EARLY DETECTION OF CHRONIC KIDNEY DISEASE USING MACHINE LEARNING

A Project report submitted in partial fulfilment of 7th semester in degree

for

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

for

HX8001 PROFESSIONAL READYINESS FOR INNOVATION EMPLOYABLITY AND ENTREPRENEURSHIP

Submitted by

Team ID: PNT2022TMID49895

T.SELVARAJ	950619104060
J.SRIDHAR	950619104063
S.MATHANSUNDAR	950619104040
A.VELMURUGAN	950619104070

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

EINSTEIN COLLEGE OF ENGINEEING,

TIRUNELVELI-627 012

BONAFIDE CERTIFICATE

Certified this Report ”**EARLY DETECTION OF CHRONIC KIDNEY DISEASE”**, for the project work of **Mr. T.SELVARAJ** (**950619104060**), **Mr. J.SRIDHAR** (**950619104063**), **Mr. S.MATHAN SUNDAR** (**950619104040**), and **Mr. A.VELMURUGAN** (**950619104070**) who carried out the project work under my supervision. Certified further that to the best of my knowledge the work reported here in does not form part of any other thesis or dissertation on the basis of which a degree or award was co-offered on the earlier occasion on this or any other candidate.

SIGNATURE
Dr. T.SURESH THANKRISHNAN
HEAD OF THE DEPARTMENT
COMPUTER SCIENCE AND ENGINEERING
EINSTEIN COLLEGE OF ENGINEEING
TIRUNELVELI-12

SIGNATURE
K.ANTONY SUDHA
MENTOR
COMPUTER SCIENCE AND ENGINEERING
EINSTEIN COLLEGE OF ENGINEERING
TIRUNELVELI-12

ACKNOWLEDGEMENT

We have successfully completed the project with blessings showered onus by god, the almighty, A project of this nature needs co-operation and support from many for successful completion.

We express our heartfelt thanks to **Mr. A.MATHIVANAN, BE., M.Sc.,(Agri),** Managing Trustee of Einstein college of Engineering, Tirunelveli, for his mortal support and device.

Our thanks to **Prof. A.AMUTHAVANAN, BE., M.S (USA).B.L,** Chairman of our collegefor making necessary arrangements to do this project.

Our heartly thanks to **Prof. EZHILVANAN, MBA.,** Secretary of our college for making necessary arrangements to do this project.

We wish to express our gratitude to **Dr. VELAYUTHAM, M.E, Ph.D., FIE.** Principal forthe support he provided us to carry out this project successfully.

We are very much thankful to **Dr. T.SURESH THANKRISHNAN**, Head of the Department , Electronics and communication Engineering who is always a constant of inspiring us.

We are extend our sincere thanks to all teaching and non-teaching staff members and our family members, friends for their help in completing this project.

ABSTRACT

Every year, an increasing number of patients are diagnosed with late stages of renal disease. Chronic Kidney Disease, also known as Chronic Renal Disease, is characterized by abnormal kidney function or a breakdown of renal function that progresses over months or years. Chronic kidney disease is often found during screening of persons who are known to be at risk for kidney issues, such as those with high blood pressure or diabetes, and those with a blood family who has chronic kidney disease (CKD). As a result, early prognosis is critical in battling the disease and providing effective therapy. Only early identification and continuous monitoring can avoid serious kidney damage or renal failure. Machine Learning (ML) plays a significant part in the healthcare system, and it may efficiently aid and help with decision support in medical institutions. The primary goals of this research are to design and suggest a machine learning method for predicting CKD. Support Vector Machine (SVR), Random Forest (LR), Artificial Neural Network (ANN), and Decision Tree are four master teaching methodologies investigated (DT). The components are built using chronic kidney disease datasets, and the outcomes of these models are compared to select the optimal model for prediction. Keywords: Chronic Kidney Disease (CKD), Machine Learning (ML), Support Vector Machine (SVR), Random Forest (LR), Artificial Neural Network (ANN), Decision Tree (DT).

TABLE OF CONTENT

CHAPTER 1

INTRODUCTION:

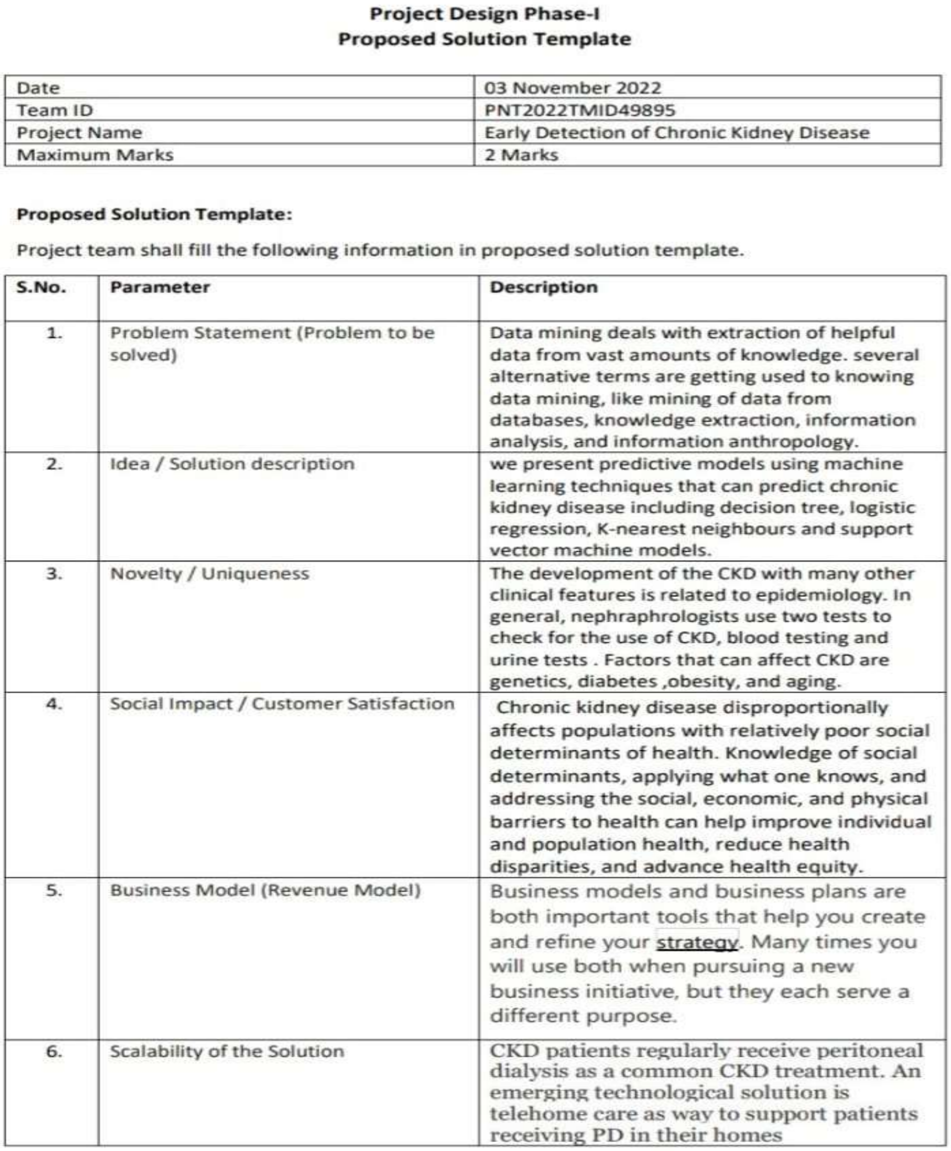
Chronic kidney disease prediction is one of the most important issues in healthcare-analytics. The most interesting and challenging tasks in day-to-day lives as one third of the adult population is affected by chronic kidney disease (CKD), and millions die each year because they do not have access to affordable treatment. Chronic Kidney Disease can be cured, if treated in the early stages. The main aim of the project is to predict whether the patient have chronic kidney disease or not in a painless, accurate and faster way based on certain diagnostic measurement like Blood Pressure(8P), Albumin(Al) etc , and then appropriate treatment can be given based on the details provided by the model.

1.1 Project Overview:

This Project aims at creating a model for early detection of Chronic Kidney Disease using Machine Learning technology. The model output is integrated with the Flask framework The front end developed in html is used to receive user input on various parameters needed to decide on the early detection of kidney disease The same model is deployed into BM cloud using API keys and scoring endpoints.

Skills Required:

Python Web Frameworks Python For data Visualization Data Preprocessing Techniques Machine Learning BM Cloud IBM Watson Studio Python-Flask.



Project Design Phase-I
Proposed Solution Template

Date	03 November 2022
Team ID	PNT2022TMID49895
Project Name	Early Detection of Chronic Kidney Disease
Maximum Marks	2 Marks

Proposed Solution Template:

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Data mining deals with extraction of helpful data from vast amounts of knowledge. several alternative terms are getting used to knowing data mining, like mining of data from databases, knowledge extraction, information analysis, and information anthropology.
2.	Idea / Solution description	we present predictive models using machine learning techniques that can predict chronic kidney disease including decision tree, logistic regression, K-nearest neighbours and support vector machine models.
3.	Novelty / Uniqueness	The development of the CKD with many other clinical features is related to epidemiology. In general, nephraphrologists use two tests to check for the use of CKD, blood testing and urine tests . Factors that can affect CKD are genetics, diabetes ,obesity, and aging.
4.	Social Impact / Customer Satisfaction	Chronic kidney disease disproportionally affects populations with relatively poor social determinants of health. Knowledge of social determinants, applying what one knows, and addressing the social, economic, and physical barriers to health can help improve individual and population health, reduce health disparities, and advance health equity.
5.	Business Model (Revenue Model)	Business models and business plans are both important tools that help you create and refine your strategy. Many times you will use both when pursuing a new business initiative, but they each serve a different purpose.
6.	Scalability of the Solution	CKD patients regularly receive peritoneal dialysis as a common CKD treatment. An emerging technological solution is telehome care as way to support patients receiving PD in their homes

1.2 Purpose:

The purpose of the project in to alert doctors for an early detection of kidney disease and hence ensure speedy recovery or prevention of kidney disease

CHACPTER 2

LITERATURE SURVEY

First we have done literature surveys of various IEEE papers to arrive at the idea of the project development. It is given below:

2.1 Existing Problem:

Our modern lifestyle and environmental pollution has increased kidney disease. Presently kidney disease is detected at late stages in many countries leading to loss of precious lives. There are very few means to identify them at an early stage.

2.2 Reference:

LITERATURE SURVEY: REVIEW 1:

TITLE OF THE PAPER:

Prediction of Chronic Kidney Disease using Adaptive Hybridized Deep Convolutional neural Network on the Internet of Medical Thing Platform

PROBLEM DESCRIPTION:

Chronic Kidney disease is a severe lifelong condition caused either by renal disease or by impaired functions of the kidneys. In the present area of research, Kidney cancer is one of the deadliest and crucial importance for the survival of the patients diagnosis and classification. Early diagnosis and proper therapy can stop or delay the development of this chronic disease into the final stage where dialysis or re transplantation is the only way of saving the life of the patient. The development of automated tools to accurately identify subtypes of kidney cancer is therefore, an urgent challenge in the recent past. In this paper, to examine the ability of various deep learning methods an Adaptive hybridized Deep Convolutional Neural Network (AHDCNN) has been proposed for the early detection of Kidney disease efficiently and effectively. Classification technology efficiency depends on the role of the data set. To enhance the accuracy of the classification system by reducing the feature dimension an algorithm model has been developed using CNN. These high-level properties help to buld a supervised tissue classifier that discriminates between the two types of tissue. The experimental process on the Internet of medical things platform concludes, with the aid of predictive analytics, that advances in machine learning which provides a promising framework for the recognition of intelligent solutions to prove their predictive capability beyond the field of kidney disease.

REVIEW-2:

TITLE OF THE PAPER:

Clinical Decision Support System for Diagnosis and Management of Chronic Renal Failure.

PROBLEM DESCRIPTION:

Chronic Renal Failure (CRF) is a gradual loss of kidney's function over a period of time, ranging from months to years. Unlike other chronic diseases, CRF is not yet thoroughly explored in literature. In this paper, we propose a new clinical decision support system for diagnosing patients with CRF. Several data classification algorithms including Artificial Neural Networks (ANNs), Naïve Bayes and Decision Tree are developed and implemented to diagnose patients with CRF and determine the progression stage of the disease. A clinical dataset of 102 instances is collected from patients records and used in this study Performance of the developed CRF diagnosis system is assessed in terms of diagnosis accuracy, sensitivity, and specificity and in evaluated by specialist physicians. Furthermore, the open source Weka software is also used in this study for performance comparison and evaluation purposes. The obtained results showed that the developed decision tree algorithm is the most accurate CRF classifier (12.2%) when compared to all other algorithms/implementations involved in this study.

REVIEW -3:

TITLE OF THE PAPER:

Chronic Kidney Disease Analysis Using Data Mining Techniques

PROBLEM DESCRIPTION:

Data mining has been a current trend for attaining diagnostic results. Huge amount of unmined data is collected by the healthcare industry in order to discover hidden information for effective diagnosis and decision making. Data mining is the process of extracting hidden information from massive dataset. categorizing valid and unique patterns in data. There are many data mining techniques like clustering, classification, association analysis, regression etc. The objective of our paper is to predict Chronic Kidney Disease(CKD) using classification techniques like Naive Bayes and Artificial Neural Network(ANN) The experimental results implemented in Rapidminer tool show that Naive Bayes produce more accurate results than Artificial Neural Network.

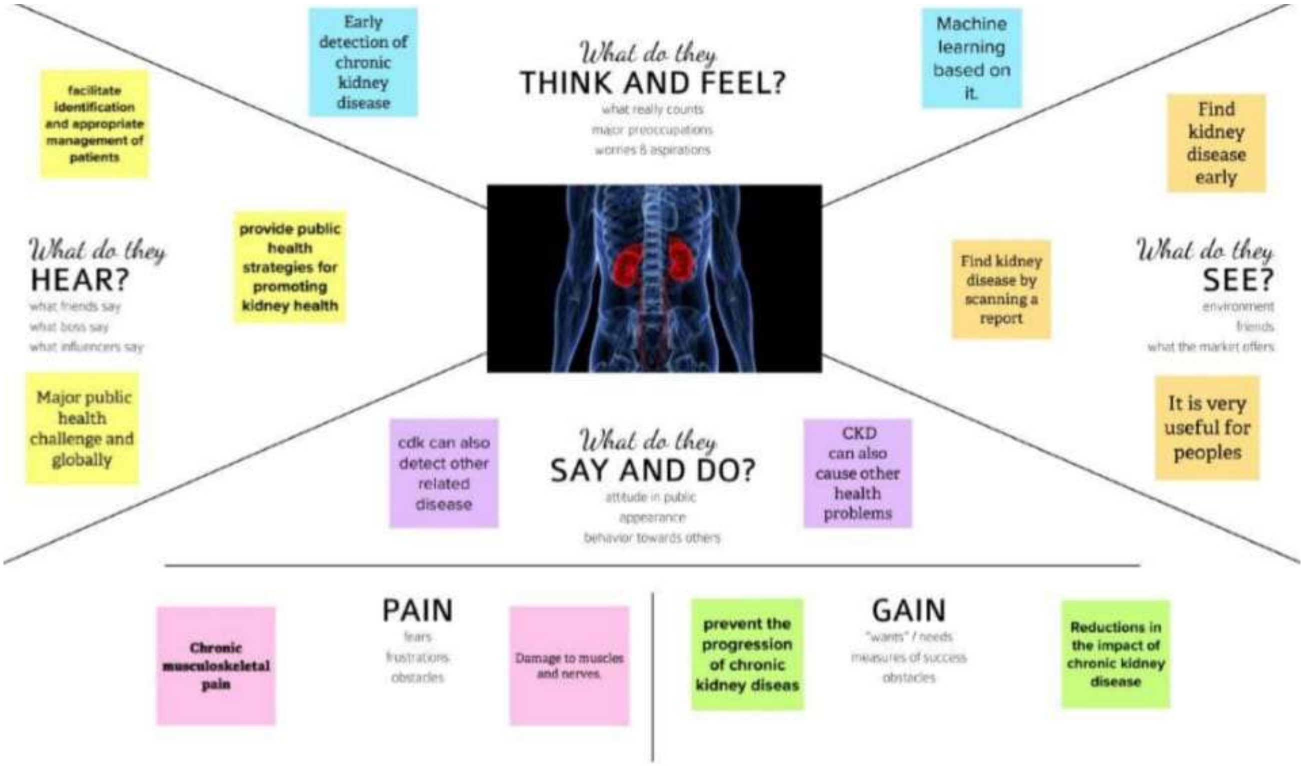
CHAPTER 3

IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

An empathy map is a collaborative visualization used to articulate what we know about a particular type of user. It externalizes knowledge about users in order to 1) create a shared understanding of user needs, and 2aid in decision making.

An empathy map helps to map what a design team knows about the potential audience This tool helps to understand the reason behind some actions a user takes deeply. This tool helps build Empathy towards users and helps design teams shift focus from the product to the users who are going to use the product.



What do they THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations

Early detection of chronic kidney disease

Machine learning based on it.

What do they HEAR?
what friends say
what boss say
what influencers say

facilitate identification and appropriate management of patients

provide public health strategies for promoting kidney health

Major public health challenge and globally

What do they SEE?
environment
friends
what the market offers

Find kidney disease early

Find kidney disease by scanning a report

It is very useful for peoples

What do they SAY AND DO?
attitude in public
appearance
behavior towards others

cdk can also detect other related disease

CKD can also cause other health problems

PAIN
fears
frustrations
obstacles

Chronic musculoskeletal pain

Damage to muscles and nerves

GAIN
"wants" / needs
measures of success
obstacles

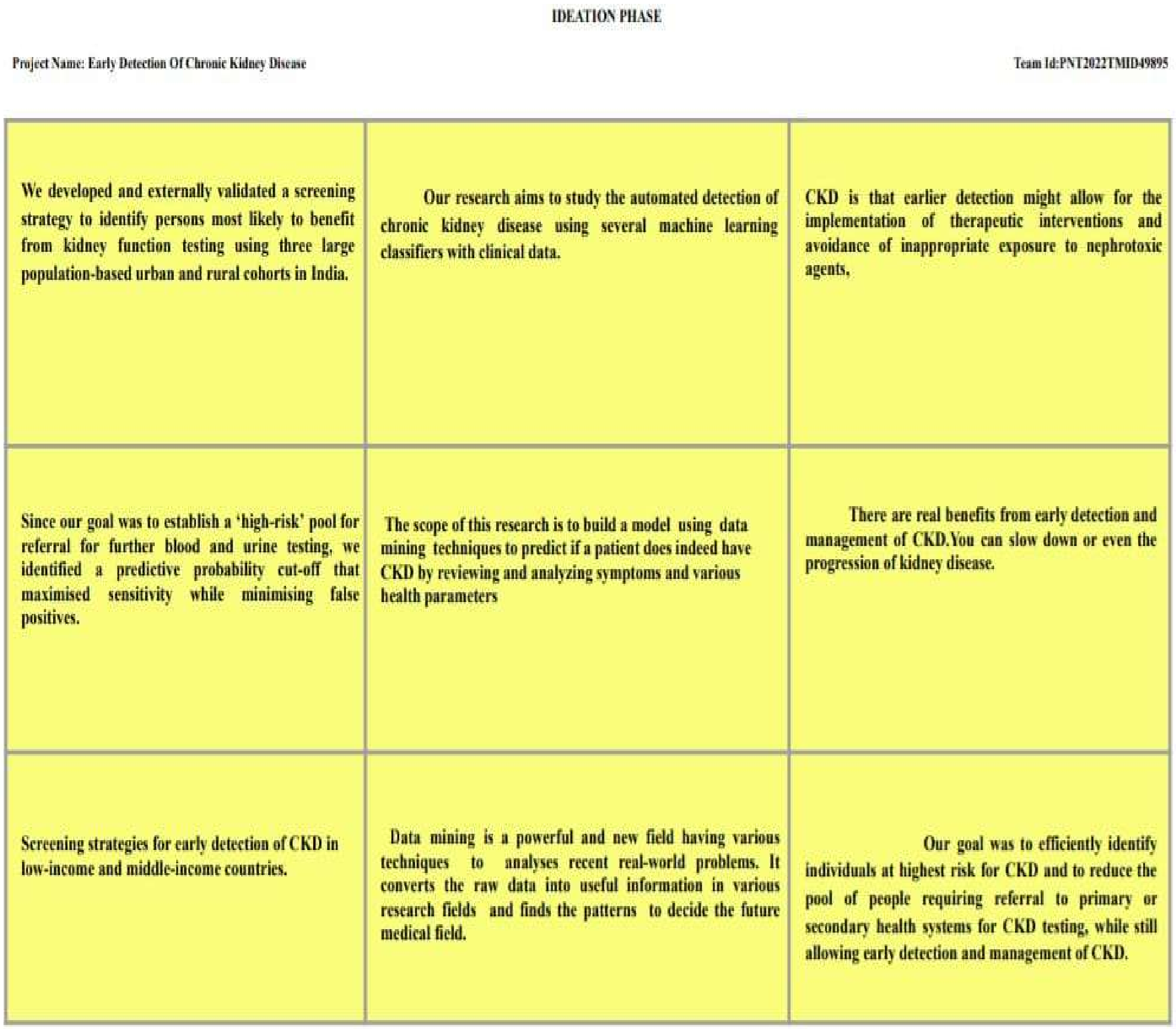
prevent the progression of chronic kidney diseas

Reductions in the impact of chronic kidney disease

3.2 IDEATION AND BRAINSTORMING

Brainstorming is an activity that will help you generate more innovative ideas. It's one of many methods of ideation the process of coming up with new ideas and it's core to the design thinking process.

Brainstorming refers to a problem-solving technique used by teams or individuals. In this process, participants generate various ideas or solutions, then begin discussing and narrowing them down to the best options



IDEATION PHASE

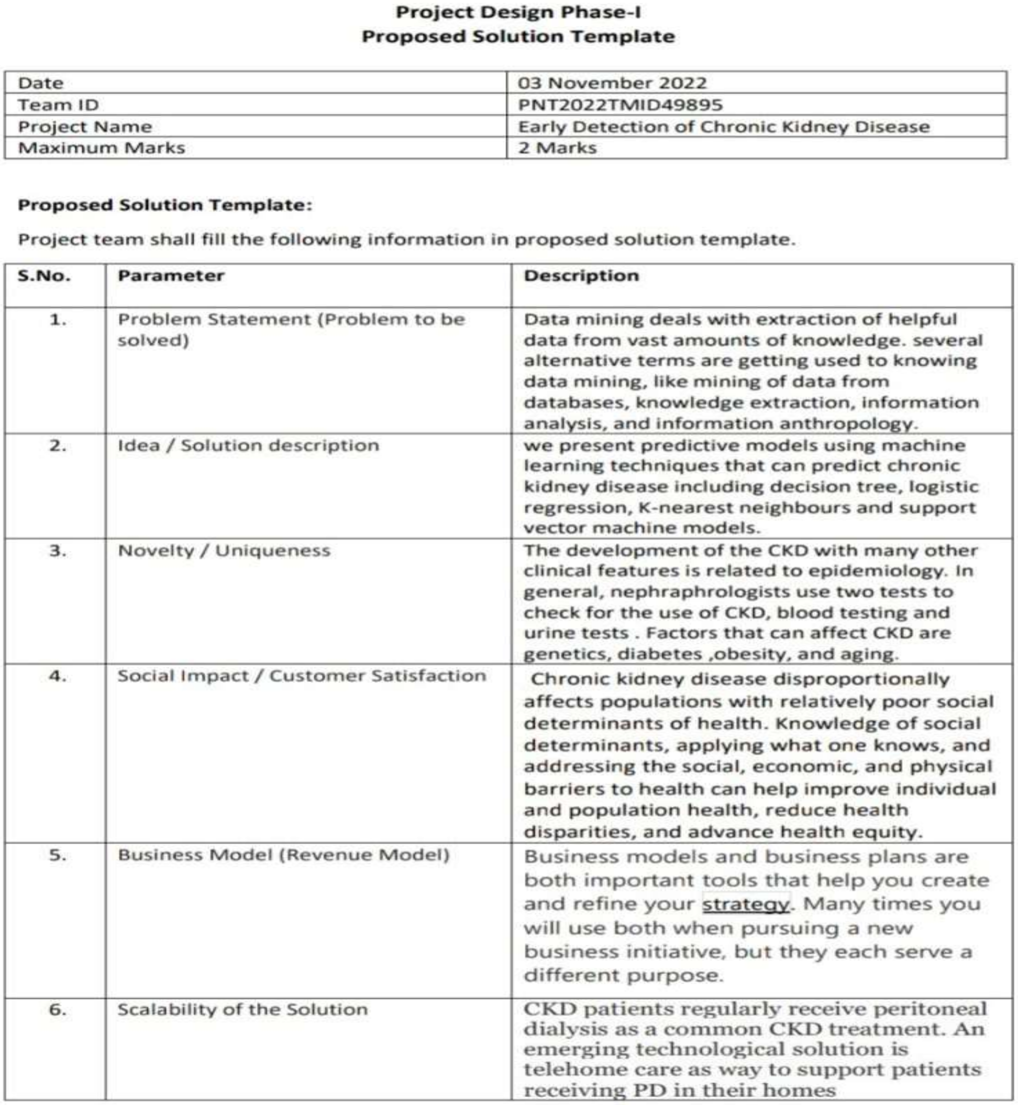
Project Name: Early Detection Of Chronic Kidney Disease

Team Id:PNT2022TMID49895

We developed and externally validated a screening strategy to identify persons most likely to benefit from kidney function testing using three large population-based urban and rural cohorts in India.	Our research aims to study the automated detection of chronic kidney disease using several machine learning classifiers with clinical data.	CKD is that earlier detection might allow for the implementation of therapeutic interventions and avoidance of inappropriate exposure to nephrotoxic agents,
Since our goal was to establish a 'high-risk' pool for referral for further blood and urine testing, we identified a predictive probability cut-off that maximised sensitivity while minimising false positives.	The scope of this research is to build a model using data mining techniques to predict if a patient does indeed have CKD by reviewing and analyzing symptoms and various health parameters	There are real benefits from early detection and management of CKD.You can slow down or even the progression of kidney disease.
Screening strategies for early detection of CKD in low-income and middle-income countries.	Data mining is a powerful and new field having various techniques to analyses recent real-world problems. It converts the raw data into useful information in various research fields and finds the patterns to decide the future medical field.	Our goal was to efficiently identify individuals at highest risk for CKD and to reduce the pool of people requiring referral to primary or secondary health systems for CKD testing, while still allowing early detection and management of CKD.

3.3 PROPOSED SOLUTION

The purpose of this tool is to provide a structured process for identifying a problem. understanding the root causes, ascertaining solution steps, and progress monitoring With a solution template, you can organize development content that you want to reuse for customer-specific solutions Solution templates enable you to easily start the development of customer specific solutions, for example, for a specific industry.



Project Design Phase-I
Proposed Solution Template

Date	03 November 2022
Team ID	PNT2022TMID49895
Project Name	Early Detection of Chronic Kidney Disease
Maximum Marks	2 Marks

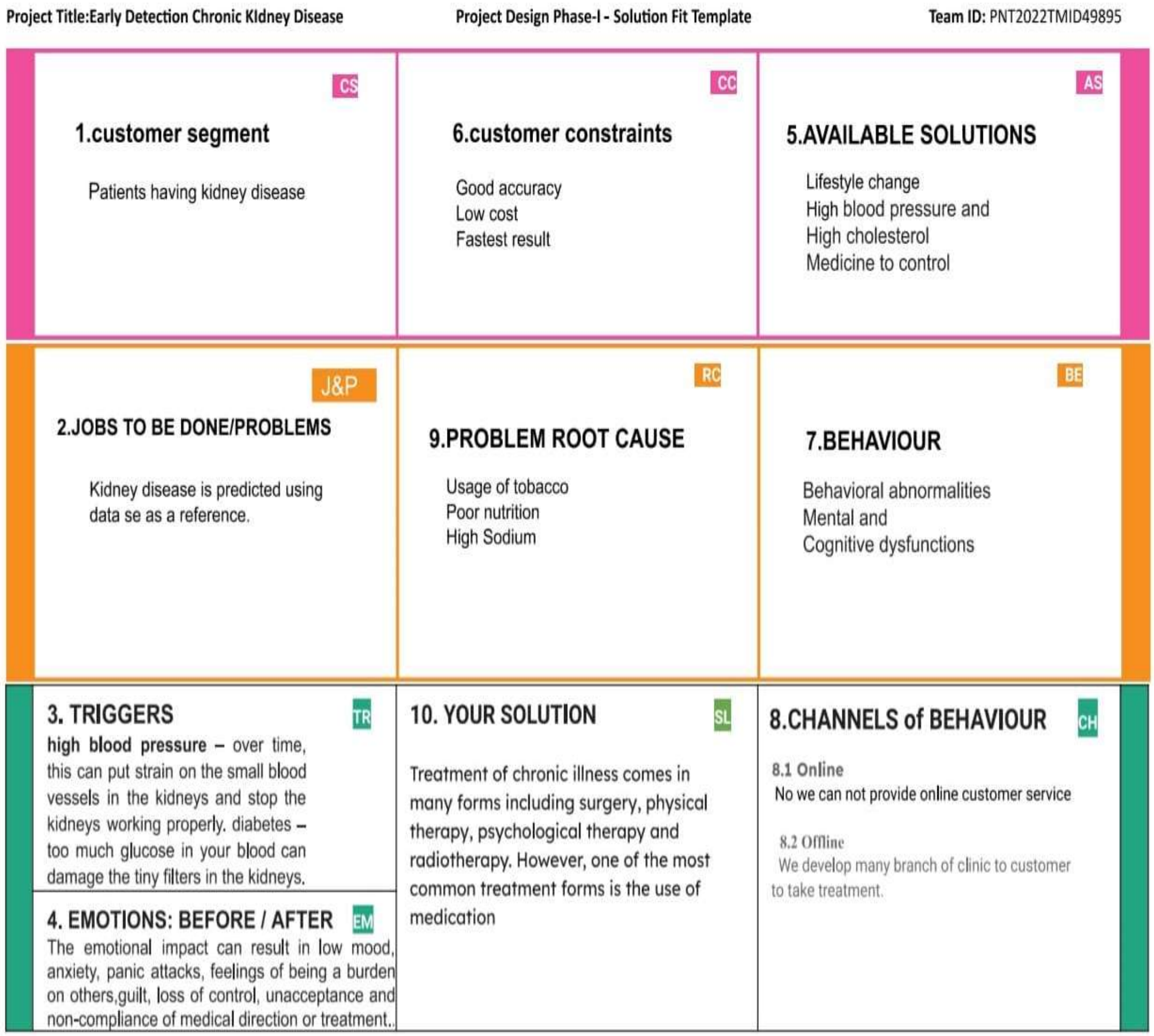
Proposed Solution Template:

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Data mining deals with extraction of helpful data from vast amounts of knowledge. several alternative terms are getting used to knowing data mining, like mining of data from databases, knowledge extraction, information analysis, and information anthropology.
2.	Idea / Solution description	we present predictive models using machine learning techniques that can predict chronic kidney disease including decision tree, logistic regression, K-nearest neighbours and support vector machine models.
3.	Novelty / Uniqueness	The development of the CKD with many other clinical features is related to epidemiology. In general, nephraphrologists use two tests to check for the use of CKD, blood testing and urine tests . Factors that can affect CKD are genetics, diabetes ,obesity, and aging.
4.	Social Impact / Customer Satisfaction	Chronic kidney disease disproportionally affects populations with relatively poor social determinants of health. Knowledge of social determinants, applying what one knows, and addressing the social, economic, and physical barriers to health can help improve individual and population health, reduce health disparities, and advance health equity.
5.	Business Model (Revenue Model)	Business models and business plans are both important tools that help you create and refine your strategy. Many times you will use both when pursuing a new business initiative, but they each serve a different purpose.
6.	Scalability of the Solution	CKD patients regularly receive peritoneal dialysis as a common CKD treatment. An emerging technological solution is telehome care as way to support patients receiving PD in their homes

3.4 PROBLEM SOLUTION FIT

The Problem-Solutions Fit simply means that you have found a problem with your customer and that the solution you have realized for it actually solves the customer's problem Problem-Solution Fit- this occurs when you have evidence that customers care about certain jobs, pains, and gains. At this stage you've proved the existence of a problem and have designed a value proposition that addresses your customers' jobs, pains and gains.



Project Title:Early Detection Chronic KIdney Disease **Project Design Phase-I - Solution Fit Template** **Team ID:** PNT2022TMID49895

CS

1.customer segment

Patients having kidney disease

CC

6.customer constraints

Good accuracy
Low cost
Fastest result

AS

5.AVAILABLE SOLUTIONS

Lifestyle change
High blood pressure and
High cholesterol
Medicine to control

J&P

2.JOBS TO BE DONE/PROBLEMS

Kidney disease is predicted using data se as a reference.

RC

9.PROBLEM ROOT CAUSE

Usage of tobacco
Poor nutrition
High Sodium

BE

7.BEHAVIOUR

Behavioral abnormalities
Mental and
Cognitive dysfunctions

3. TRIGGERS TR

high blood pressure – over time, this can put strain on the small blood vessels in the kidneys and stop the kidneys working properly. diabetes – too much glucose in your blood can damage the tiny filters in the kidneys.

4. EMOTIONS: BEFORE / AFTER EM

The emotional impact can result in low mood, anxiety, panic attacks, feelings of being a burden on others,guilt, loss of control, unacceptance and non-compliance of medical direction or treatment..

10. YOUR SOLUTION SL

Treatment of chronic illness comes in many forms including surgery, physical therapy, psychological therapy and radiotherapy. However, one of the most common treatment forms is the use of medication

8.CHANNELS of BEHAVIOUR CH

8.1 Online

No we can not provide online customer service

8.2 Offline

We develop many branch of clinic to customer to take treatment.

CHAPTER 4

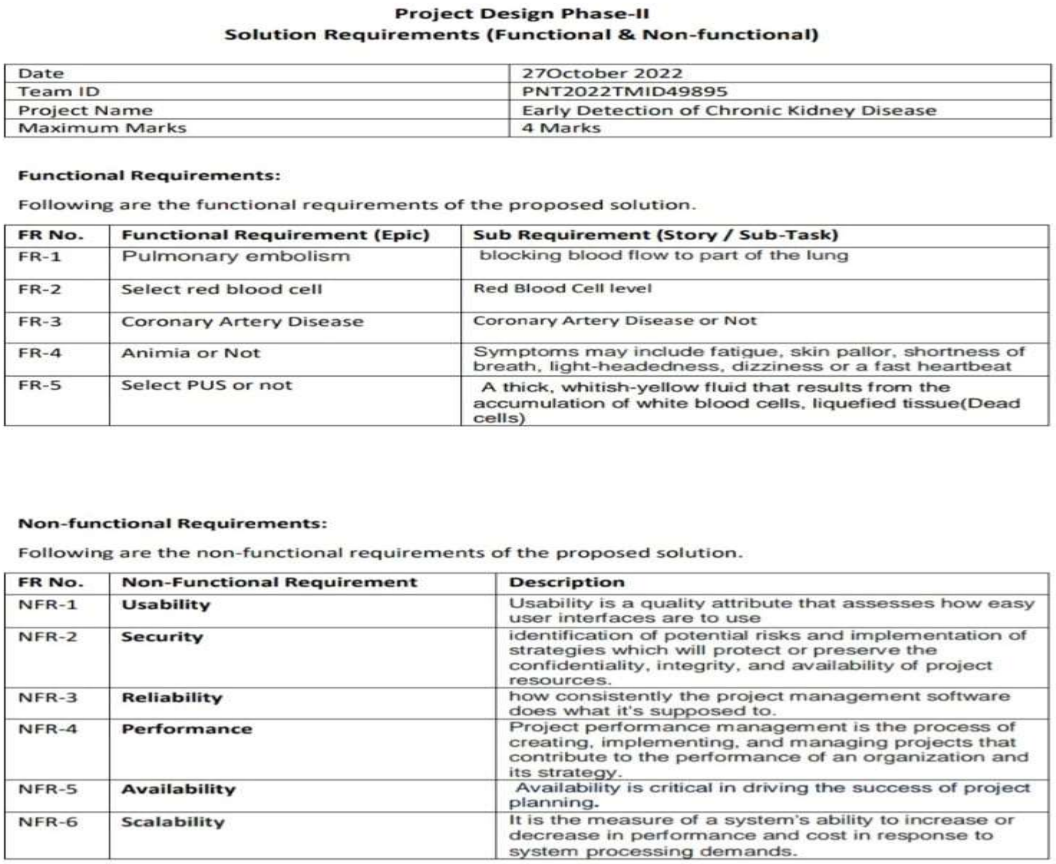
REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT

Solution Requirements are identified before the technical solution is selected and/or designed. They describe the characteristics of a solution (functional and non-functional) that meet business requirements and stakeholder requirements.

4.1 NON-FUNCTIONAL REQUIREMENT

A solution requirement is aimed at the concerns of the people who will build and deliver the solution. It tells those people what the functional and non-functional requirements for the solution will be and how the solution will deliver on the business and stakeholder requirements. Solution Requirements Describe the features, functions, and characteristics of a product, service, or result that will meet the business and stakeholder requirements.



Project Design Phase-II
Solution Requirements (Functional & Non-functional)

Date	27October 2022
Team ID	PNT2022TMID49895
Project Name	Early Detection of Chronic Kidney Disease
Maximum Marks	4 Marks

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Pulmonary embolism	blocking blood flow to part of the lung
FR-2	Select red blood cell	Red Blood Cell level
FR-3	Coronary Artery Disease	Coronary Artery Disease or Not
FR-4	Animia or Not	Symptoms may include fatigue, skin pallor, shortness of breath, light-headedness, dizziness or a fast heartbeat
FR-5	Select PUS or not	A thick, whitish-yellow fluid that results from the accumulation of white blood cells, liquefied tissue(Dead cells)

Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

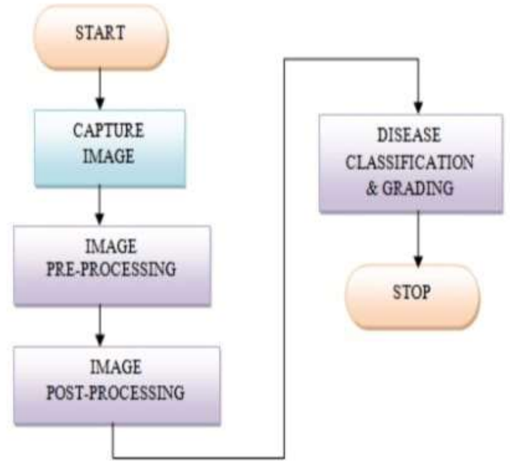
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Usability is a quality attribute that assesses how easy user interfaces are to use
NFR-2	**Security**	identification of potential risks and implementation of strategies which will protect or preserve the confidentiality, integrity, and availability of project resources.
NFR-3	**Reliability**	how consistently the project management software does what it's supposed to.
NFR-4	**Performance**	Project performance management is the process of creating, implementing, and managing projects that contribute to the performance of an organization and its strategy.
NFR-5	**Availability**	Availability is critical in driving the success of project planning.
NFR-6	**Scalability**	It is the measure of a system's ability to increase or decrease in performance and cost in response to system processing demands.

CHAPTER 5

PROJECT DESIGN

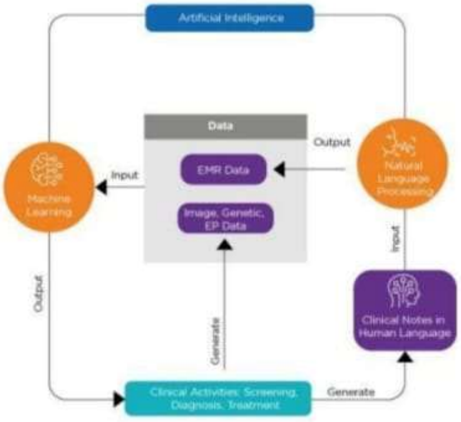
5.1 DATA FLOW DIAGRAMS

A Data Flow Diagram (DFD) is a graphical representation of the "flow" of data through an information system modeling its process aspects. Often it is a preliminary step used to create an overview of the system that can later be elaborated in software development and product management, a user story is an informal, natural language description of features of a software system. They are written from the perspective of an end user or user of a system, and may be recorded on index cards, Post-it notes, or digitally in project management software.



Example: Flow chart for disease detection

Example: DFD Level 0 (AI driven digitization for chronic disease)



5.2 SOLUTION & TECHNICAL ARCHITECTURE

A solutions architect creates the overall technical vision for a specific solution to a business problem. A solutions architect creates the overall technical vision for a specific solution to a business problem. They design describe, and manage the solution

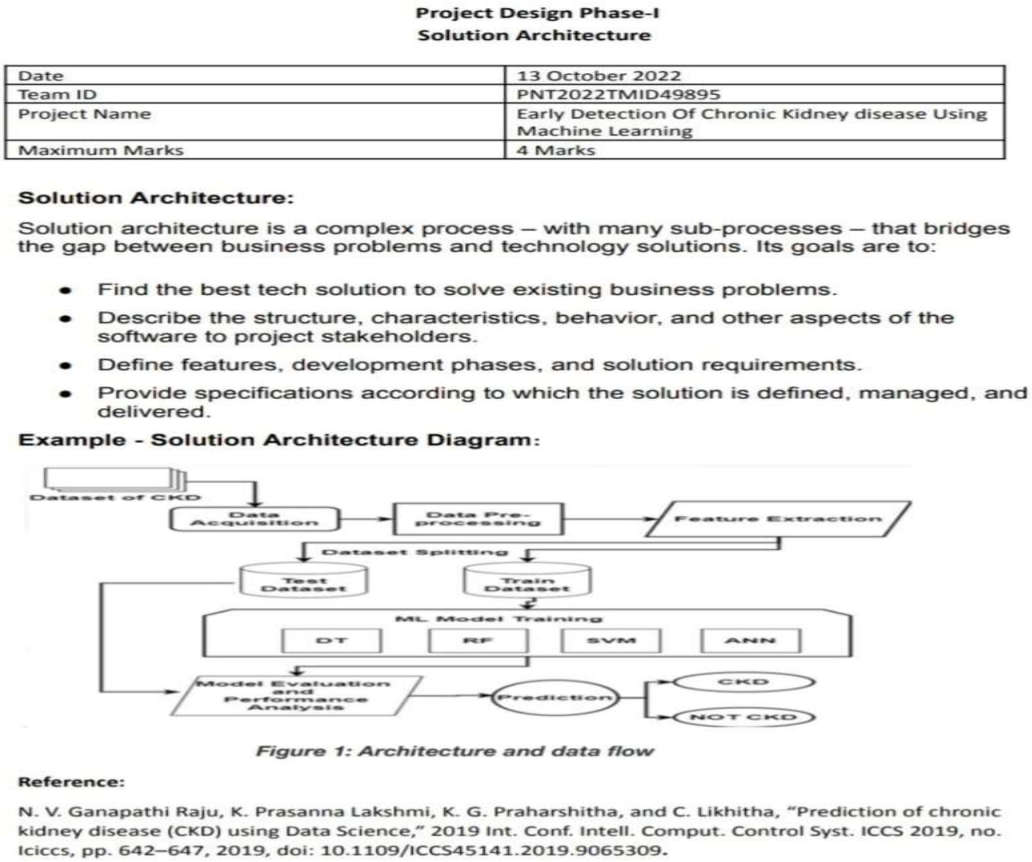
- ❖ Solution Architect Processes, Role Description, Responsibilities, and Certifications.

- ❖ Matching solutions with the corporate environment

- ❖ Meeting the requirements of all stakeholders.
- ❖ Accounting for project constraints.

- ❖ Selecting the project technology stack.

- ❖ Compliance with non-functional requirements



Project Design Phase-I
Solution Architecture

Date	13 October 2022
Team ID	PNT2022TMID49895
Project Name	Early Detection Of Chronic Kidney disease Using Machine Learning
Maximum Marks	4 Marks

Solution Architecture:

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

- Find the best tech solution to solve existing business problems.
- Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.
- Define features, development phases, and solution requirements.
- Provide specifications according to which the solution is defined, managed, and delivered.

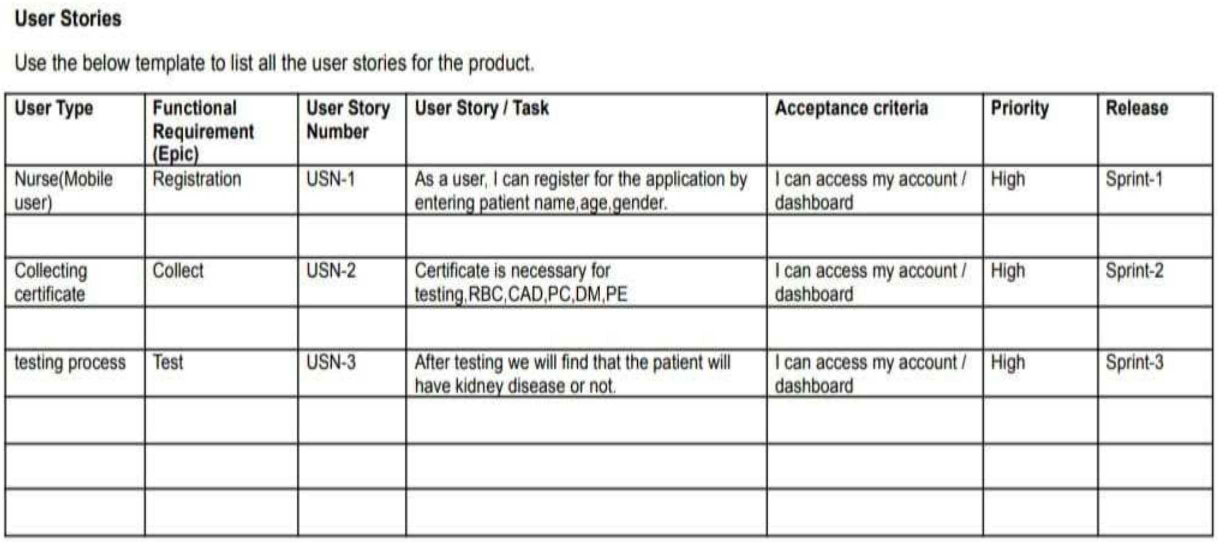
Example - Solution Architecture Diagram:

Figure 1: Architecture and data flow

Reference:

N. V. Ganapathi Raju, K. Prasanna Lakshmi, K. G. Praharshitha, and C. Likhitha, "Prediction of chronic kidney disease (CKD) using Data Science," 2019 Int. Conf. Intell. Comput. Control Syst. ICCS 2019, no. Iciccs, pp. 642–647, 2019, doi: 10.1109/ICCS45141.2019.9065309.

5.3 USER STORIES



User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Nurse(Mobile user)	Registration	USN-1	As a user, I can register for the application by entering patient name,age,gender.	I can access my account / dashboard	High	Sprint-1
Collecting certificate	Collect	USN-2	Certificate is necessary for testing,RBC,CAD,PC,DM,PE	I can access my account / dashboard	High	Sprint-2
testing process	Test	USN-3	After testing we will find that the patient will have kidney disease or not.	I can access my account / dashboard	High	Sprint-3

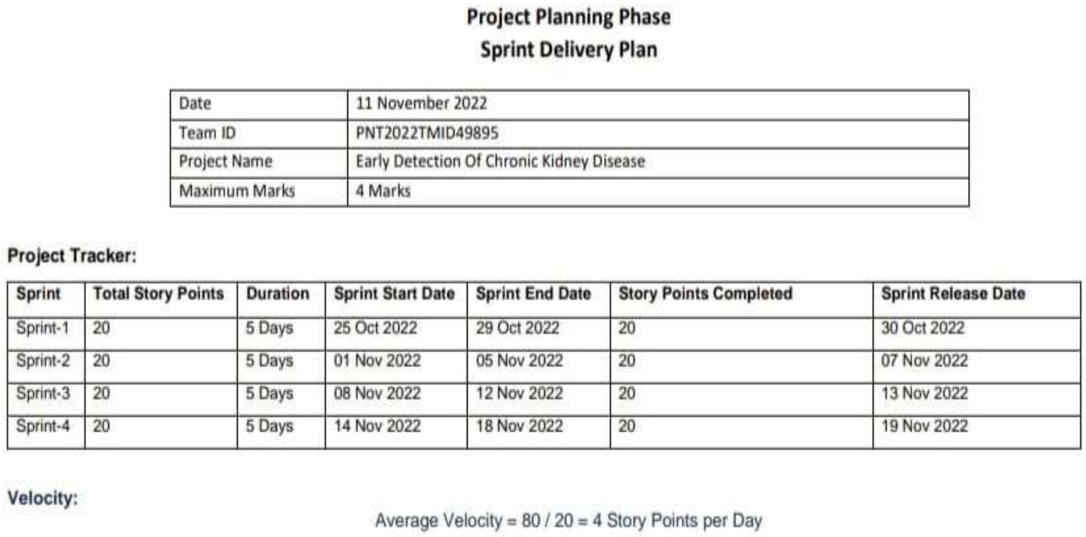
CHAPETR 6

PROJECT PLANNING & SCHEDULING

During the Project Planning Phase we have done Project planning template, Milestone and activity list and Jira sprint delivery plan.

6.1 SPRINT PLANNING & ESTIMATION

A project plan template is a document that creates a standard format for a project plan. Typically, it contains a list of the essential elements of a project, such as stakeholders. scope, timelines, estimated cost and communication methods. The project manager typically lists the information based on the assignment.



Project Planning Phase
Sprint Delivery Plan

Date	11 November 2022
Team ID	PNT2022TMID49895
Project Name	Early Detection Of Chronic Kidney Disease
Maximum Marks	4 Marks

Project Tracker:

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date	Story Points Completed	Sprint Release Date
Sprint-1	20	5 Days	25 Oct 2022	29 Oct 2022	20	30 Oct 2022
Sprint-2	20	5 Days	01 Nov 2022	05 Nov 2022	20	07 Nov 2022
Sprint-3	20	5 Days	08 Nov 2022	12 Nov 2022	20	13 Nov 2022
Sprint-4	20	5 Days	14 Nov 2022	18 Nov 2022	20	19 Nov 2022

Velocity:

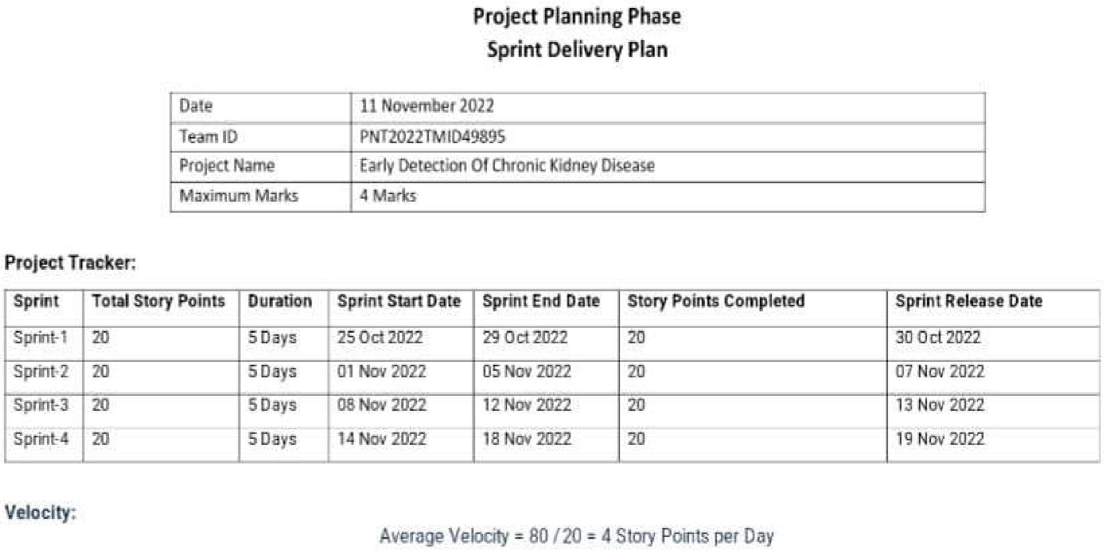
Average Velocity = 80 / 20 = 4 Story Points per Day

6.2 SPRINT DELIVERY SCHEDULE

A milestone list is a project management document that identifies all project milestones. A milestone is a significant event or a point in a project. It represents nothing more than a moment in time, hence, when scheduling, milestones should be assigned zero duration.

A milestone is a specific point within a project's life cycle used to measure the progress toward the ultimate goal. Milestones in project management are used as signal posts for a project's start or end date, external reviews or input, budget checks, submission of a major deliverable, etc.

Milestone is a point on the schedule, which has one clearly defined deliverable, whereas a task is an activity that should be completed to achieve a milestone. An activity/task has a start and end date. It's two dimensional, whereas a milestone is a single date upon which a deliverable gets completed



Project Planning Phase
Sprint Delivery Plan

Date	11 November 2022
Team ID	PNT2022TMID49895
Project Name	Early Detection Of Chronic Kidney Disease
Maximum Marks	4 Marks

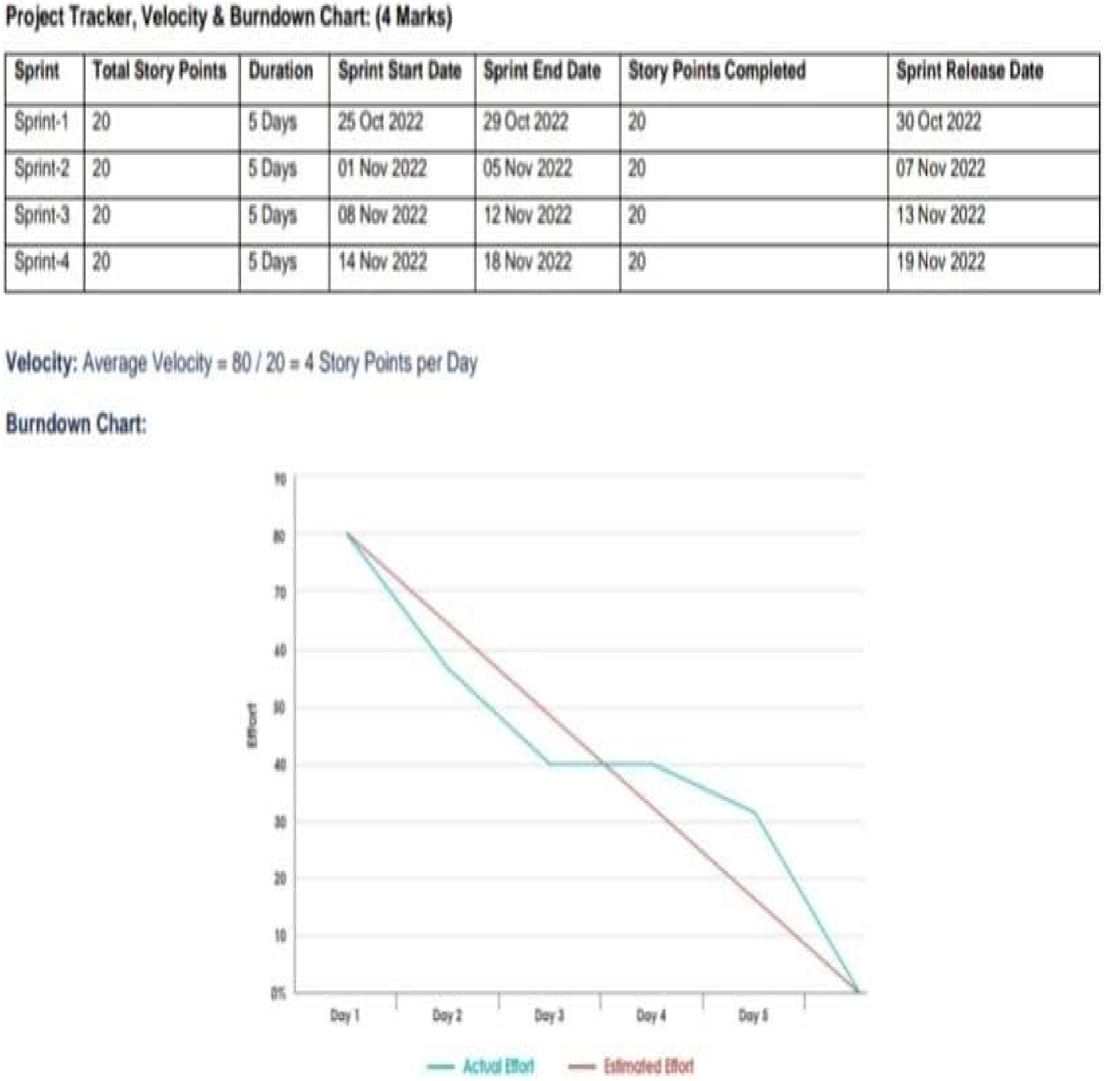
Project Tracker:

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date	Story Points Completed	Sprint Release Date
Sprint-1	20	5 Days	25 Oct 2022	29 Oct 2022	20	30 Oct 2022
Sprint-2	20	5 Days	01 Nov 2022	05 Nov 2022	20	07 Nov 2022
Sprint-3	20	5 Days	08 Nov 2022	12 Nov 2022	20	13 Nov 2022
Sprint-4	20	5 Days	14 Nov 2022	18 Nov 2022	20	19 Nov 2022

Velocity:

Average Velocity = 80 / 20 = 4 Story Points per Day

6.3 BURNDOWN CHART



Project Tracker, Velocity & Burndown Chart: (4 Marks)

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date	Story Points Completed	Sprint Release Date
Sprint-1	20	5 Days	25 Oct 2022	29 Oct 2022	20	30 Oct 2022
Sprint-2	20	5 Days	01 Nov 2022	05 Nov 2022	20	07 Nov 2022
Sprint-3	20	5 Days	08 Nov 2022	12 Nov 2022	20	13 Nov 2022
Sprint-4	20	5 Days	14 Nov 2022	18 Nov 2022	20	19 Nov 2022

Velocity: Average Velocity = 80 / 20 = 4 Story Points per Day

Burndown Chart:

CHAPTER 7

CODING AND SOLUTIONING:

During the Project Development Phase we have done four Sprints they are Sprint 1, Sprint2, Sprint 3 and Sprint 4. In Agile product development, sprint is a set period of time during which specific work has to be completed and made ready for review Each sprint begins with a planning meeting. During the meeting, the product owner (the person requesting the work) and the development team agree upon exactly what work will be accomplished during the sprint. The development team has the final say when it comes to determining how much work can realistically be accomplished during the sprint, and the product owner has the final say on what criteria need to be met for the work to be approved and accepted.

The duration of a sprint is determined by the scrum master, the team's facilitator and manager of the Scrum framework. Once the team reaches a consensus for how many- days a sprint should last, all future sprints should be the same. Traditionally, a sprint fasts 30 days After a sprint begins, the product owner must step back and let the team do their work During the sprint, the team holds daily stand-up meetings to discuss progress and brainstorm solutions to challenges. The project owner may attend these meetings as an observer but is not allowed to participate unless it is to answer questions. The project owner may not make requests for changes during a sprint and only the scrum master or project manager has the power to interrupt or stop the sprint.

At the end of the sprint, the team presents its completed work to the project owner and the project owner uses the criteria established at the sprint planning meeting to either accept or reject the work.

7.1 FEATURE 1 Software Requirement Specification (SRS)

PANDAS

Pandas is an open-source, BSD-authorized Python library giving elite, simple to-utilize information structures and information examination instruments for the Python programming language. Python with Pandas is utilized in a wide scope of fields including scholastic and business areas including money, financial matters, Statistics, examination, and so on. In this instructional exercise, we will get familiar with the different highlights of Python Pandas and how to utilize them practically speaking.

This instructional exercise has been set up for the individuals who try to become familiar with the essentials and different elements of Pandas. It will be explicitly valuable for individuals working with information purging and examination. In the wake of finishing this instructional exercise, you will wind up at a moderate dimension of ability from where you can take yourself to more elevated amounts of skill. You ought to have a fundamental comprehension of Computer Programming phrasings. A fundamental comprehension of any of the programming dialects is an or more.

Pandas library utilizes the vast majority of the functionalities of NumPy. It is recommended that you experience our instructional exercise on NumPy before continuing with this instructional exercise. 2.4.5 ANACONDA constrictor is bundle director. Jupyter is an introduction layer. Boa constrictor end to explain the reliance damnation in python—where distinctive tasks have diverse reliance variants—in order to not influence distinctive venture conditions to require diverse adaptations, which may meddle with one another. Jupyter to fathom the issue of reproducibility in investigation by empowering an iterative and hands-on way to deal with clarifying and imagining code; by utilizing rich content documentations joined with visual portrayals, in a solitary arrangement.

Boa constrictor is like it's intended to accomplish a python situation that is 100% reproducible on another condition, autonomous of whatever different forms of a task's conditions are accessible. It's somewhat like Docker, however limited to the Python biological system. Jupyter is an astounding introduction device for expository work; where you can display code in "squares," joins with rich content depictions among squares, and the consideration of organized yield from the squares, and charts created in an all around planned issue by method for another square's code.

Jupyter is extraordinarily great in expository work to guarantee reproducibility in somebody's exploration, so anybody can return numerous months after the fact and outwardly comprehend what somebody attempted to clarify, and see precisely which code drove which representation and end. Regularly in diagnostic work you will finish up with huge amounts of half-completed note pads clarifying Proof-of-Concept thoughts, of which most won't lead anyplace at first. A portion of these introductions may months after the fact—or even years after the fact— present an establishment to work from for another issue.

PYTHON

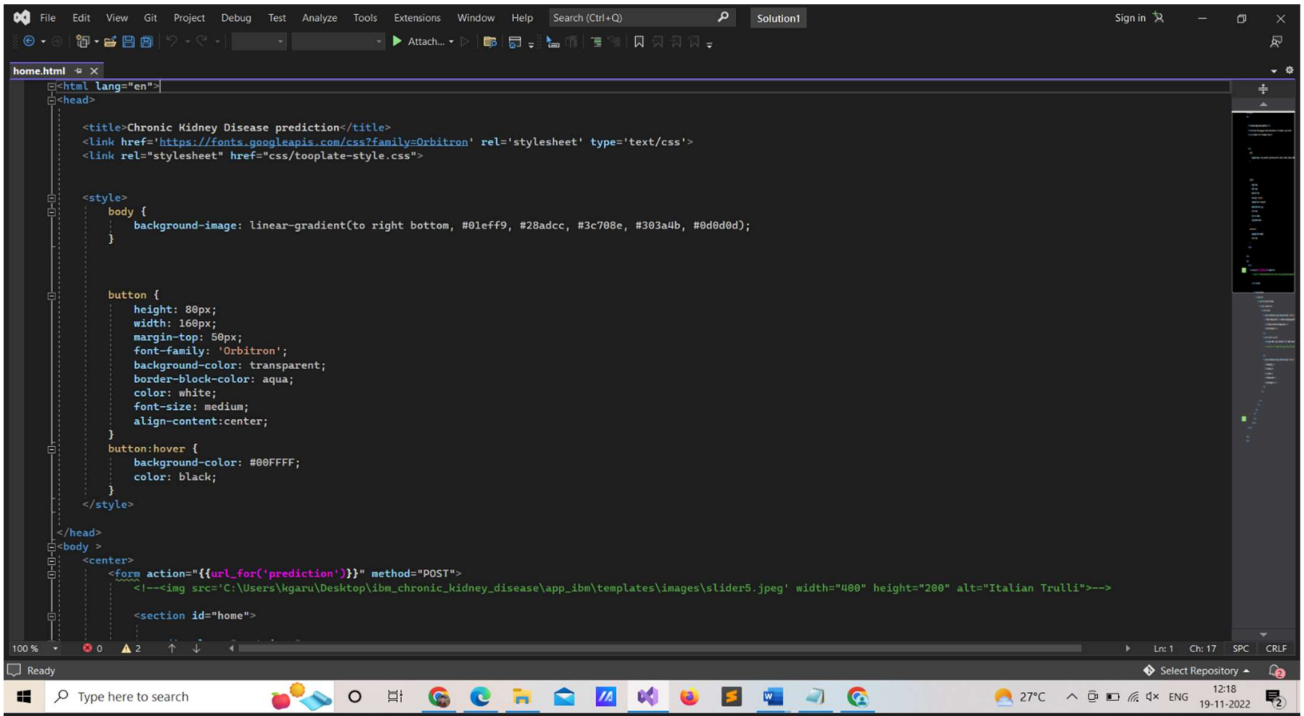
Python is a translated, object-arranged, abnormal state programming language with dynamic semantics. Its abnormal state worked in information structures, joined with dynamic composing and dynamic authoritative, make it appealing for Rapid Application Development, just as for use as a scripting or paste language to interface existing segments together. Python's basic, simple to learn language structure underlines intelligibility and hence decreases the expense of program support. Python underpins modules and bundles, which empowers program seclusion and code reuse. The Python translator and the broad standard library are accessible in source or parallel structure without charge for every single significant stage, and can be openly appropriated. Frequently, software engineers begin to look all starry eyed at Python on account of the expanded efficiency it gives. Since there is no aggregation step, the alter test-troubleshoot cycle is staggeringly quick.

Troubleshooting Python programs is simple: a bug or awful information will never cause a division blame. Rather, when the mediator finds a blunder, it raises a special case. At the point when the program doesn't get the special case, the translator prints a stack follow. A source level debugger permits assessment of nearby and worldwide factors, assessment of discretionary articulations, setting breakpoints, venturing through the code a line at any given moment, etc. The debugger is written in Python itself, vouching for Python's contemplative power. Then again, frequently the speediest method to troubleshoot a program is to add a couple of print proclamations to the source: the quick alter test investigate cycle makes this straightforward methodology successful. Python is an item situated, abnormal state programming language with incorporated unique semantics essentially for web and application improvement. It is amazingly alluring in the field of Rapid Application Development since it offers dynamic composing and dynamic restricting alternatives.

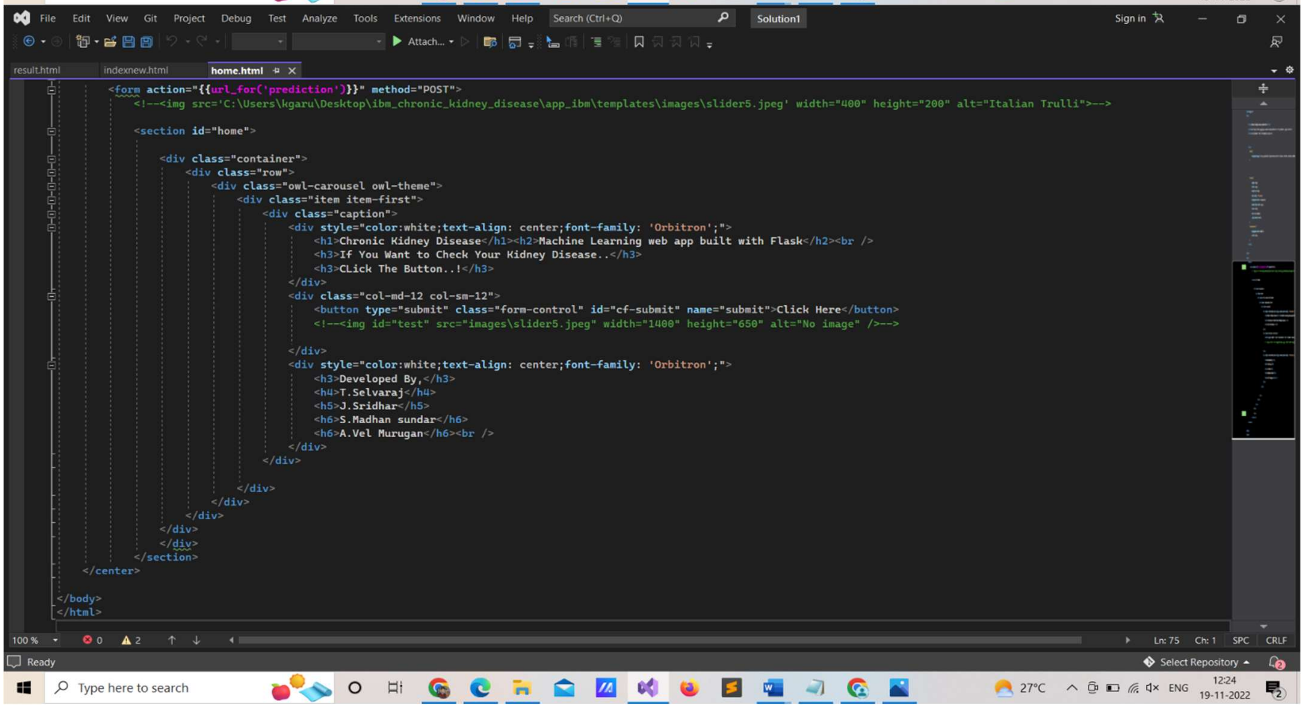
Python is generally basic, so it's anything but difficult to learn since it requires a one of a kind language structure that centers around coherence. Designers can peruse and interpret Python code a lot simpler than different dialects. Thusly, this decreases the expense of program upkeep and improvement since it enables groups to work cooperatively without huge language and experience obstructions. A standout amongst the most encouraging advantages of Python is that both the standard library and the mediator are accessible for nothing out of pocket, in both parallel and source structure. There is no restrictiveness either, as Python and all the important instruments are accessible on every single real stage. In this way, it is a tempting alternative for designers who would prefer not to stress over paying high improvement costs.

7.2 FEATURE

HTML FOR HOME PAGE

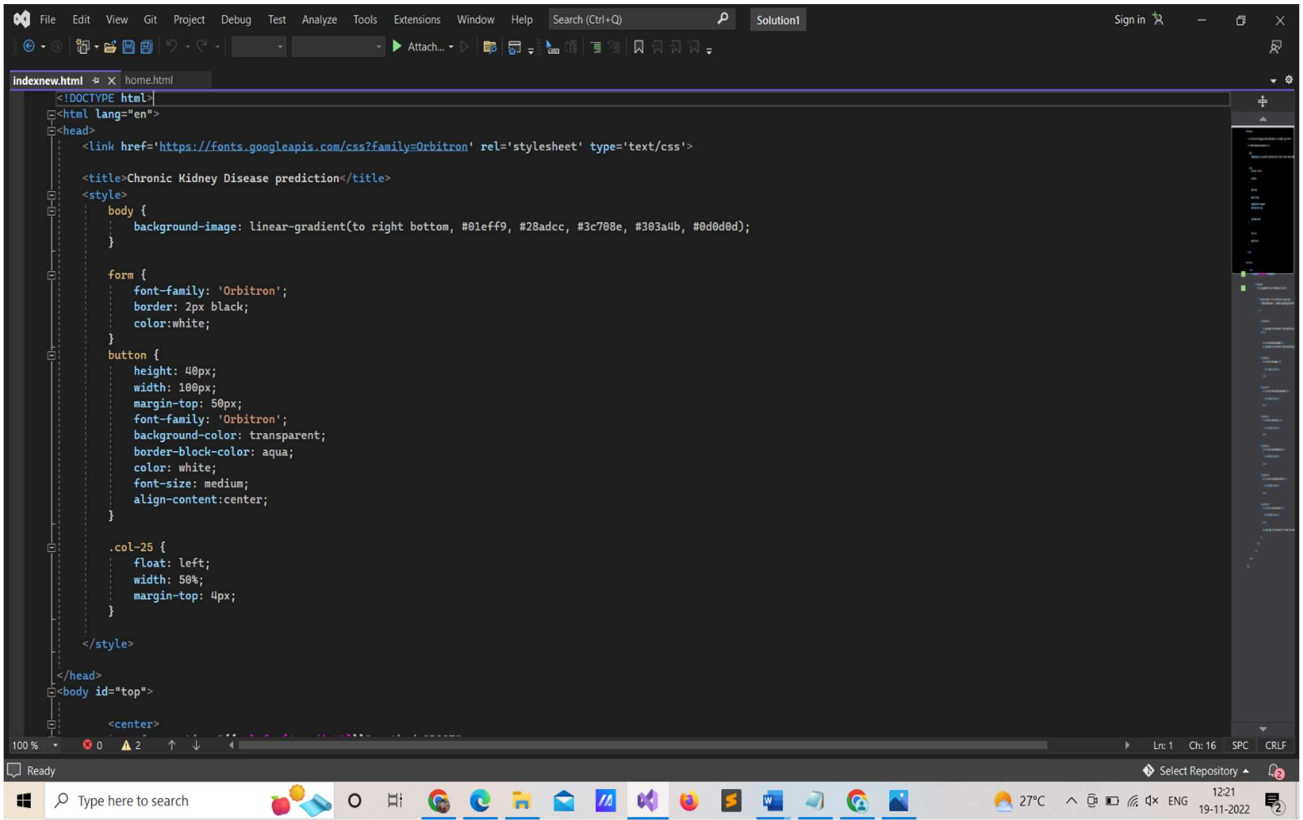


```html
<html lang="en">
<head>

    <title>Chronic Kidney Disease prediction</title>
    <link href='https://fonts.googleapis.com/css?family=Orbitron' rel='stylesheet' type='text/css'>
    <link rel="stylesheet" href="css/tooplate-style.css">


    <style>
        body {
            background-image: linear-gradient(to right bottom, #01eff9, #28adcc, #3c708e, #303a4b, #0d0d0d);
        }



        button {
            height: 80px;
            width: 160px;
            margin-top: 50px;
            font-family: 'Orbitron';
            background-color: transparent;
            border-block-color: aqua;
            color: white;
            font-size: medium;
            align-content:center;
        }
        button:hover {
            background-color: #00FFFF;
            color: black;
        }
    </style>

</head>
<body >
    <center>
        <form action="{{url_for('prediction')}}" method="POST">
            <!--<img src='C:\Users\kgaru\Desktop\ibm_chronic_kidney_disease\app_ibm\templates\images\slider5.jpeg' width="400" height="200" alt="Italian Trulli">-->

            <section id="home">
```

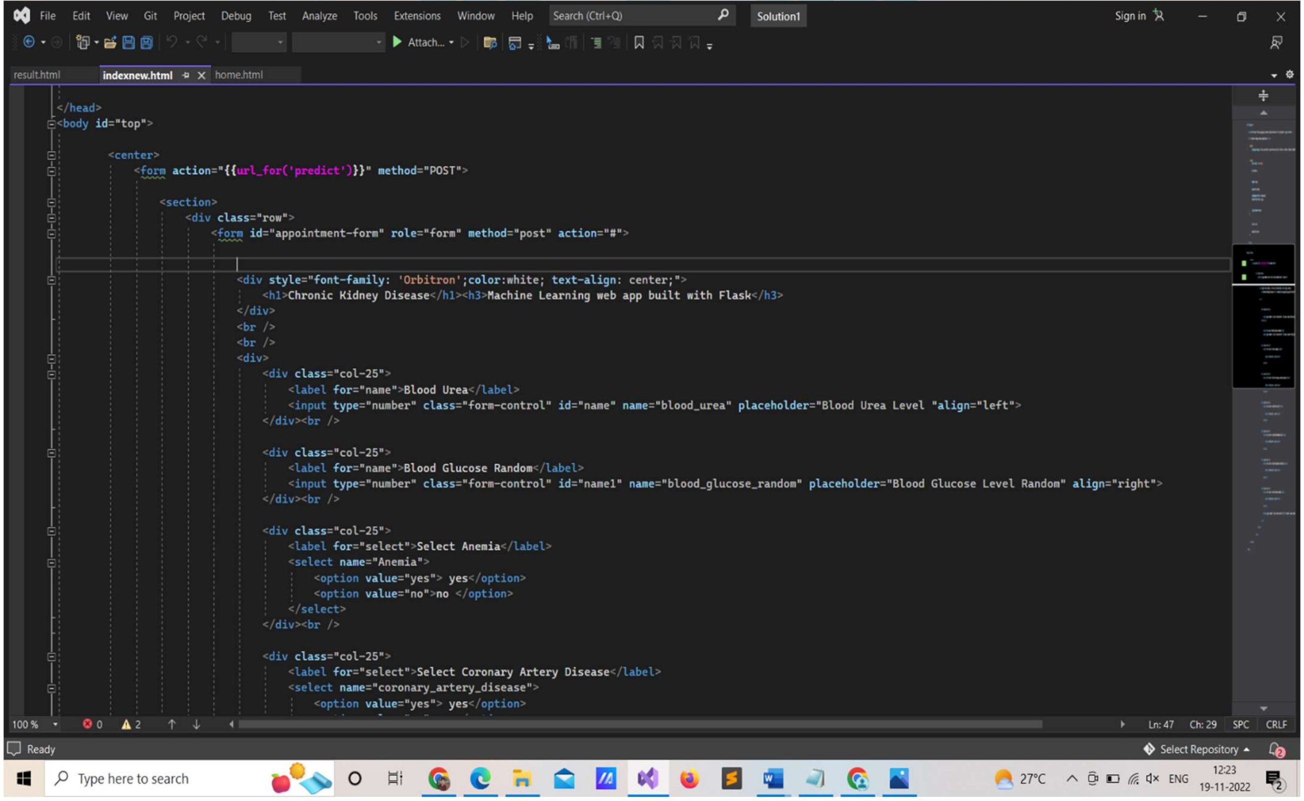


```html
        <form action="{{url_for('prediction')}}" method="POST">
            <!--<img src='C:\Users\kgaru\Desktop\ibm_chronic_kidney_disease\app_ibm\templates\images\slider5.jpeg' width="400" height="200" alt="Italian Trulli">-->

            <section id="home">

                <div class="container">
                    <div class="row">
                        <div class="owl-carousel owl-theme">
                            <div class="item item-first">
                                <div class="caption">
                                    <div style="color:white;text-align: center;font-family: 'Orbitron';">
                                        <h1>Chronic Kidney Disease</h1><h2>Machine Learning web app built with Flask</h2><br />
                                        <h3>If You Want to Check Your Kidney Disease..</h3>
                                        <h3>CLick The Button..!</h3>
                                    </div>
                                    <div class="col-md-12 col-sm-12">
                                        <button type="submit" class="form-control" id="cf-submit" name="submit">Click Here</button>
                                        <!--<img id="test" src="images\slider5.jpeg" width="1400" height="650" alt="No image" />-->

                                    </div>
                                    <div style="color:white;text-align: center;font-family: 'Orbitron';">
                                        <h3>Developed By,</h3>
                                        <h4>T.Selvaraj</h4>
                                        <h5>J.Sridhar</h5>
                                        <h6>S.Madhan sundar</h6>
                                        <h6>A.Vel Murugan</h6><br />
                                    </div>
                                </div>

                            </div>
                        </div>
                    </div>
                </div>
                </div>
            </section>
    </center>

</body>
</html>
```

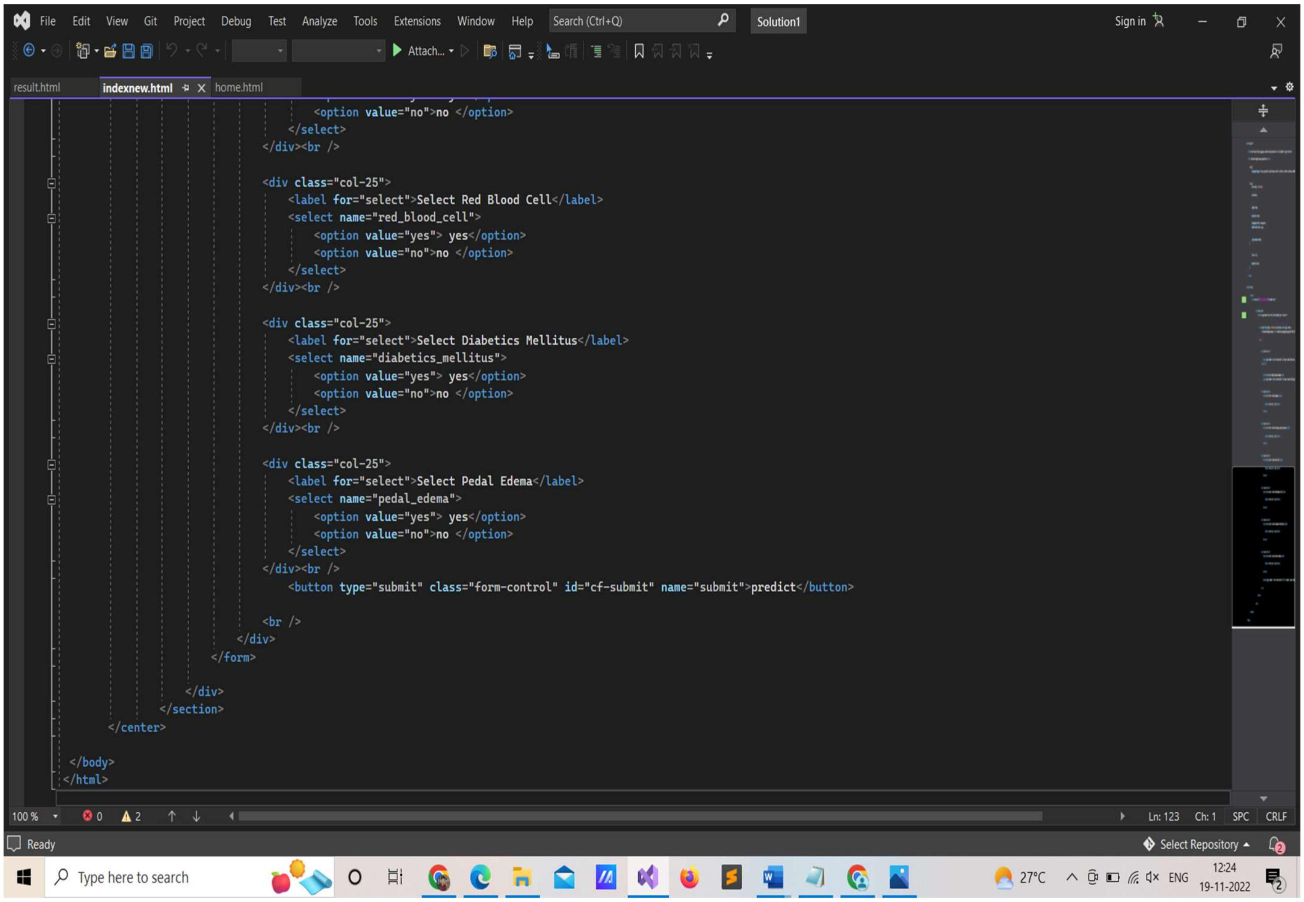
HTML FOR INDEXNEW PAGE



```html
<!DOCTYPE html>
<html lang="en">
<head>
    <link href='https://fonts.googleapis.com/css?family=Orbitron' rel='stylesheet' type='text/css'>

    <title>Chronic Kidney Disease prediction</title>
    <style>
        body {
            background-image: linear-gradient(to right bottom, #01eff9, #28adcc, #3c708e, #303a4b, #0d0d0d);
        }

        form {
            font-family: 'Orbitron';
            border: 2px black;
            color:white;
        }
        button {
            height: 40px;
            width: 100px;
            margin-top: 50px;
            font-family: 'Orbitron';
            background-color: transparent;
            border-block-color: aqua;
            color: white;
            font-size: medium;
            align-content:center;
        }

        .col-25 {
            float: left;
            width: 50%;
            margin-top: 4px;
        }

    </style>

</head>
<body id="top">

        <center>
```

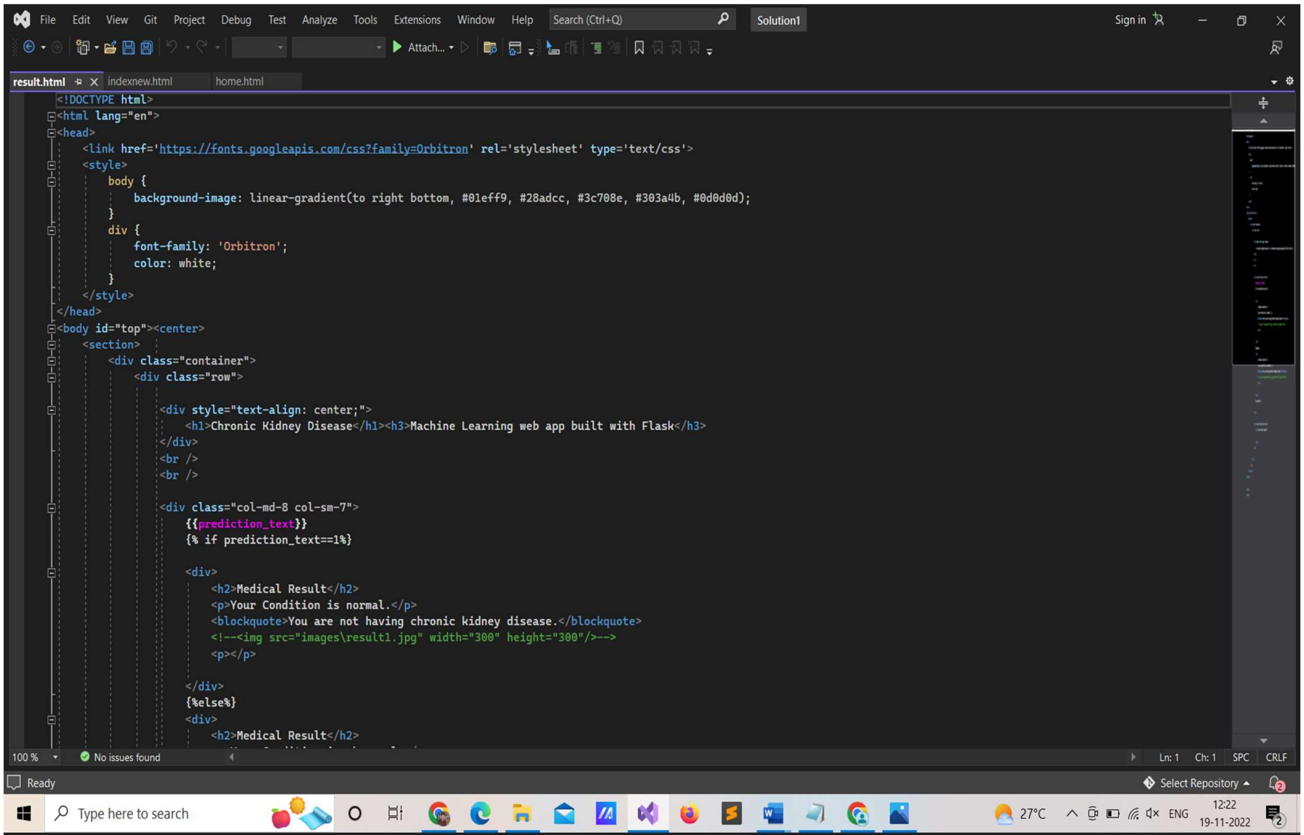


```html
</head>
<body id="top">

        <center>
            <form action="{{url_for('predict')}}" method="POST">

                <section>
                    <div class="row">
                        <form id="appointment-form" role="form" method="post" action="#">

                            <div style="font-family: 'Orbitron';color:white; text-align: center;">
                                <h1>Chronic Kidney Disease</h1><h3>Machine Learning web app built with Flask</h3>
                            </div>
                            <br />
                            <br />
                            <div>
                                <div class="col-25">
                                    <label for="name">Blood Urea</label>
                                    <input type="number" class="form-control" id="name" name="blood_urea" placeholder="Blood Urea Level "align="left">
                                </div><br />

                                <div class="col-25">
                                    <label for="name">Blood Glucose Random</label>
                                    <input type="number" class="form-control" id="name1" name="blood_glucose_random" placeholder="Blood Glucose Level Random" align="right">
                                </div><br />

                                <div class="col-25">
                                    <label for="select">Select Anemia</label>
                                    <select name="Anemia">
                                        <option value="yes"> yes</option>
                                        <option value="no">no </option>
                                    </select>
                                </div><br />

                                <div class="col-25">
                                    <label for="select">Select Coronary Artery Disease</label>
                                    <select name="coronary_artery_disease">
                                        <option value="yes"> yes</option>
```

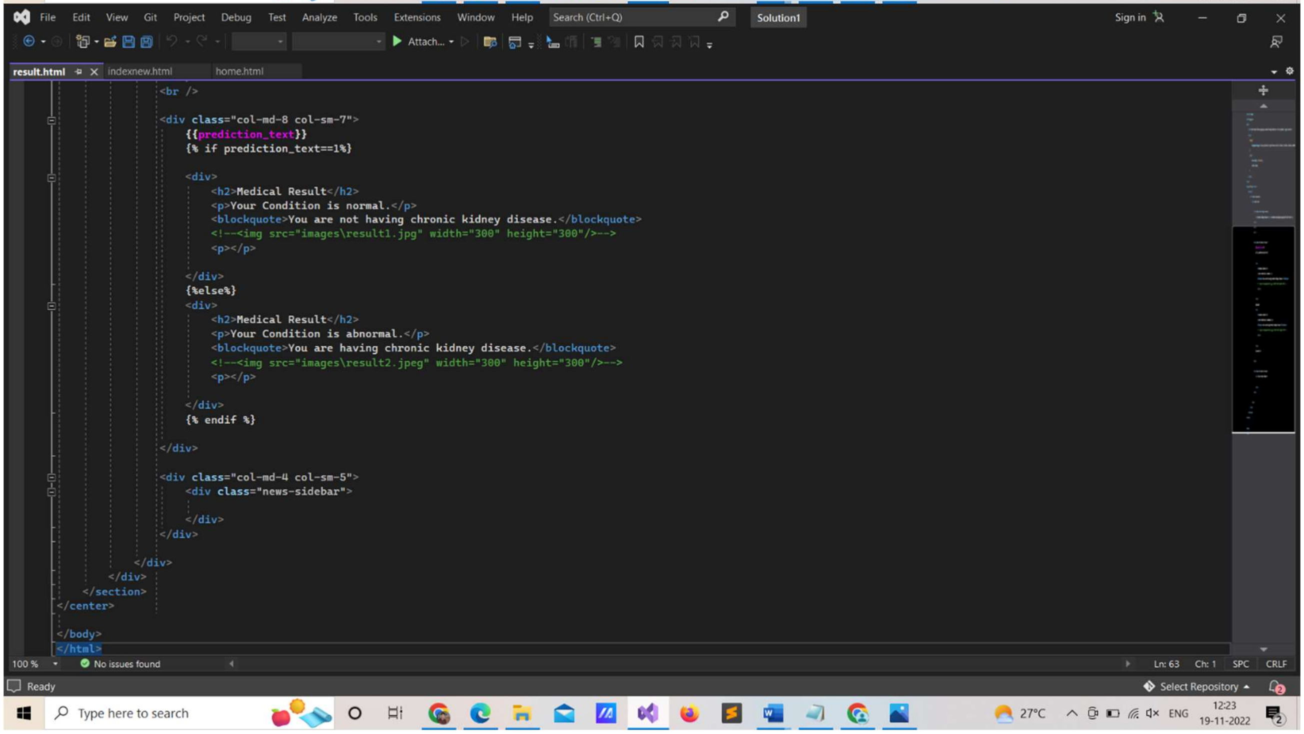


```
                                        <option value="no">no </option>
                                    </select>
                                </div><br />

                                <div class="col-25">
                                    <label for="select">Select Red Blood Cell</label>
                                    <select name="red_blood_cell">
                                        <option value="yes"> yes</option>
                                        <option value="no">no </option>
                                    </select>
                                </div><br />

                                <div class="col-25">
                                    <label for="select">Select Diabetics Mellitus</label>
                                    <select name="diabetics_mellitus">
                                        <option value="yes"> yes</option>
                                        <option value="no">no </option>
                                    </select>
                                </div><br />

                                <div class="col-25">
                                    <label for="select">Select Pedal Edema</label>
                                    <select name="pedal_edema">
                                        <option value="yes"> yes</option>
                                        <option value="no">no </option>
                                    </select>
                                </div><br />
                                    <button type="submit" class="form-control" id="cf-submit" name="submit">predict</button>

                                <br />
                            </div>
                        </form>

                    </div>
                </section>
        </center>

 </body>
</html>
```

HTML FOR RESULT PAGE



```html
<!DOCTYPE html>
<html lang="en">
<head>
    <link href='https://fonts.googleapis.com/css?family=Orbitron' rel='stylesheet' type='text/css'>
    <style>
        body {
            background-image: linear-gradient(to right bottom, #01eff9, #28adcc, #3c708e, #303a4b, #0d0d0d);
        }
        div {
            font-family: 'Orbitron';
            color: white;
        }
    </style>
</head>
<body id="top"><center>
    <section>
        <div class="container">
            <div class="row">

                <div style="text-align: center;">
                    <h1>Chronic Kidney Disease</h1><h3>Machine Learning web app built with Flask</h3>
                </div>
                <br />
                <br />

                <div class="col-md-8 col-sm-7">
                    {{prediction_text}}
                    {% if prediction_text==1%}

                    <div>
                        <h2>Medical Result</h2>
                        <p>Your Condition is normal.</p>
                        <blockquote>You are not having chronic kidney disease.</blockquote>
                        <!--<img src="images\result1.jpg" width="300" height="300"/>-->
                        <p></p>

                    </div>
                    {%else%}
                    <div>
                        <h2>Medical Result</h2>
```



```html
                <br />

                <div class="col-md-8 col-sm-7">
                    {{prediction_text}}
                    {% if prediction_text==1%}

                    <div>
                        <h2>Medical Result</h2>
                        <p>Your Condition is normal.</p>
                        <blockquote>You are not having chronic kidney disease.</blockquote>
                        <!--<img src="images\result1.jpg" width="300" height="300"/>-->
                        <p></p>

                    </div>
                    {%else%}
                    <div>
                        <h2>Medical Result</h2>
                        <p>Your Condition is abnormal.</p>
                        <blockquote>You are having chronic kidney disease.</blockquote>
                        <!--<img src="images\result2.jpeg" width="300" height="300"/>-->
                        <p></p>

                    </div>
                    {% endif %}

                </div>

                <div class="col-md-4 col-sm-5">
                    <div class="news-sidebar">

                    </div>
                </div>

            </div>
        </div>
    </section>
</center>

</body>
</html>
```

CHAPTER 8

TESTING

8.1 Test Cases

Functional Testing

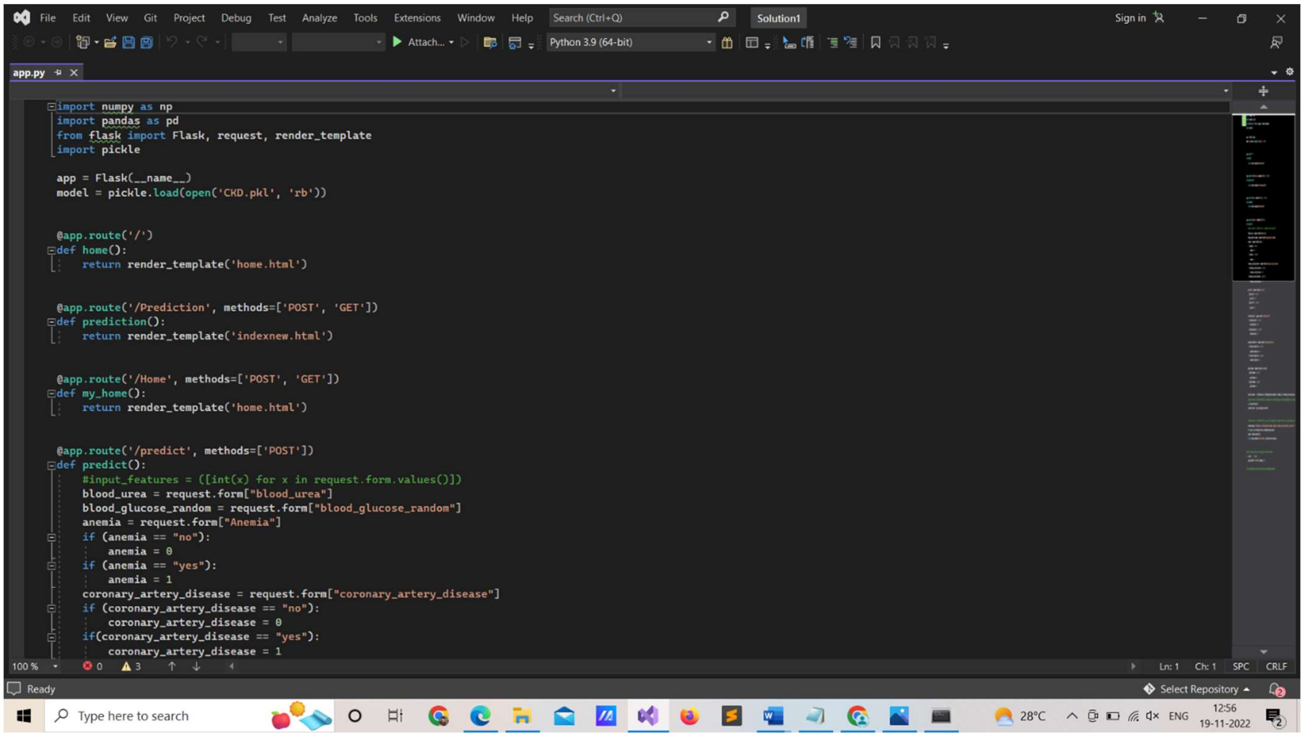
Functional Testing is also known as functional completeness testing, Functional Testing involves trying to think of any possible missing functions. Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals

Functional testing is centered on the following items

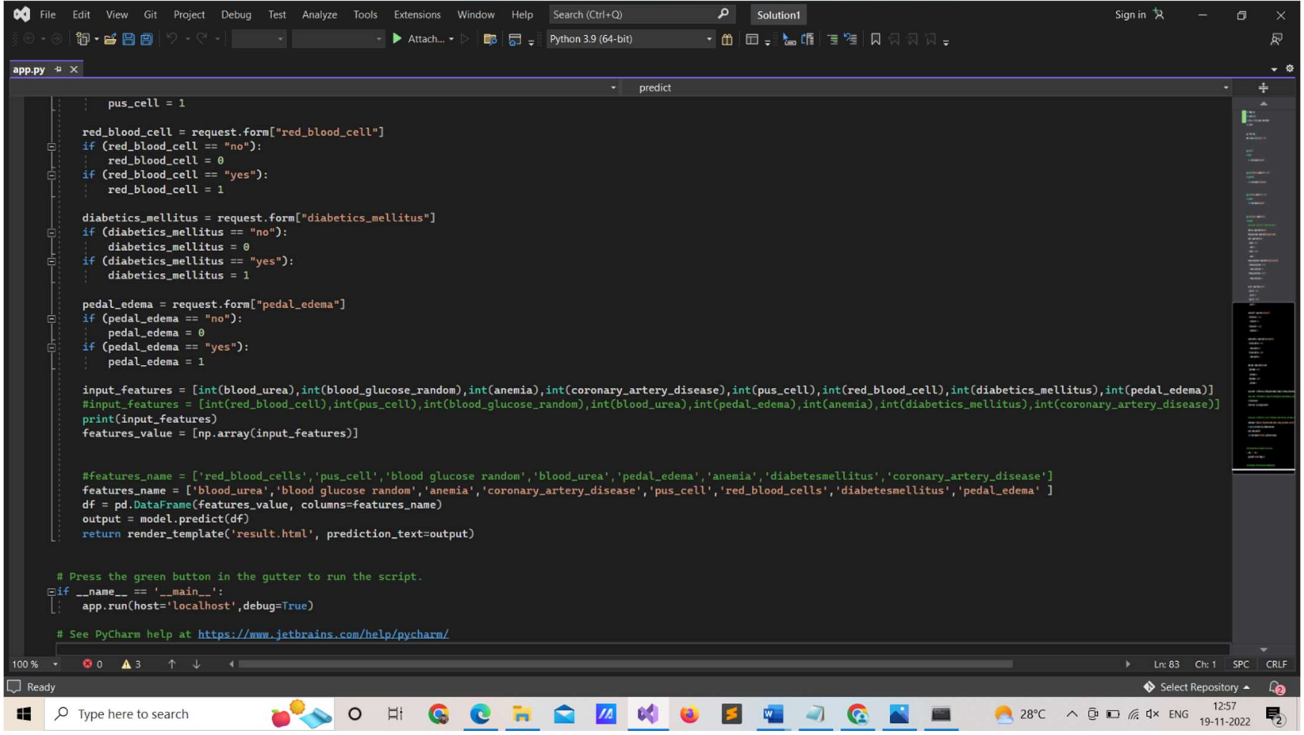
- ❖ Valid Input - identified closes of valid input must be accepted.
- ❖ Invalid Input - identified classes of invalid input must be rejected.
- ❖ Functions - identified functions must be exercise.
- ❖ Output - identified classes of application outputs mint be exercised.

Systems/Procedures: Interfacing systems or procedures must be invoked**.**

8.2 User Acceptance Testing



```python
import numpy as np
import pandas as pd
from flask import Flask, request, render_template
import pickle

app = Flask(__name__)
model = pickle.load(open('CKD.pkl', 'rb'))


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


@app.route('/Prediction', methods=['POST', 'GET'])
def prediction():
    return render_template('indexnew.html')


@app.route('/Home', methods=['POST', 'GET'])
def my_home():
    return render_template('home.html')


@app.route('/predict', methods=['POST'])
def predict():
    #input_features = ([int(x) for x in request.form.values()])
    blood_urea = request.form["blood_urea"]
    blood_glucose_random = request.form["blood_glucose_random"]
    anemia = request.form["Anemia"]
    if (anemia == "no"):
        anemia = 0
    if (anemia == "yes"):
        anemia = 1
    coronary_artery_disease = request.form["coronary_artery_disease"]
    if (coronary_artery_disease == "no"):
        coronary_artery_disease = 0
    if(coronary_artery_disease == "yes"):
        coronary_artery_disease = 1
```



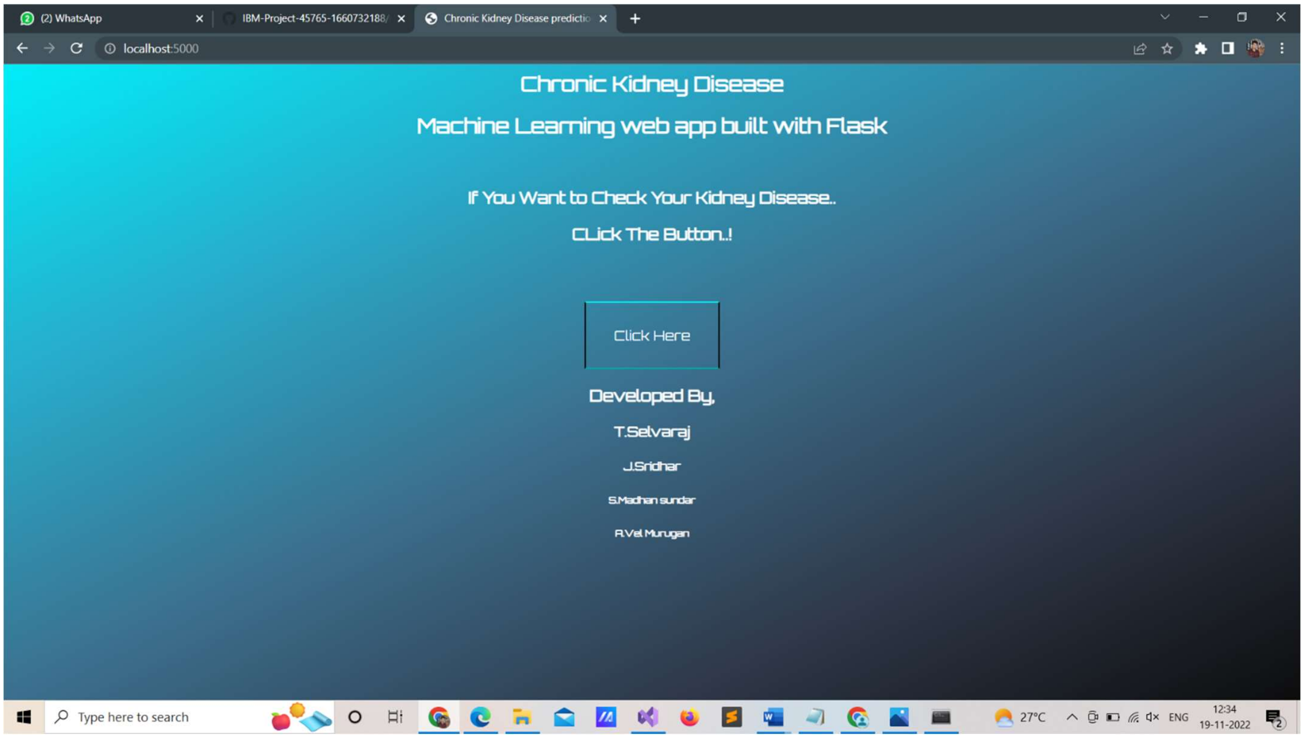
```python
        pus_cell = 1

    red_blood_cell = request.form["red_blood_cell"]
    if (red_blood_cell == "no"):
        red_blood_cell = 0
    if (red_blood_cell == "yes"):
        red_blood_cell = 1

    diabetics_mellitus = request.form["diabetics_mellitus"]
    if (diabetics_mellitus == "no"):
        diabetics_mellitus = 0
    if (diabetics_mellitus == "yes"):
        diabetics_mellitus = 1

    pedal_edema = request.form["pedal_edema"]
    if (pedal_edema == "no"):
        pedal_edema = 0
    if (pedal_edema == "yes"):
        pedal_edema = 1

    input_features = [int(blood_urea),int(blood_glucose_random),int(anemia),int(coronary_artery_disease),int(pus_cell),int(red_blood_cell),int(diabetics_mellitus),int(pedal_edema)]
    #input_features = [int(red_blood_cell),int(pus_cell),int(blood_glucose_random),int(blood_urea),int(pedal_edema),int(anemia),int(diabetics_mellitus),int(coronary_artery_disease)]
    print(input_features)
    features_value = [np.array(input_features)]


    #features_name = ['red_blood_cells','pus_cell','blood glucose random','blood_urea','pedal_edema','anemia','diabetesmellitus','coronary_artery_disease']
    features_name = ['blood_urea','blood glucose random','anemia','coronary_artery_disease','pus_cell','red_blood_cells','diabetesmellitus','pedal_edema' ]
    df = pd.DataFrame(features_value, columns=features_name)
    output = model.predict(df)
    return render_template('result.html', prediction_text=output)


# Press the green button in the gutter to run the script.
if __name__ == '__main__':
    app.run(host='localhost',debug=True)

# See PyCharm help at https://www.jetbrains.com/help/pycharm/
```

CHAPTER 9

RESULT

9.1 PERFORMANCE MATRICE

Result 1 normal not Chronic Kidney Disease



Chronic Kidney Disease

Machine Learning web app built with Flask

If You Want to Check Your Kidney Disease..

CLick The Button..!

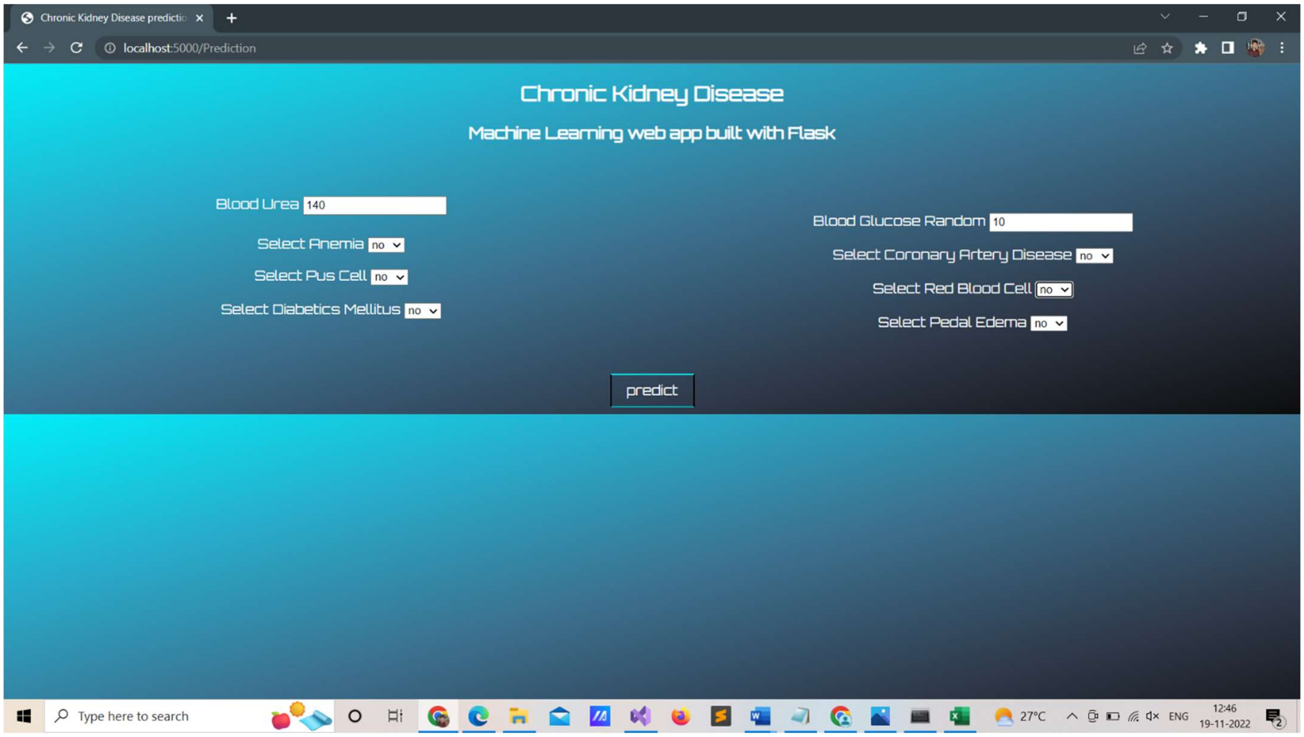
Developed By,

T.Selvaraj

J.Sridhar

S.Madhan sundar

R.Vel Murugan



Chronic Kidney Disease

Machine Learning web app built with Flask

Blood Urea 140

Select Anemia no

Select Pus Cell no

Select Diabetics Mellitus no

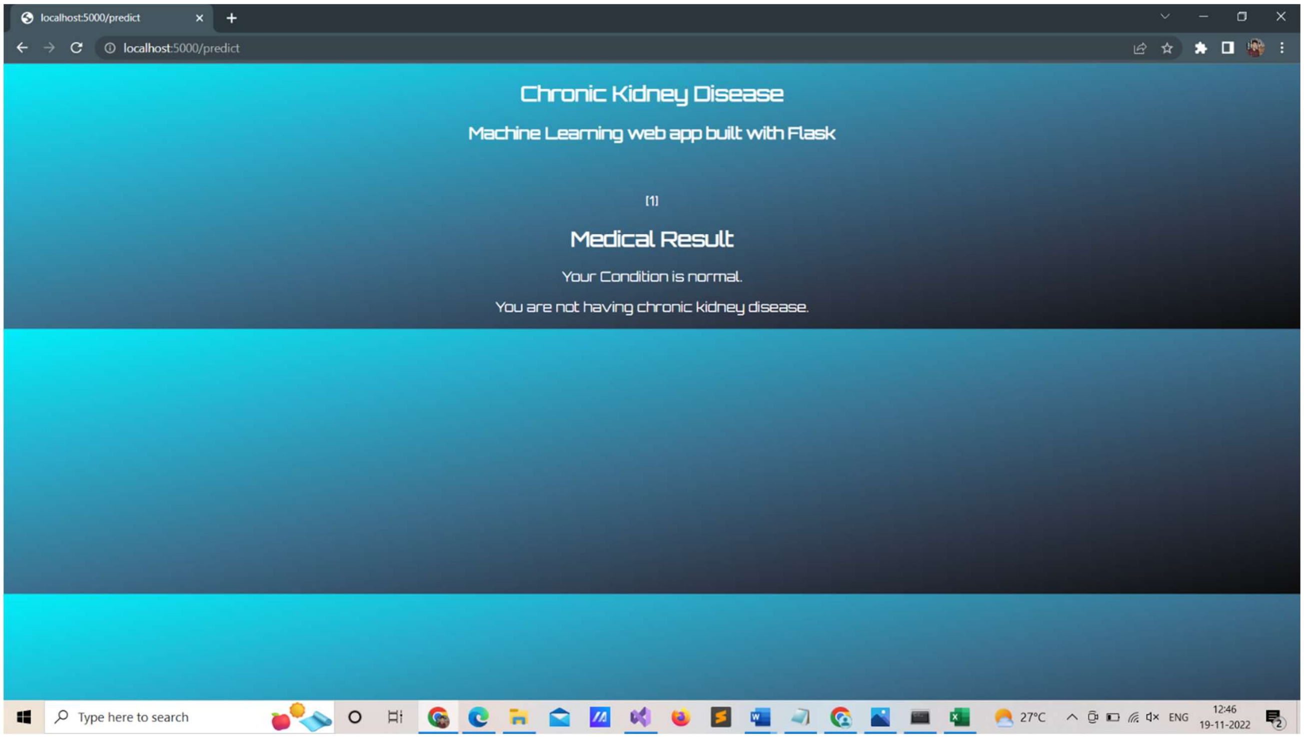
Blood Glucose Random 10

Select Coronary Artery Disease no

Select Red Blood Cell no

Select Pedal Edema no

predict



Chronic Kidney Disease

Machine Learning web app built with Flask

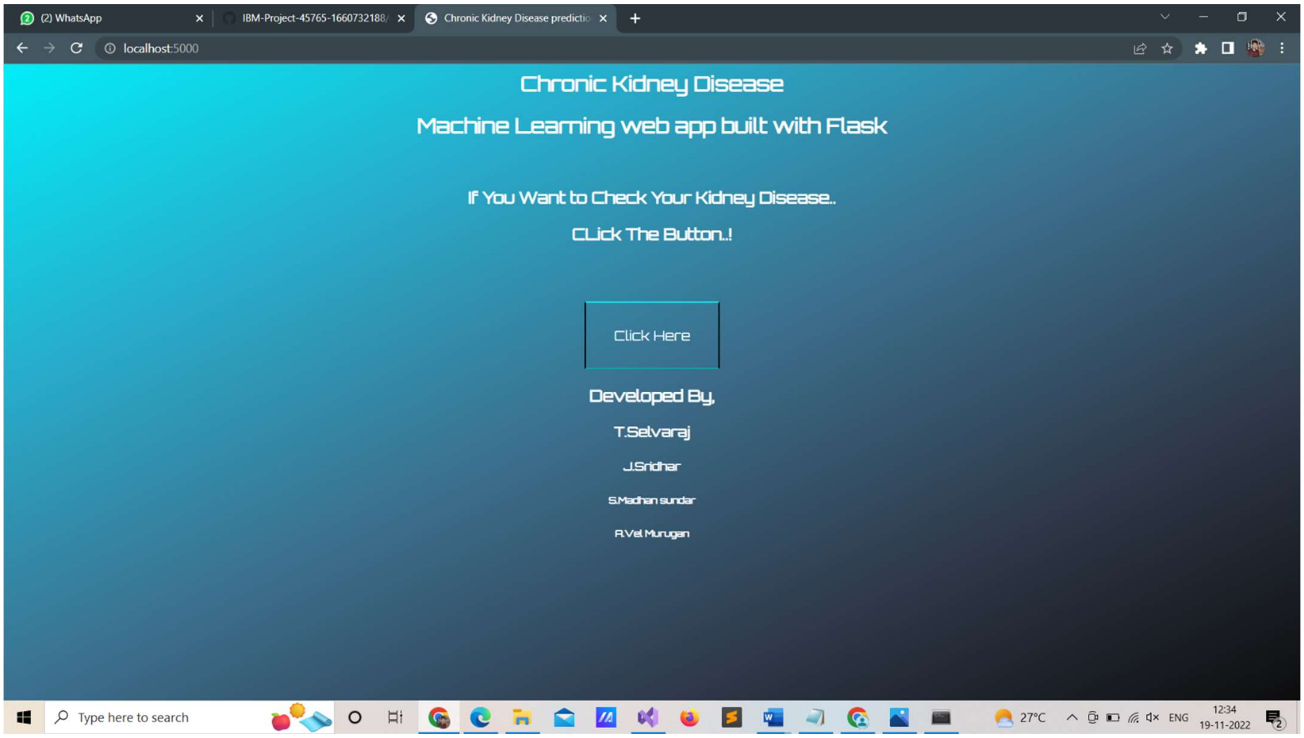
[1]

Medical Result

Your Condition is normal.

You are not having chronic kidney disease.

Result 2 abnormal Chronic Kidney Disease



(2) WhatsApp
IBM-Project-45765-1660732188/
Chronic Kidney Disease predictio
localhost:5000

Chronic Kidney Disease

Machine Learning web app built with Flask

If You Want to Check Your Kidney Disease..

CLick The Button..!

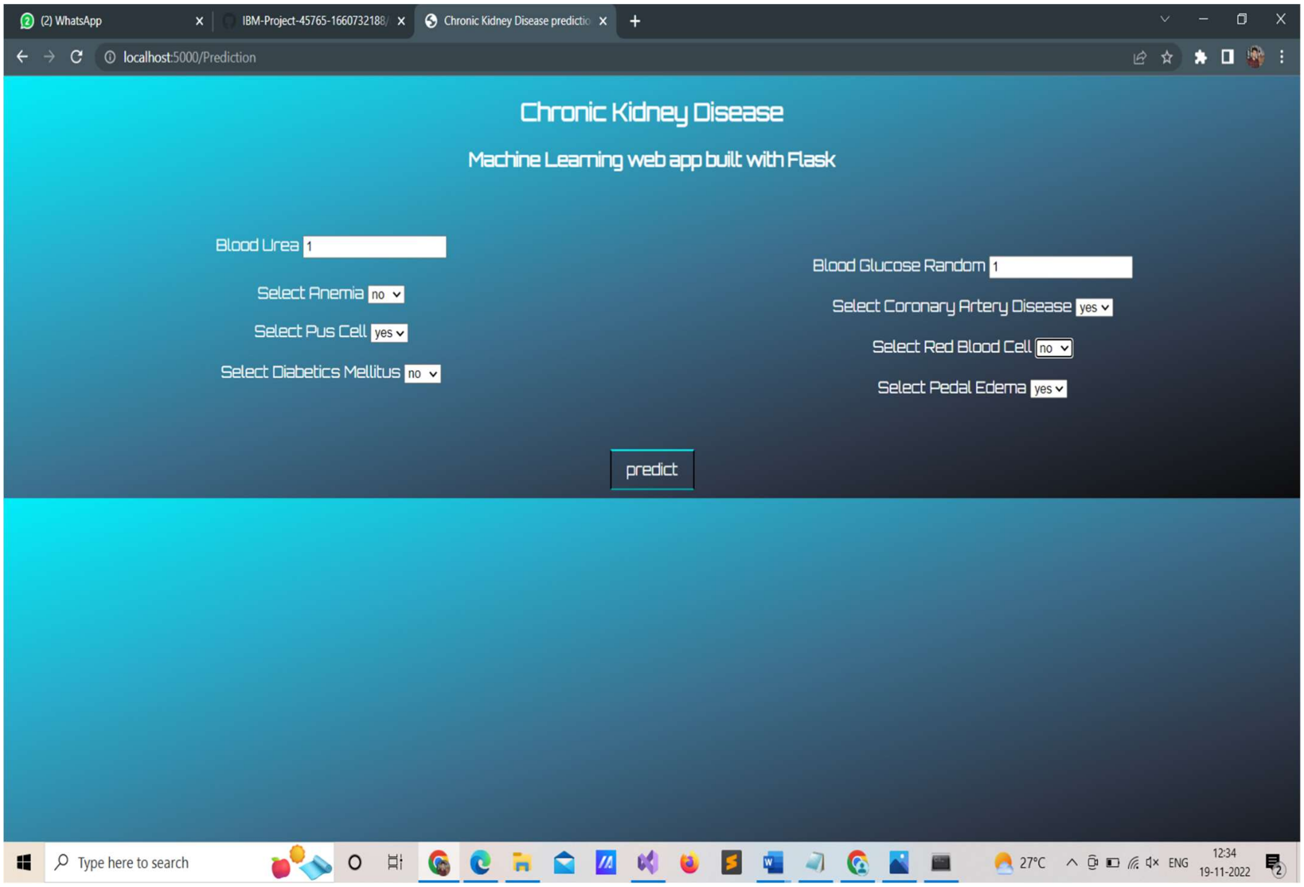
Developed By,

T.Selvaraj

J.Sridhar

S.Madhan sundar

R.Vel Murugan


(2) WhatsApp
IBM-Project-45765-1660732188/
Chronic Kidney Disease predictio
localhost:5000/Prediction
Type here to search
27°C
ENG
12:34
19-11-2022

Chronic Kidney Disease

Machine Learning web app built with Flask

Blood Urea 1

Blood Glucose Random 1

Select Anemia no

Select Coronary Artery Disease yes

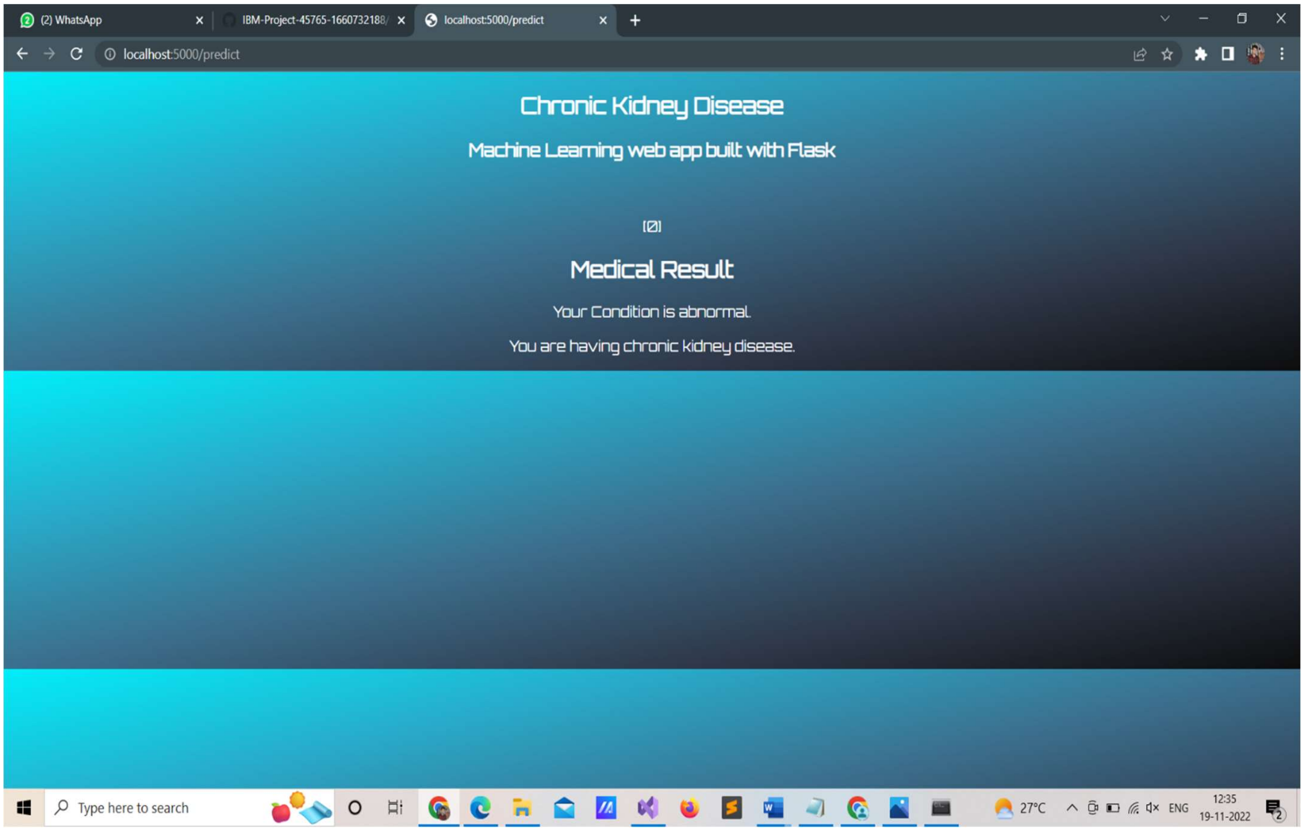
Select Pus Cell yes

Select Red Blood Cell no

Select Diabetics Mellitus no

Select Pedal Edema yes

predict



Chronic Kidney Disease

Machine Learning web app built with Flask

[0]

Medical Result

Your Condition is abnormal.

You are having chronic kidney disease.

CHAPTER 10

ADVANTAGES AND DISADVANTAGES

- This software has various advantages where it can be used as an expert guide to doctors for early detection of chronic kidney disease. It is also seen in performance metrics that it has an accuracy of 95% which gives good confidence to the users.

- Machine Learning is autonomous but highly susceptible to errors. Suppose you train an algorithm with data sets small enough to not be inclusive. You end up with biased predictions coming from a biased training set.

- Easy for small parameter multiple processing unis are connected each other very effective and efficient error less.

- No need to separate linear and non-linear of records fast to train the model.

- Independent between attributes it is very fast algorithm.

- Its complex the time is long to train sometime overlap slow learning.

- It requires more memory space slow to test depend on a number of dimensions sensitive to nosy.

- In small data set to reduce precision it requires very large.

CHAPTER 11

CONCLUSION:

This Project has helped team members to understand various concepts of Machine learning, Flask file, IBM cloud and Python notebook

This project can be scaled for usage in prediction of other chronic diseases which will help doctors in diagnosis of disease at an early stage thereby helping in early detection of various diseases.

This work examines the ability to detect CKD using machine learning algorithms while considering the least number of tests or features. We approach this aim by applying four machine learning classifiers: logistic regression, SVM, random forest, and gradient boosting on a small dataset of 400 records. In order to reduce the number of features and remove redundancy, the association between variables have been studied. A filter feature selection method has been applied to the remaining attributes and found that there are haemoglobin, albumin, and specific gravity have the most impact to predict the CKD.

The classifiers have been trained, tested, and validated using 10-fold cross-validation. Higher performance was achieved with the gradient boosting algorithm by F1-measure (99.1 %), sensitivity (98.8%), and specificity (99.3%). This result is the highest among previous studies with less number of features and hence less cost. Therefore, we conclude that CKD can be detected with only three features. Also, we found that hemoglobin has the highest contribution in detecting CKD, whereas albumin has the lowest using RF and GB models.

CHAPTER 12

FUTURE SCOPE:

- This software can be used to detect various other chronic diseases by modifying the dataset and the user inputs received. The model can be further trained with enormous amounts of data to improve the accuracy.

- This would help detect the chances of a person having CKD further on in his life which would be really helpful and cost-effective people.

- This model could be integrated with normal blood report generation, which could automatically flag out if there is person at risk.

- Patients would not have so go to a doctor unless they are flagged by the algorithms. This would make it cheaper and easier for the modern busy person.

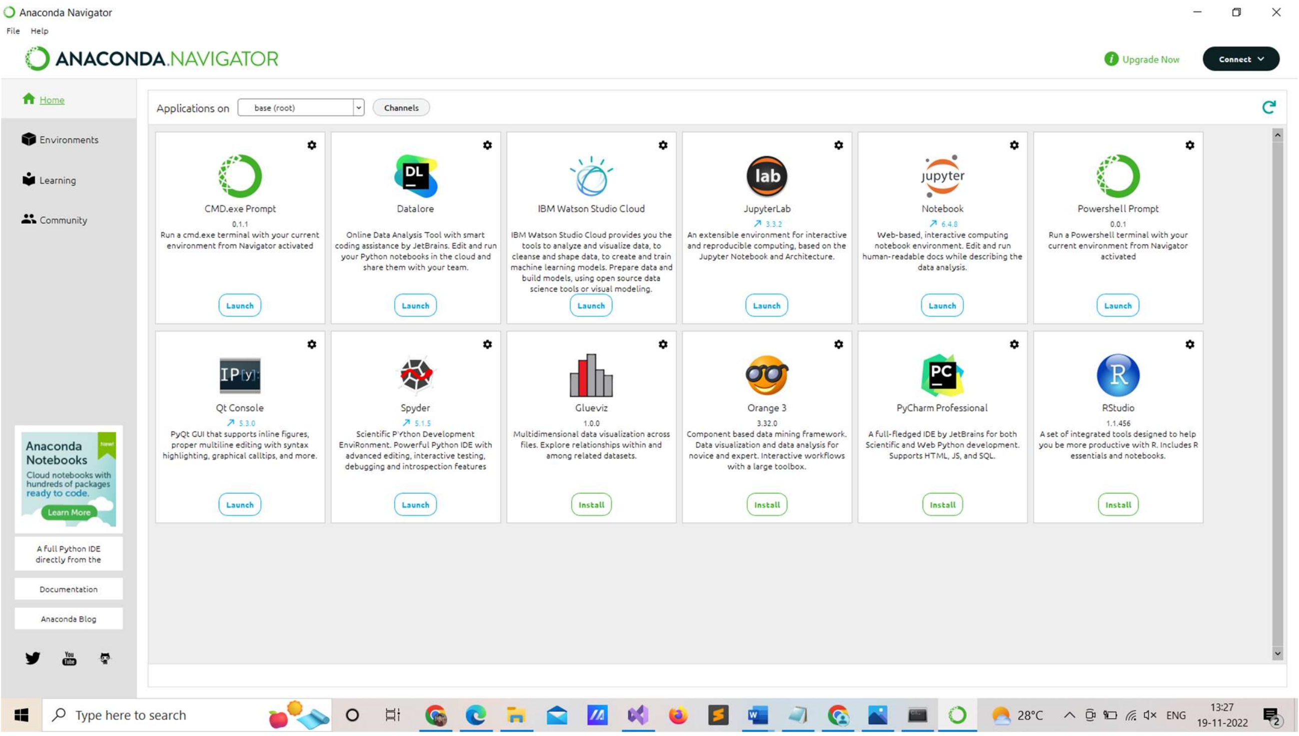
CHAPTER 13

APPENDIX

Source Code :

Python flask coding:

```
import numpy as np
import pandas as pd
from flask import Flask, request, render_template
import pickle

app = Flask(__name__)
model = pickle.load(open('CKD.pkl', 'rb'))


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


@app.route('/Prediction', methods=['POST', 'GET'])
def prediction():
    return render_template('indexnew.html')


@app.route('/Home', methods=['POST', 'GET'])
def my_home():
    return render_template('home.html')


@app.route('/predict', methods=['POST'])
def predict():
    #input_features = ([int(x) for x in request.form.values()])
    blood_urea = request.form["blood_urea"]
    blood_glucose_random = request.form["blood_glucose_random"]
    anemia = request.form["Anemia"]
    if (anemia == "no"):
        anemia = 0
    if (anemia == "yes"):
        anemia = 1
    coronary_artery_disease = request.form["coronary_artery_disease"]
    if (coronary_artery_disease == "no"):
        coronary_artery_disease = 0
    if(coronary_artery_disease == "yes"):
        coronary_artery_disease = 1

    pus_cell = request.form["pus_cell"]
    if (pus_cell == "no"):
        pus_cell = 0
    if (pus_cell == "yes"):
        pus_cell = 1

    red_blood_cell = request.form["red_blood_cell"]
    if (red_blood_cell == "no"):
        red_blood_cell = 0
    if (red_blood_cell == "yes"):
        red_blood_cell = 1

    diabetics_mellitus = request.form["diabetics_mellitus"]
    if (diabetics_mellitus == "no"):
        diabetics_mellitus = 0
    if (diabetics_mellitus == "yes"):
        diabetics_mellitus = 1

    pedal_edema = request.form["pedal_edema"]
    if (pedal_edema == "no"):
        pedal_edema = 0
    if (pedal_edema == "yes"):
        pedal_edema = 1

    input_features =
[int(blood_urea),int(blood_glucose_random),int(anemia),int(coronary_artery_disease),int(pus_
cell),int(red_blood_cell),int(diabetics_mellitus),int(pedal_edema)]
    #input_features =
[int(red_blood_cell),int(pus_cell),int(blood_glucose_random),int(blood_urea),int(pedal_edema
),int(anemia),int(diabetics_mellitus),int(coronary_artery_disease)]
    print(input_features)
    features_value = [np.array(input_features)]


    #features_name = ['red_blood_cells','pus_cell','blood glucose
random','blood_urea','pedal_edema','anemia','diabetesmellitus','coronary_artery_disease']
    features_name = ['blood_urea','blood glucose
random','anemia','coronary_artery_disease','pus_cell','red_blood_cells','diabetesmellitus','
pedal_edema' ]
    df = pd.DataFrame(features_value, columns=features_name)
    output = model.predict(df)
    return render_template('result.html', prediction_text=output)


# Press the green button in the gutter to run the script.
if __name__ == '__main__':
    app.run(host='localhost',debug=True)

# See PyCharm help at https://www.jetbrains.com/help/pycharm/
```



CMD.exe Prompt
0.1.1
Run a cmd.exe terminal with your current environment from Navigator activated
Launch

Datalore
Online Data Analysis Tool with smart coding assistance by JetBrains. Edit and run your Python notebooks in the cloud and share them with your team.
Launch

IBM Watson Studio Cloud
IBM Watson Studio Cloud provides you the tools to analyze and visualize data, to cleanse and shape data, to create and train machine learning models. Prepare data and build models, using open source data science tools or visual modeling.
Launch

JupyterLab
3.3.2
An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.
Launch

Notebook
6.4.8
Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.
Launch

Powershell Prompt
0.0.1
Run a Powershell terminal with your current environment from Navigator activated
Launch

Qt Console
5.3.0
PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.
Launch

Spyder
5.1.5
Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features
Launch

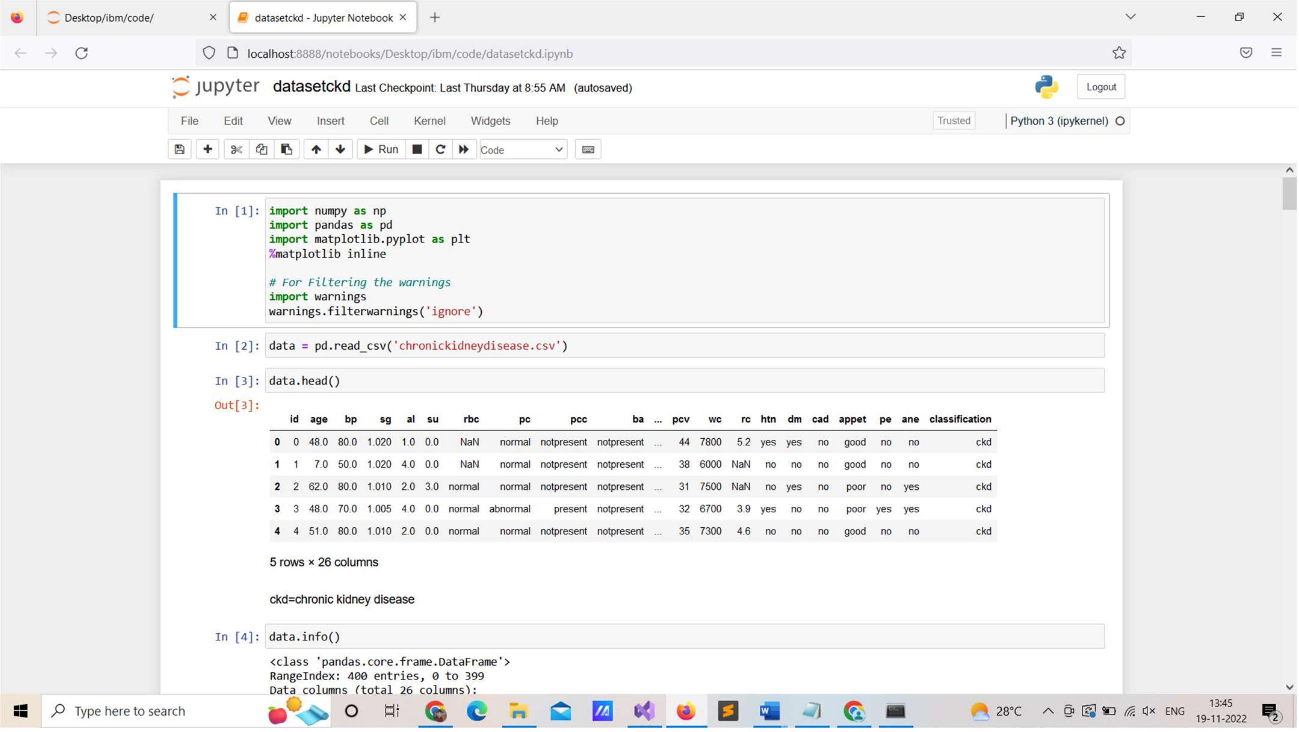
Glueviz
1.0.0
Multidimensional data visualization across files. Explore relationships within and among related datasets.
Install

Orange 3
3.32.0
Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.
Install

PyCharm Professional
A full-fledged IDE by JetBrains for both Scientific and Web Python development. Supports HTML, JS, and SQL.
Install

RStudio
1.1.456
A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.
Install

Fig: show the import in computer.


Desktop/ibm/code/
datasetckd - Jupyter Notebook
localhost:8888/notebooks/Desktop/ibm/code/datasetckd.ipynb
jupyter
datasetckd Last Checkpoint: Last Thursday at 8:55 AM (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3 (ipykernel)
Run
Code
Type here to search
28°C
ENG
13:45
19-11-2022

```
In [1]: import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        %matplotlib inline

        # For Filtering the warnings
        import warnings
        warnings.filterwarnings('ignore')

In [2]: data = pd.read_csv('chronickidneydisease.csv')

In [3]: data.head()
```

Out[3]:

	id	age	bp	sg	al	su	rbc	pc	pcc	ba	...	pcv	wc	rc	htn	dm	cad	appet	pe	ane	classification
0	0	48.0	80.0	1.020	1.0	0.0	NaN	normal	notpresent	notpresent	...	44	7800	5.2	yes	yes	no	good	no	no	ckd
1	1	7.0	50.0	1.020	4.0	0.0	NaN	normal	notpresent	notpresent	...	38	6000	NaN	no	no	no	good	no	no	ckd
2	2	62.0	80.0	1.010	2.0	3.0	normal	normal	notpresent	notpresent	...	31	7500	NaN	no	yes	no	poor	no	yes	ckd
3	3	48.0	70.0	1.005	4.0	0.0	normal	abnormal	present	notpresent	...	32	6700	3.9	yes	no	no	poor	yes	yes	ckd
4	4	51.0	80.0	1.010	2.0	0.0	normal	normal	notpresent	notpresent	...	35	7300	4.6	no	no	no	good	no	no	ckd

5 rows × 26 columns

ckd=chronic kidney disease

```
In [4]: data.info()

        <class 'pandas.core.frame.DataFrame'>
        RangeIndex: 400 entries, 0 to 399
        Data columns (total 26 columns):
```

Fig: show to read the data set.

GITHUB LINK:

https://github.com/IBM-EPBL/IBM-Project-45765-1660732188

DEMO LINK:

https://drive.google.com/file/d/14cs_auJkxs9NW8hbC0WmnZFCaNS2hol5/view?usp=drivesdk