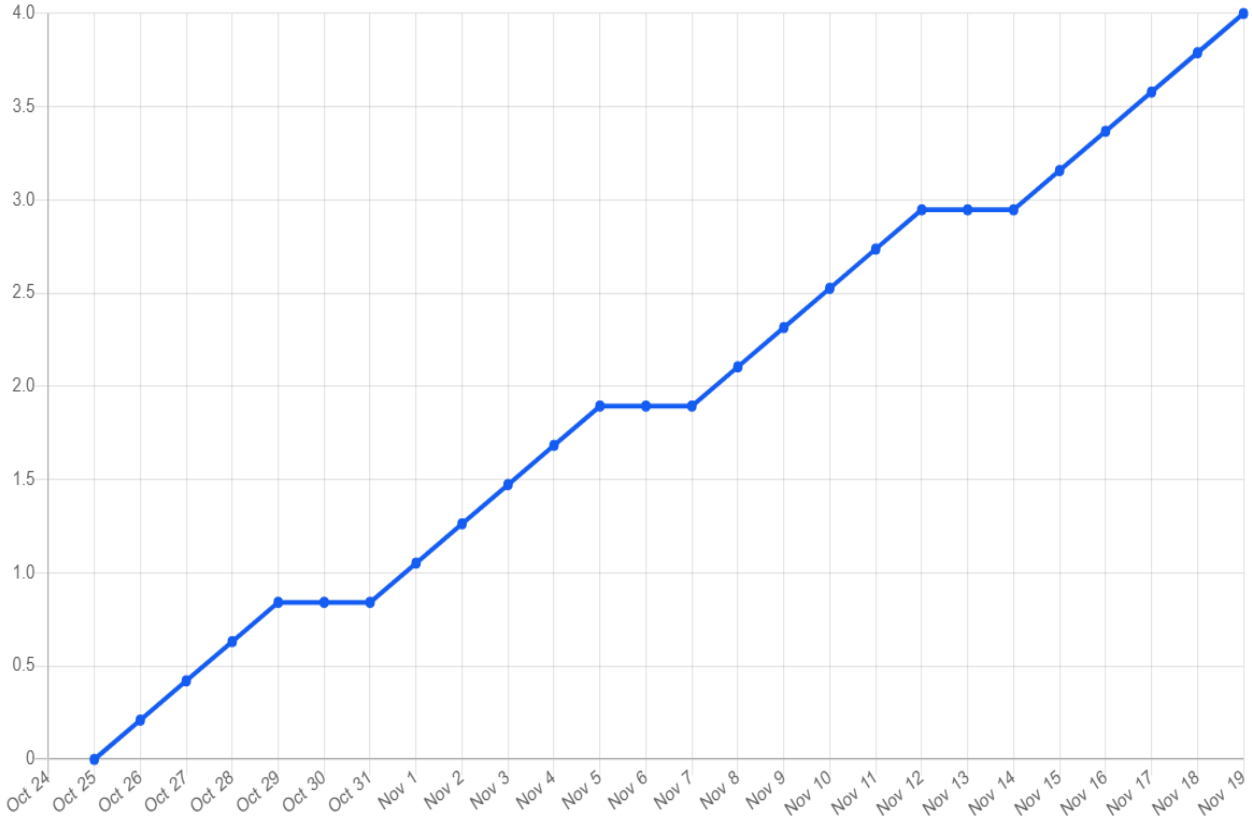
| Sprint | Total Story Points | Duration | Sprint Start Date | Sprint End Date (Planned) | Story Points Completed (as on Planned End Date) | Sprint Release Date (Actual) |
|---|---|---|---|---|---|---|
| Sprint-1 | 20 | 6 Days | 24 Oct 2022 | 29 Oct 2022 | 3 | 29 Oct 2022 |
| Sprint-2 | 20 | 6 Days | 31 Oct 2022 | 05 Nov 2022 | 2 | 05 Nov 2022 |
| Sprint-3 | 20 | 6 Days | 07 Oct 2022 | 12 Nov 2022 | 3 | 12 Nov 2022 |
| Sprint-4 | 20 | 6 Days | 14 Oct 2022 | 19 Nov 2022 | 3 | 19 Nov 2022 |

**Velocity:**

We have a 6-day sprint duration, and the velocity of the team is 20 (points per sprint). Theteam's average velocity (AV) per iteration unit (story points per day)

**AV = Sprint duration / velocity = 20 / 6 = 3.3**

## 6.3 REPORTS FROM JIRA




**Sprint 1** **:** Oct 24,2022 - Oct 29,2022

**Sprint 2** **:** Oct 31,2022 - Nov 05,2022

**Sprint 3** **:** Nov 07,2022 - Nov 12,2022

**Sprint 4** **:** Nov 14,2022 - Nov 19,2022

# 7. CODING AND SOLUTIONING

## 7.1 FLASK DEPLOYMENT

Using Flask we are lOcally deplOying our machine Learning mOdel.Flask acts as a web FramewOrk .Additionally we have app.py file tO lOcally deplOy the mOdel

### index.html

index - Notepad

File Edit Format View Help

```
<!--

                Online HTML Compiler.
            Code, Compile, Run and Debug HTML program online.
Write your code in this editor and press "Run" button to execute it.

-->

<!DOCTYPE html>
<html>

<head>
</head>

<body style="background:#419E94;background-image: url('C:/Users/elcot/Desktop/code/kidney.jpg'); background-position: center; background-repeat: no-repeat;">
   <div style="text-align:right;color:blue;font-Size:100%;background-Color:#B92F2D;">
      <a href ="C:/Users/elcot/Desktop/code/index.html" style="text-align:Left;color:green;font-Size:100%;background-Color:#CBE0E3"><strong>Home</strong></a>
      <a href ="C:/Users/elcot/Desktop/code/indexnew.html" style="text-align:Left;color:green;font-Size:100%;background-Color:#CBE0E3"><strong>Prediction</strong></a>
   </div>
<br></br>
<br></br>
   <div style="Vertical-align:middle">
      <h1 STYLE="text-align:center;color:white;font-family:Verdana;font-Size:Bold">
         CHRONIC KIDNEY DISEASE</h1>
      <h1 STYLE="text-align:center;color:white;font-family:Verdana;">PREDICTION</h1>
   </div>
</body>

</html>
```

# indexnew.html

indexnew - Notepad

File Edit Format View Help

```
<!--

                Online HTML Compiler.
            Code, Compile, Run and Debug HTML program online.
Write your code in this editor and press "Run" button to execute it.

-->

<!DOCTYPE html>
<html>

<head>
</head>

<body style="background:#419E94; ">
   <div style="background-Color:#B92F2D;">
   <div style="text-align:right;color:blue;font-Size:100%;padding-bottom:0px;margin-bottom:0px;background-Color:#B92F2D;border-color:#B92F2D;">
      <a href ="C:/Users/Lenovo/index.html" style="text-align:Left;color:green;font-Size:100%;background-Color:#CBE0E3"><strong>Home</strong></a>
   </div>
   <div style="text-align:center;color:blue;font-Size:200%;padding-top:0px;margin-top:0px;background-Color:#B92F2D;font-family:Snell Roundhand, cursive">
      <h1 style="text-align:center;color:white;font-Size:100%"><strong>Chronic Kidney Disease</strong></h1>
   </div>
   </div>
   <br></br>
   <fieldset  style ="background-image: url('D:/CKD/Image/Home.jpg');background-position: center; background-repeat: no-repeat;"><legend  ><b>Prediction Analysis</b></legen
   <div style="text-align:center; Vertical-align:center;Horizontal-align:center;">
<label for="Enter Your blood_urea">Enter Your blood_urea:</label>

<input type="text" id="PRED1" name="fname" >
   </div>
   <span style="margin-bottom: 0.4em;display:block"></span>
```

# result.html

result - Notepad

File Edit Format View Help

```
<!--

                Online HTML Compiler.
            Code, Compile, Run and Debug HTML program online.
Write your code in this editor and press "Run" button to execute it.

-->

<!DOCTYPE html>
<html>

<head>
</head>

<body style="background:#419E94; " onload="onloadfunction()">
   <div style="background-Color:#F37355;">
   <div style="text-align:right;color:blue;font-Size:100%;padding-bottom:0px;margin-bottom:0px;background-Color:#F37355;border-color:#F7ACFA;">
      <a href ="C:/Users/Lenovo/index.html" style="text-align:Left;color:blue;font-Size:100%;background-Color:#CBE0E3"><strong>Home</strong></a>
     <a href ="C:/Users/Lenovo/indexnew.html" style="text-align:Left;color:blue;font-Size:100%;background-Color:#CBE0E3"><strong>Return</strong></a>

  </div>
   <div style="text-align:center;color:blue;font-Size:200%;padding-top:0px;margin-top:0px;background-Color:#F37355;font-family:Snell Roundhand, cursive">
      <h1 style="text-align:center;color:white;font-Size:100%"><strong>Chronic Kidney Disease</strong></h1>
   </div>
   </div>
   <div style ="text-align:center">
      <h1 id="text1"></h1>
   </div>
   <div style="text-align:center">
      <img id="Image1" src="D:/CKD/Image/Failed.jpg" alt="Prediction" width="500" height="600" >
```

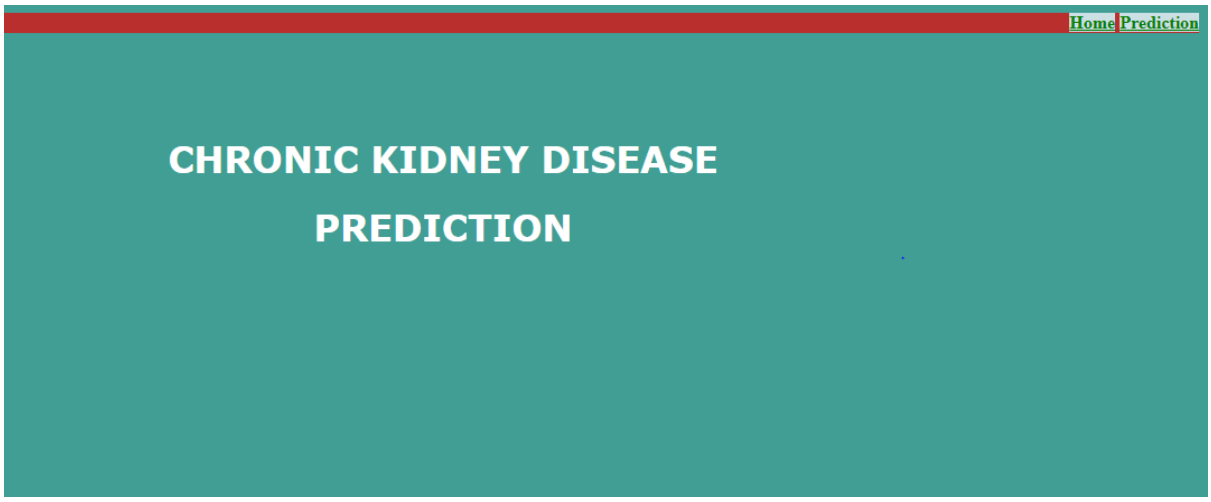
# App.py

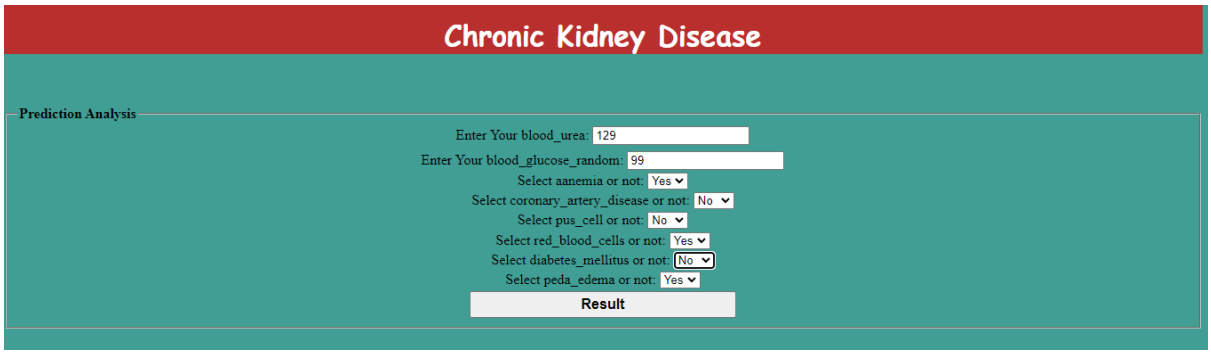
Flask > app.py

```python
from flask import Flask, render_template, request, send_from_directory


app = Flask(__name__)

import pickle
model = pickle.load(open(r'Model_one.pkl', 'rb'))

@app.route('/static/<path:path>')
def send_report(path):
    return send_from_directory('static', path)

@app.route('/')
def helloworld():
    return render_template("home.html")


@app.route('/login', methods = ['POST'])
def login():
    a = request.form["age"]
    b = request.form["bp"]
    c = request.form["sg"]
    d = request.form["alb"]
    e = request.form["sugar"]
    f = request.form["RBC"]
    g = request.form["bacteria"]
    h = request.form["bgr"]
    i = request.form["bu"]
    j = request.form["sc"]
    k = request.form["sodium"]
    l = request.form["haemo"]
    m = request.form["pcv"]
    n = request.form["rbc-count"]
    o = request.form["hypertension"]
    p = request.form["pe"]

    t = [[float(a), float(b), float(c), float(d), float(e), float(f), float(g), float(h), float(i), float(j), flo
    output = model.predict(t)
    print(output)

    return render_template("home.html", y = "The predicted result is: "  + str(output[0]))

@app.route('/admin')
def admin():
    return "Hey Admin How are you?"

app.run(host='localhost', port=5000)
if __name__ == '__main__' :
    app.run(debug = True)
```

## 8.TESTING

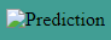
## 8.1 TEST CASES



# Chronic Kidney Disease

## Prediction: You Don't have Chronic Kidney Disease



## 8.2 USER ACCEPTANCE TESTING

---

### 1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [Early Detection of Chronic Kidney Disease] project at the time of the release to User Acceptance Testing (UAT).

### 2. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

| Resolution | Severity 1 | Severity 2 | Severity 3 | Severity 4 | Subtotal |
|---|---|---|---|---|---|
| By Design | 3 | 1 | 1 | 1 | 6 |
| Duplicate | 4 | 0 | 2 | 0 | 6 |
| External | 2 | 2 | 0 | 1 | 5 |
| Fixed | 1 | 1 | 1 | 1 | 4 |
| Not Reproduced | 0 | 0 | 0 | 0 | 0 |
| Skipped | 0 | 0 | 0 | 0 | 0 |
| Won't Fix | 0 | 0 | 0 | 0 | 0 |
| Totals | 10 | 4 | 4 | 3 | 21 |

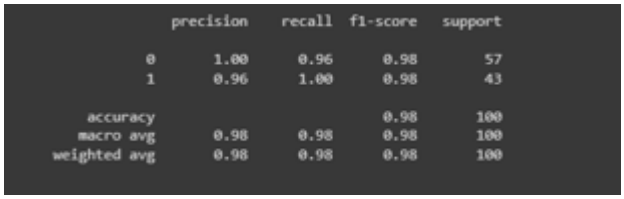
### 3. Test Case Analysis

This report shows the number of test cases that have passed, failed, and untested

| Section | Total Cases | Not Tested | Fail | Pass |
|---|---|---|---|---|
| Home Screen | 1 | 0 | 0 | 1 |
| User Input | 3 | 0 | 0 | 3 |
| Chronic Kidney Disease testing | 2 | 0 | 0 | 2 |
| No Chronic Kidney Disease testing | 2 | 0 | 0 | 2 |
| Version Control | 2 | 0 | 0 | 2 |

## 9.RESULTS

### 9.1 PERFORMANCE METRICS



| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 1.00 | 0.96 | 0.98 | 57 |
| 1 | 0.96 | 1.00 | 0.98 | 43 |
| accuracy | | | 0.98 | 100 |
| macro avg | 0.98 | 0.98 | 0.98 | 100 |
| weighted avg | 0.98 | 0.98 | 0.98 | 100 |

# 10.ADVANTAGES AND DISADVANNTAGES

### ADVANTAGES

Predictive modeling is a statistical technique using machine learning and data mining to predict and forecast likely future outcomes with the aid of historical and existing data. It works by analyzing current and historical data and projecting what it learns on a model generated to forecast likely outcomes.We found that machine learning can predict the occurrence of individual chronic diseases, progression, and their determinants and in many contexts. The findings are original and relevant to improve clinical decisions and the organization of health c**are facilities**

### DISADVANTAGES

In Chronic Disease prediction, for classification problem we get a very good accuracybut for regression, we get considerable error rate. So, we need to add some more data or change the machine algorithm or by using deep learning techniques for reducing the error in predicting the probability values

## 11.CONCLUSION

This paper deals with the prediction of CKD in people. Out of the 24 attributes present 12 best attributes are taken for prediction. Prediction is done using the machine learning technique**.** The main objective of this study was to predict patients with CKD using less number attributes while maintaining a higher accuracy. Here we obtain an accuracy of about  percentage.

## 12.FUTURE SCOPE

Diseases related to kidney is becoming more and more common with time. With continuous technological advancements, these are only going to increase in the future. Although people are becoming more conscious of health nowadays and are joining yoga classes, dance classes; still the sedentary lifestyle and luxuries that are continuously being introduced and enhanced; the problem is going to last long. So, in such a scenario, our projectwill be extremely helpful to the society. With the dataset that we used for this project, we got 89% accuracy for Random forest model, and though it might be difficult to get such accuracies with very large datasets, from this projects results, one can clearly conclude that we can predict the risk of chronic diseases with accuracy of 95 % or more. Also it can be incorporated into a wide range commercial website and these app and website will be highly beneficial for a large section of society

## 13.APPENDIX

## SOURCE CODE

# IMPORTING LIBRARIES

```
import pandas as pd

import numpy as np

from collections import Counter as c

import matplotlib.pyplot as plt

import seaborn as sns

import missingno as msno

from sklearn.metrics import accuracy_score

from sklearn.metrics import confusion_matrix

from sklearn.model_selection import train_test_split

from sklearn.preprocessing import LabelEncoder

from sklearn.linear_model import LogisticRegression

import pickle
```

## LOADING THE DATASET

```
data=pd.read_csv(r'D:\IBM\CKD.csv')

data.head()

data.tail()

data.head(10)
```

## DROP ID COLUMN

```
data.drop('id', axis = 1, inplace = True)
```

## RENAMING THE COLUMNS

data.columns = ['age', 'blood_pressure', 'specific_gravity', 'albumin', 'sugar', 'red_blood_cells', 'pus_cell', 'pus_cell_clumps', 'bacteria', 'blood_glucose_random', 'blood_urea', 'serum_creatinine', 'sodium',potassium', 'haemoglobin', 'packed_cell_volume', 'white_blood_cell_count', 'red_blood_cell_count', 'hypertension', 'diabetes_mellitus', 'coronary_artery_disease', 'appetite', 'peda_edema', 'aanemia', 'class']

data.columns

## UNDERSTANDING DATA TYPE AND SUMMARY OF FEATURES

data['class'].unique()

data['class']=data['class'].replace("ckd\t", "ckd")

data['class'].unique()

catcols = [col for col in data.columns if data[col].dtype == 'object']

numcols = [col for col in data.columns if data[col].dtype != 'object']

print(catcols)

## REMOVING THE COLUMNS WHICH ARE NOT CATOGORICAL

catcols.remove('red_blood_cell_count')

catcols.remove('packed_cell_volume')

catcols.remove('white_blood_cell_count')

print(catcols)

contcols = [col for col in data.columns if data[col].dtype != 'object']

print(contcols)

## REMOVING THE COLUMNS WHICH ARE NOT NUMERICAL

contcols.remove('specific_gravity')

contcols.remove('albumin')

contcols.remove('sugar')

print(contcols)

## ADDING COLUMNS WHICH WE FOUND CONTINUOUS

contcols.append('red_blood_cell_count')

contcols.append('packed_cell_volume')

contcols.append('white_blood_cell_count')

print(contcols)

## ADDING COLUMNS WHICH WE FOUND CATEGRICAL

catcols.append('specific_gravity')

catcols.append('albumin')

catcols.append('sugar')

print(catcols)

## RECTIFYING THE CATEGORICAL COLUMNS CLASSES

data['coronary_artery_disease']=data.coronary_artery_disease.replace('\tno','no')

c(data['coronary_artery_disease'])

data['diabetes_mellitus']=data.diabetes_mellitus.replace('\tno','no')

data['diabetes_mellitus']=data.diabetes_mellitus.replace(' yes','yes')

```
data['diabetes_mellitus']=data.diabetes_mellitus.replace('\tyes','yes')

c(data['diabetes_mellitus'])
```

## HANDLING THE MISSING VALUES

```
data.isnull().any()

data.isnull().sum()

data.packed_cell_volume = pd.to_numeric(data.packed_cell_volume, errors='coerce')

data.white_blood_cell_count = pd.to_numeric(data.white_blood_cell_count, errors='coerce')

data.red_blood_cell_count = pd.to_numeric(data.red_blood_cell_count, errors='coerce')
```

## REPLACING THE MISSING VALUES

```
data['blood_glucose_random'].fillna(data['blood_glucose_random'].mean(),inplace=True)

data['blood_pressure'].fillna(data['blood_pressure'].mean(),inplace=True)

data['blood_urea'].fillna(data['blood_urea'].mean(),inplace=True)

data['haemoglobin'].fillna(data['haemoglobin'].mean(),inplace=True)

data['packed_cell_volume'].fillna(data['packed_cell_volume'].mean(),inplace=True)

data['potassium'].fillna(data['potassium'].mean(),inplace=True)

data['red_blood_cell_count'].fillna(data['red_blood_cell_count'].mean(),inplace=True)

data['serum_creatinine'].fillna(data['serum_creatinine'].mean(),inplace=True)

data['sodium'].fillna(data['sodium'].mean(),inplace=True)

data['white_blood_cell_count'].fillna(data['white_blood_cell_count'].mean(),inplace=True)

data['age'].fillna(data['age'].mode()[0],inplace=True)

data['hypertension'].fillna(data['hypertension'].mode()[0],inplace=True)
```

```
data['pus_cell_clumps'].fillna(data['pus_cell_clumps'].mode()[0],inplace=True)

data['appetite'].fillna(data['appetite'].mode()[0],inplace=True)

data['albumin'].fillna(data['albumin'].mode()[0],inplace=True)

data['pus_cell'].fillna(data['pus_cell'].mode()[0],inplace=True)

data['red_blood_cells'].fillna(data['red_blood_cells'].mode()[0],inplace=True)

data['coronary_artery_disease'].fillna(data['coronary_artery_disease'].mode()[0],inplace=True)

data['bacteria'].fillna(data['bacteria'].mode()[0],inplace=True)

data['aanemia'].fillna(data['aanemia'].mode()[0],inplace=True)

data['sugar'].fillna(data['sugar'].mode()[0],inplace=True)

data['diabetes_mellitus'].fillna(data['diabetes_mellitus'].mode()[0],inplace=True)

data['peda_edema'].fillna(data['peda_edema'].mode()[0],inplace=True)

data['specific_gravity'].fillna(data['specific_gravity'].mode()[0],inplace=True)
```

## LABEL ENCODING

```
from sklearn.preprocessing import LabelEncoder

for i in catcols:

    print("LABEL ENCODING OF:",i)

    LEi = LabelEncoder()

    print(c(data[i]))

    data[i] = LEi.fit_transform(data[i])

    print(c(data[i]))

    print("*"*100)
```

## SPLITTING THE DATASET INTO DEPENDENT AND INDEPENDENT VARIABLE

```
selcols=['red_blood_cells', 'pus_cell','blood_glucose_random', 'blood_urea','peda_edema', 'aanemia','diabetes_mellitus', 'coronary_artery_disease']

x=pd.DataFrame(data,columns=selcols)

y=pd.DataFrame(data,columns=['class'])

print(x.shape)

print(y.shape)
```

## SPLIT THE DATASET INTO TRAIN SET AND TEST SET

```
from sklearn.model_selection import train_test_split

x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=2)

print(x_train.shape)

print(y_train.shape)

print(x_test.shape)

print(y_test.shape)
```

## BUILDING A MACHINE LEARNING MODEL

```
from sklearn.linear_model import LogisticRegression

lgr = LogisticRegression()

lgr.fit(x_train,y_train)
```

## TEST THE MODEL

```
y_pred = lgr.predict(x_test)

y_pred = lgr.predict([[129,99,1,0,0,1,0,1]])

print(y_pred)

c(y_pred)
```

## MODEL EVALUATION

```
y_pred = lgr.predict(x_test)

accuracy_score(y_pred,y_test)

confusion_matrix(y_test,y_pred)
```

## SAVE THE MODEL

```
pickle.dump(lgr,open('CKD.pkl','wb'))
```

## GITHUB AND PROJECT DEMOLINK

**GITHUB LINK:** https://github.com/IBM-EPBL/IBM-Project-21438-1659780356

**DEMO LINK:** https://drive.google.com/drive/folders/1MYVJAzJ0YYmpOjbLhZtlZZLS99Izv-RR?usp=share_link