# Project Report

# Analytics for Hospitals Health-care data

# INTRODUCTION

# CHAPTER-1
# INTRODUCTION

## 1.1 Project Overview

In this Project, we will be closely working with the hospitals health care data and for that, we will be looking into the health care dataset from that dataset we will derive various insights that help us know the weightage of each feature and how they are interrelated to each other but this time our sole aim is to detect the probability of person that will be visited and treatment or not.

Machine learning proves to be effective in assisting in making decisions and predictions from the large quantity of data produced by the health care industry. This project aims to predict future health care by analysing data of patients which classifies whether they have any disease or not using machine-learning algorithm. Machine Learning techniques can be a boon in this regard. Even though  disease can occur in different forms, there is a common set of core risk factors that influence whether someone will ultimately be at risk for severe disease or not. By collecting the data from various sources, classifying them under suitable headings & finally analysing to extract the desired data we can say that this technique can be very well adapted to do the analysis of health care data.

## 1.2 Purpose

The purpose of healthcare data is to save lives and improve the quality of life, so companies and governments are doing their best to offer new solutions. Artificial intelligence has enough capacities to store, process, and analyze vast volumes of information.

Collecting quality data from current patients means you can find similar potential customers and tweak your marketing campaigns and healthcare procedures accordingly. Better relations with your customers (in this case, your patients) is the key to success for healthcare facilities. Therefore, using electronic tools in the health care institution ensures safe and efficient data management. Therefore, it is important to establish appropriate medical data management systems for efficient health care delivery. Keywords: electronic medical data, health care data, medical data processing.

Data analytics in healthcare uses clinical and patient data to improve care, enhance patient outcomes, and make health business management more efficient. Your chosen healthcare consulting provider should specialize in health data analytics to maximize your potential non-labor cost savings.

# LITERATURE SURVEY

# CHAPTER-2
# LITERATURE SURVEY

## 2.1 Existing problem

Like Oxygen, the world is surrounded by data today. The quantity of data that we harvest and eat up is thriving aggressively in the digitized world. Increasing use of new innovations and social media generate vast amount of data that can earn splendid information if properly analyzed. This large dataset generally known as big data, do not fit in traditional databases because of its' rich size. Organizations need to manage and analyze big data for better decision making and outcomes. So, big data analytics is receiving a great deal of attention today. In healthcare, big data analytics has the possibility of advanced patient care and clinical decision support. In this paper, we review the background and the various methods of big data analytics in healthcare. This paper also elaborates various platforms and algorithms for big data analytics and discussion on its advantages and challenges. This survey winds up with a discussion of challenges and future directions.

## 2.2 References

[1] Alexandros Labrinidis and H. V. Jagadish, "Challenges and opportunities with big data," Proc. VLDB Endow. 5, pp. 2032-2033, August 2012.

[2] Aneeshkumar, A.S. and C.J. Venkateswaran, "Estimating the surveillance of liver disorder using classification algorithms". Int. J. Comput. Applic., 57: pp. 39-42, 2012.

[3] Amir Gandomi, Murtaza Haider, "Beyond the hype: Big data concepts, methods, and analytics," International Journal of Information Management 35, pp. 137-144, 2015.

[4] Chaitrali, S., D. Sulabha and S. Apte, "Improved study of heart disease prediction system using data mining classification techniques," Int. J. Comput. Applic. 47: 44-48, 2012.

[5] Doug Beaver, Sanjeev Kumar, Harry C. Li, Jason Sobel, Peter Vajgel, Facebook Inc, “Finding a Needle in Haystack: Facebook’s Photo Storage” 2010.

[6] Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C. Hsieh, Deborah A. Wallach, Mike Burrows, Tushar Chandra, Andrew Fikes, and Robert E. Gruber, “Bigtable: A Distributed Storage System for Structured Data,” ACM Trans. Comput. Syst. 26, 2, Article 4, June 2008.

[7] Giuseppe DeCandia, Deniz Hastorun, Madan Jampani, Gunavardhan Kakulapati, Avinash Lakshman, Alex Pilchin, Swaminathan Sivasubramanian, Peter Vosshall, and Werner Vogels, “Dynamo: amazon's highly available key-value store,” In Proceedings of twenty-first ACM SIGOPS symposium on Operating systems principles (SOSP '07). ACM, New York, NY, USA, 205-220.

[8] Hsi-Jen et al. “A retrospective analysis of prognostic indicators in dental implant therapy using the C5.0 decision tree algorithm”, Journal of Dental Sciences, Volume 8, Issue 3 , 248-255, 2013.

[9] I.A.T. Hashem, et al, “The rise of “big data” on cloud computing: Review and open research issues,” Information Systems, 2014.

[10]Jakrarin Therdphapiyanak, Krerk Piromsopa, “An analysis of suitable parameters for efficiently applying K-means clustering to large TCPdump data set using Hadoop framework,” In Electrical Engineering/ Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2013 10th International Conference, pp. 1-6, May 2013.

[11]Jason Brownlee, “Machine Learning Foundations, Master the definitions and concepts”, Machine Learning Mastery, 2011.

[12]Jawei Han and Micheline Kamber, “Data Mining: Concepts and Techniques,” Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2000.

[13]L. Hall, N. Chawla, and K. Bowyer, "Decision tree learning on very large data sets," in International Conference on Systems, Man and Cybernetics, pp. 2579-2584, IEEE Oct 1998.

[14]Mark A. Beyer, Douglas Laney, “The importance of ‘Big Data’: A Definition,” Gartner, retrieved on 21 June 2012.

[15]Nilima Patil and Rekha Lathi, “Comparison of C5.0 and CART Classification algorithms use pruning technique”, 2012.

[16]Patil D.V, Prof. Dr. R. S. Bichkar, "A Hybrid Evolutionary Approach To Construct Optimal Decision Trees with Large Data Sets", IEEE, 2006.

[17]Rajesh, K. and S. Anand, “Analysis of SEER dataset for breast cancer diagnosis using C4.5 classification algorithm,” Int. J. Adv. Res. Comput. Commun. Eng., 1: 72-77, 2012.

[18]Ramalingam, V.V., S.G. Kumar and V. Sugumaran, “Analysis of EEG signals using data mining approach,” Int. J. Comput. Eng. Technol., 3: 206-212, 2012.

## 2.3 Problem Statement Definition

The major challenge in health care is its analysis. There are instruments available which can analyse health care data but either it are expensive or are not efficient to calculate chance of disease in human. Early detection of health care data can decrease the mortality rate and overall complications. However, it is not possible to monitor patients every day in all cases accurately and consultation of a patient for 24 hours by a doctor is not available since it requires more sapience, time and expertise. Since we have a good amount of data in today's world, we can use various machine learning algorithms to analyse the data for hidden patterns. The hidden patterns can be used for health diagnosis in medicinal data.

Between overwhelming hospitalization rates, intensifying cybersecurity threats, and an aggravating number of mental illnesses due to strict lockdown measures, hospitals are desperately searching for help. Big data in healthcare seems like a viable solution.

# IDEATION & PROPOSED SOLUTION

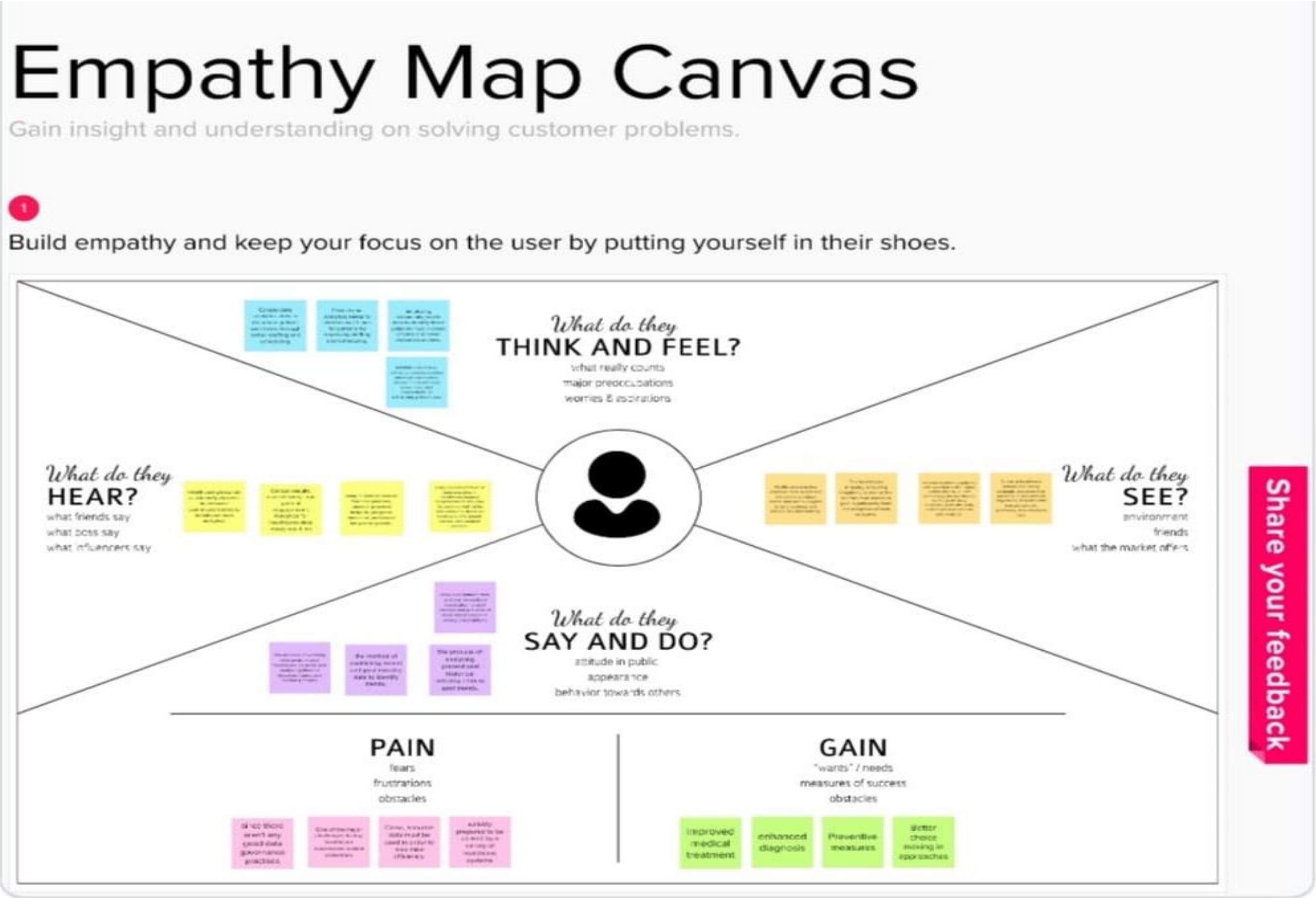
# CHAPTER-3
# IDEATION & PROPOSED SOLUTION

## 3.1 Empathy Map Canvas

An empathy map canvas is a more in-depth version of the original empathy map, which helps identify and describe the user's needs and pain points. And this is valuable information for improving the user experience. Teams rely on user insights to map out what is important to their target audience, what influences them, and how they present themselves.

This information is then used to create personas that help teams visualize users and empathize with them as individuals, rather than just as a vague marketing demographic or account number. An empathy map canvas helps brands provide a better experience for users by helping teams understand the perspectives and mindset of their customers. Using a template to create an empathy map canvas reduces the preparation time and standardizes the process so you create empathy map canvases of similar quality.

## Empathy Map Canvas Visualizing and Predicting Heart Diseases with an Interactive Dashboard:



Empathy Map Canvas

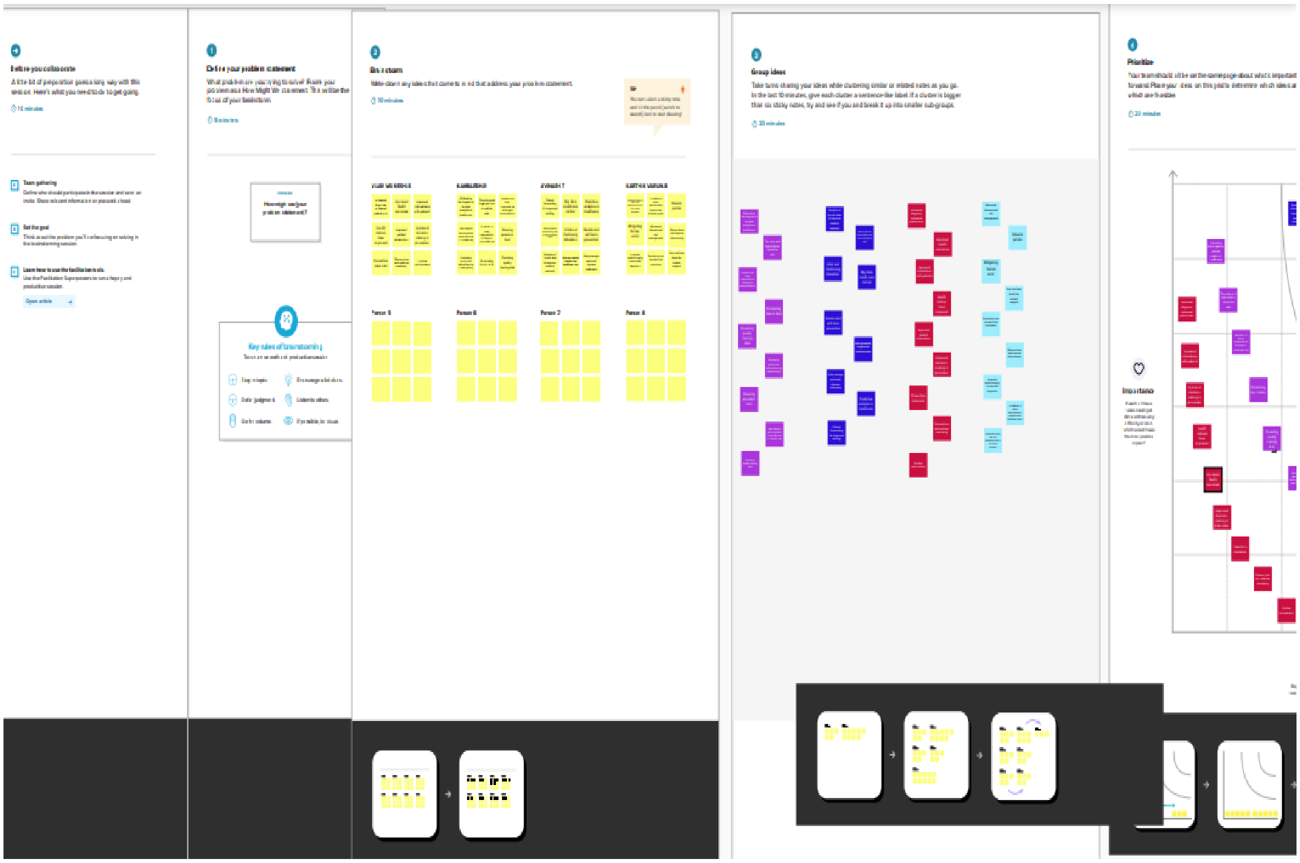
Gain insight and understanding on solving customer problems.

1

Build empathy and keep your focus on the user by putting yourself in their shoes.

## 3.2 Ideation & Brainstorming

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich number of creative solutions.



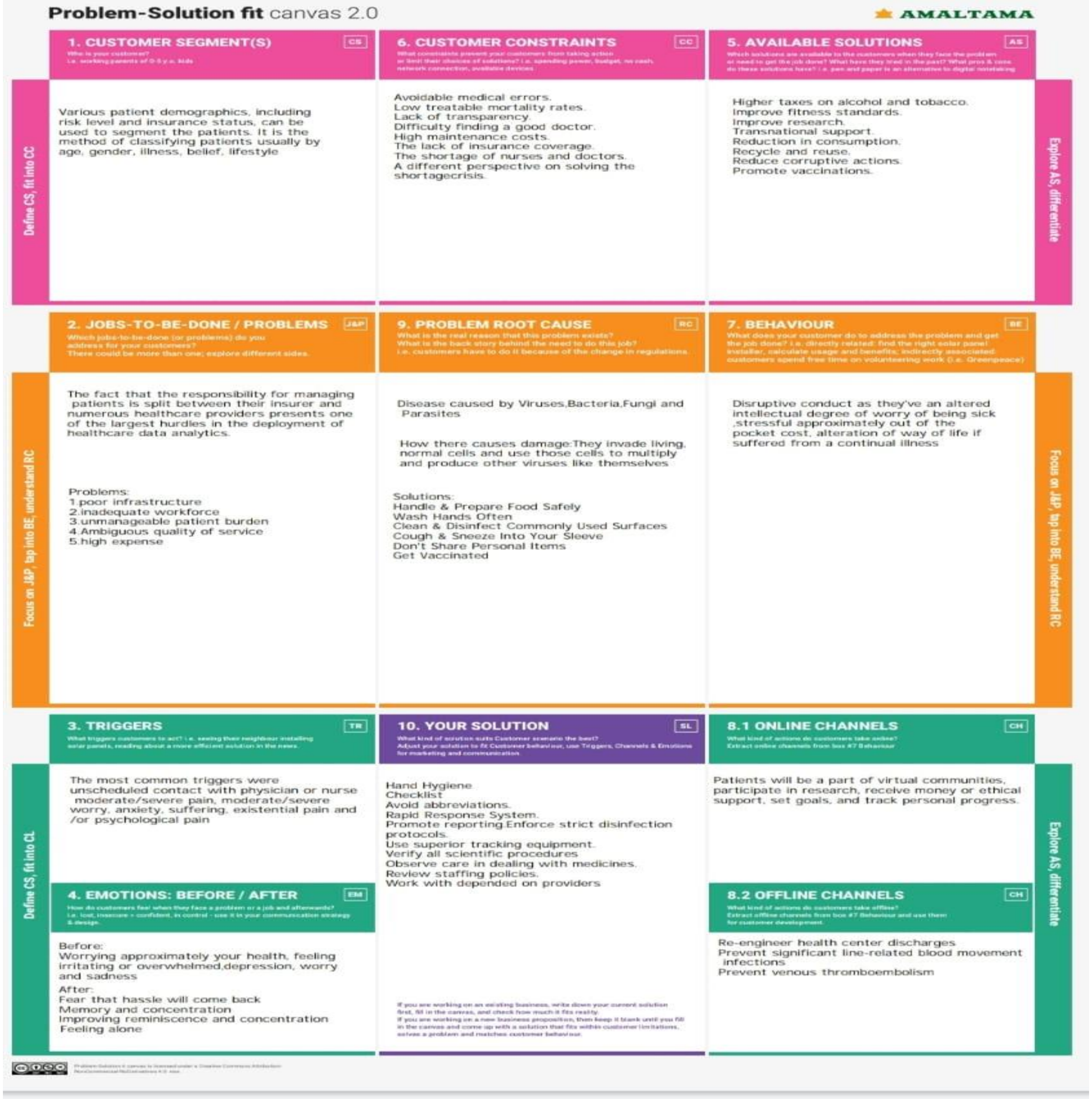
# 3.3 Proposed Solution

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

| S.No. | Parameter | Description |
|---|---|---|
| 1. | Problem Statement (Problem to be solved) | EHR data matched patient-reported data in 23.5 percent of records in a study at an ophthalmology practise. Patients' EHR data did not agree in any way when they reported having three or more eye health complaints. |
| 2. | Idea / Solution description | Predictive analytics can create patient journey dashboards and disease trajectories that can lead to effective, and result-driven healthcare. It improves treatment delivery, cuts costs, improves efficiencies, and so on. |
| 3. | Novelty / Uniqueness | Healthcare data frequently resides in several locations. from various departments, such as radiology or pharmacy, to various source systems, such as EMRs or HR software. The organisation as a whole contributes to the data. This data becomes accessible and usable when it is combined into a single, central system, such as an enterprise data warehouse (EDW). |
| 4. | Social Impact / Customer Satisfaction | Enhanced diagnosis<br>Improved medical treatment<br>Improved health results<br>Improved relationships with patients<br>More positive health indicators |
| 5. | Business Model (Revenue Model) | The two factors that have the biggest negative effects on hospital income are claim denials and patient incapacity to pay their part. 90% more uncollectible claim denials were written off by hospitals and healthcare systems in 2017 compared to the preceding six years. |
| 6. | Scalability of the Solution | A variety of institutions must store, evaluate, and take action on the massive amounts of data being produced by the health care sector as it expands quickly. India is a vast, culturally varied nation with a sizable population that is increasingly able to access centralised healthcare services. |

## 3.4 Problem Solution fit

The Problem-Solution 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** canvas 2.0

AMALTAMA

Define CS, fit into CC

**1. CUSTOMER SEGMENT(S)** CS

Who is your customer?
i.e. working parents of 0-5 y.o. kids

Various patient demographics, including risk level and insurance status, can be used to segment the patients. It is the method of classifying patients usually by age, gender, illness, belief, lifestyle

**6. CUSTOMER CONSTRAINTS** CC

What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices

Avoidable medical errors.
Low treatable mortality rates.
Lack of transparency.
Difficulty finding a good doctor.
High maintenance costs.
The lack of insurance coverage.
The shortage of nurses and doctors.
A different perspective on solving the shortagecrisis.

**5. AVAILABLE SOLUTIONS** AS

Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking

Higher taxes on alcohol and tobacco.
Improve fitness standards.
Improve research.
Transnational support.
Reduction in consumption.
Recycle and reuse.
Reduce corruptive actions.
Promote vaccinations.

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

**2. JOBS-TO-BE-DONE / PROBLEMS** J&P

Which jobs-to-be-done (or problems) do you address for your customers?
There could be more than one; explore different sides.

The fact that the responsibility for managing patients is split between their insurer and numerous healthcare providers presents one of the largest hurdles in the deployment of healthcare data analytics.

Problems:
1.poor infrastructure
2.inadequate workforce
3.unmanageable patient burden
4.Ambiguous quality of service
5.high expense

**9. PROBLEM ROOT CAUSE** RC

What is the real reason that this problem exists?
What is the back story behind the need to do this job?
i.e. customers have to do it because of the change in regulations.

Disease caused by Viruses,Bacteria,Fungi and Parasites

How there causes damage:They invade living, normal cells and use those cells to multiply and produce other viruses like themselves

Solutions:
Handle & Prepare Food Safely
Wash Hands Often
Clean & Disinfect Commonly Used Surfaces
Cough & Sneeze Into Your Sleeve
Don't Share Personal Items
Get Vaccinated

**7. BEHAVIOUR** BE

What does your customer do to address the problem and get the job done? i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

Disruptive conduct as they've an altered intellectual degree of worry of being sick ,stressful approximately out of the pocket cost, alteration of way of life if suffered from a continual illness

Focus on J&P, tap into BE, understand RC

Define CS, fit into CL

**3. TRIGGERS** TR

What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.

The most common triggers were unscheduled contact with physician or nurse moderate/severe pain, moderate/severe worry, anxiety, suffering, existential pain and /or psychological pain

**4. EMOTIONS: BEFORE / AFTER** EM

How do customers feel when they face a problem or a job and afterwards?
i.e. lost, insecure > confident, in control - use it in your communication strategy & design.

Before:
Worrying approximately your health, feeling irritating or overwhelmed,depression, worry and sadness

After:
Fear that hassle will come back
Memory and concentration
Improving reminiscence and concentration
Feeling alone

**10. YOUR SOLUTION** SL

What kind of solution suits Customer scenario the best?
Adjust your solution to fit Customer behaviour, use Triggers, Channels & Emotions for marketing and communication

Hand Hygiene
Checklist
Avoid abbreviations.
Rapid Response System.
Promote reporting.Enforce strict disinfection protocols.
Use superior tracking equipment.
Verify all scientific procedures
Observe care in dealing with medicines.
Review staffing policies.
Work with depended on providers

If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality.
If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.

**8.1 ONLINE CHANNELS** CH

What kind of actions do customers take online?
Extract online channels from box #7 Behaviour

Patients will be a part of virtual communities, participate in research, receive money or ethical support, set goals, and track personal progress.

**8.2 OFFLINE CHANNELS** CH

What kind of actions do customers take offline?
Extract offline channels from box #7 Behaviour and use them for customer development.

Re-engineer health center discharges
Prevent significant line-related blood movement infections
Prevent venous thromboembolism

Explore AS, differentiate

# REQUIREMENT ANALYSIS

# CHAPTER-4

# REQUIREMENT ANALYSIS

## 4.1 Functional requirement

Following are the functional requirements of the proposed solution.

| FR No. | Functional Requirement (Epic) | Sub Requirement (Story / Sub-Task) |
|---|---|---|
| FR-1 | User SignUp | SignUp through Form SignUp through Gmail |
| FR-2 | Credential Confirmation | Confirmation via Email Confirmation via OTP |
| FR-3 | User Login | Login through Form |
| FR-4 | Forgot password | OTP via email |
| FR-5 | Data collection | The majority of hospitals in the United States today have electronic health records, which are the focus of hospital data analytics. A comprehensive record that contains all relevant information about the patient's health is known as a digital health record. |

## 4.2 Non-Functional requirements

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

| FR No. | Non-Functional Requirement | Description |
|---|---|---|
| NFR-1 | **Reliability** | A random sample of 10% of the medical records was examined independently by two reviewers to ascertain inter-rater reliability. We applied a straightforward computer-based random sample technique to choose these medical records. |
| NFR-2 | **Maintainability** | Based on four layers, the Maintainability Information Database (MID) is organised. The database has all the project data pertaining to maintenance planning, and this data is integrated with the BIM models of the project for improved project aspect integration. |
| NFR-3 | **Performance** | People value their health more than the majority of other products and services. Both governments and people spend a lot of money on healthcare. People want to choose their healthcare with knowledge. |

# PROJECT DESIGN

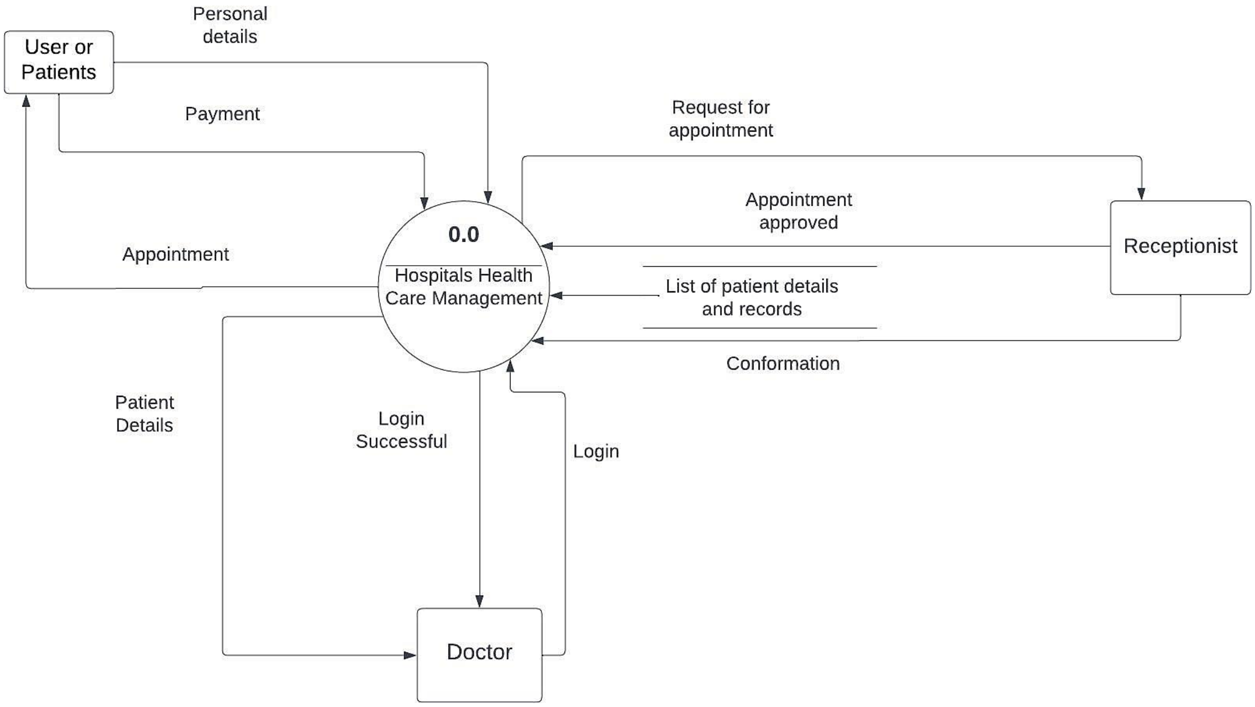
# CHAPTER-5
# PROJECT DESIGN

## 5.1 Data Flow Diagrams

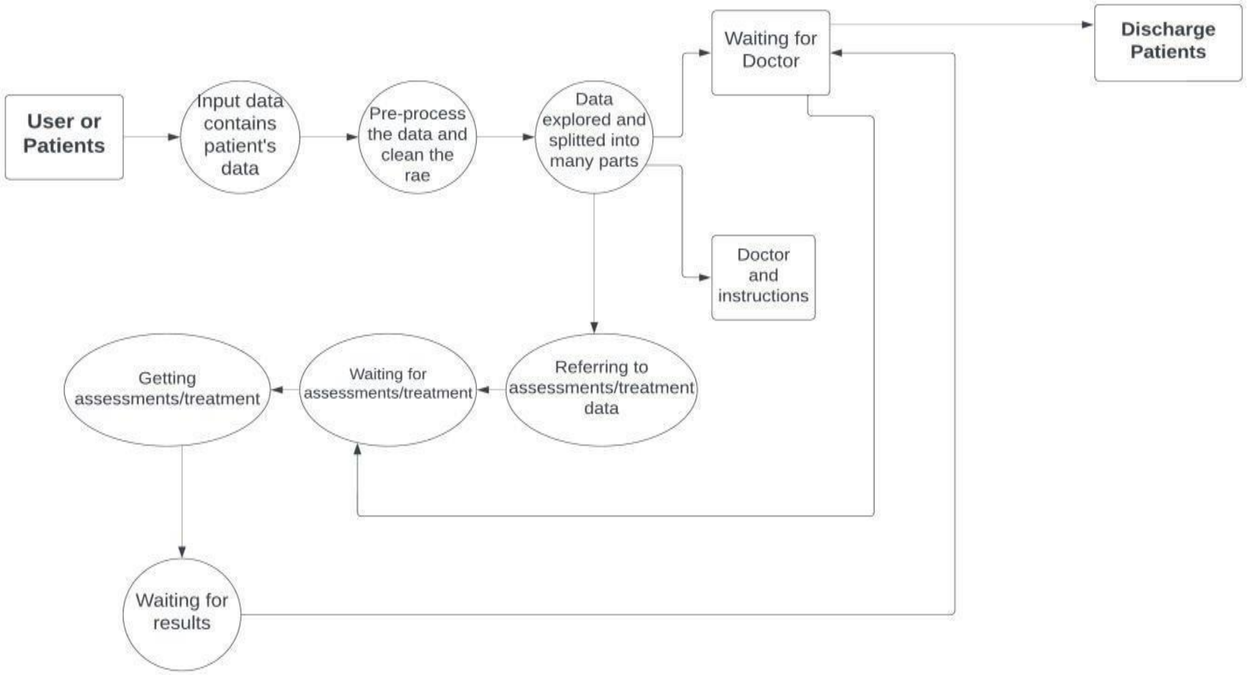
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

### Data Flow Diagram for Heart Disease Prediction Dashboard:

**DFD LEVEL 0:**



# Flow:


User or Patients
Input data contains patient's data
Pre-process the data and clean the rae
Data explored and splitted into many parts
Waiting for Doctor
Discharge Patients
Doctor and instructions
Getting assessments/treatment
Waiting for assessments/treatment
Referring to assessments/treatment data
Waiting for results

1) User creates an account in the application.

2) User enters the medical records in the dashboard.

3) User can view the visualizations of trends in the form of graphs and charts for his/her medical records with the trained dataset.

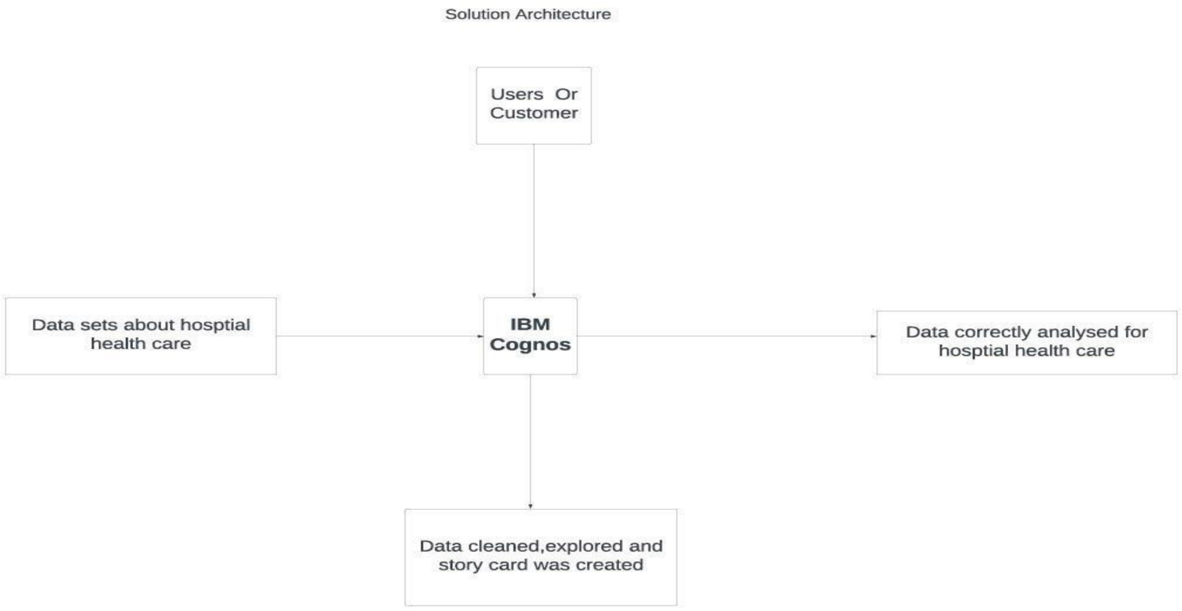
4) User can view the accuracy of probability of occurrence of heart disease in the dashboard.

## 5.2 Solution & Technical 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, behaviour, 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.

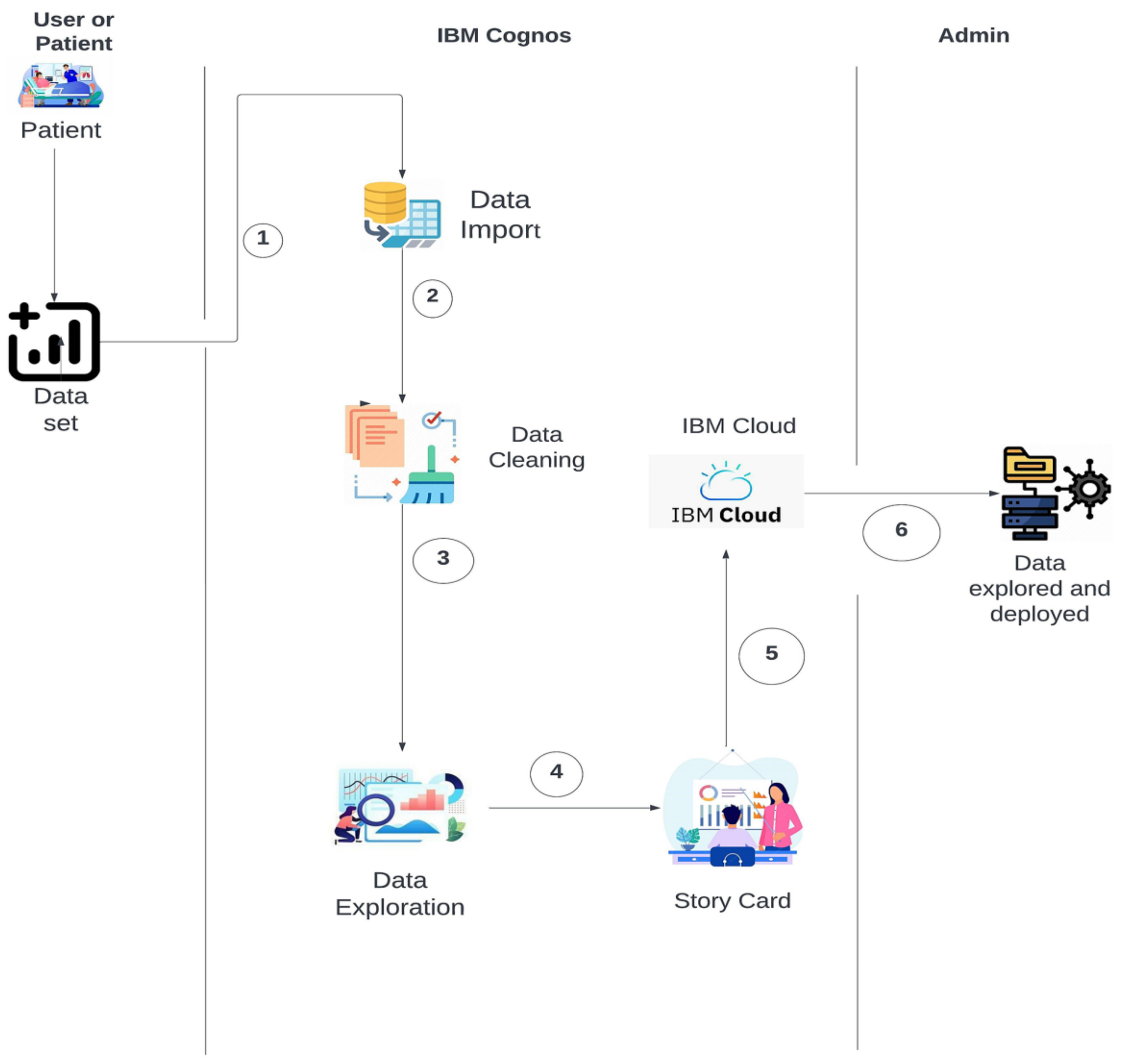
**Solution Architecture Diagram:**


Solution Architecture
Users Or Customer
Data sets about hosptial health care
IBM Cognos
Data correctly analysed for hosptial health care
Data cleaned,explored and story card was created

# Technology Stack (Architecture & Stack):

## Technical Architecture:

## Analysis of Hospitals Health-Care data:



:

**Table-1 : Components & Technologies:**

| **S.No** | **Component** | **Description** | **Technology** |
|---|---|---|---|
| 1. | User Interface | How user interacts with application e.g. Web UI, Mobile App, Chatbot etc. | IBM Cognos / Python . |
| 2. | Data Set | The data set prepared for hospitals health care | Python . |
| 3. | IBM Cognos | Data analytics platform | IBM Watson service |
| 4. | Data Import | Data set is imported in IBM cognos | IBM Watson Assistant |
| 5. | Data Cleaning | Data is cleaned by using some mathematical techniques such as mean,mode etc.to clean the null and missing data. | IBM Assistant |
| 6. | Data Exploration | Cleaned data can be explored. | IBM Cognos |
| 7. | Story Card | Data is explored and story card was prepared for visual representation | IBM Cognos |
| 8. | IBM Cloud | Storage of data | IBM DB2 |
| 9. | Data Explored and Deployed | Purpose of External API to explored and deployed | Data deployed to user by UI |
| 10. | Admin | Purpose of Data set model | Recognition of data set model etc. |

**Table-2: Application Characteristics:**

| S.No | Characteristics | Description | Technology |
|---|---|---|---|
| 1. | Open-Source | Open source model is used for the data set | Python |
| 2. | Security Implementations | Security for our data set | SHA 256, SHA 1 |
| 3. | Scalable Architecture | health care service utilizes the relational patient<br>data and big data analytics to tailor the medication recommendations | Python |
| 4. | Availability | The availability of technology used in data analytics | Python-Anaconda distribution and jupyter notebook is available and open source application |
| 5. | Performance | The performance of the application and its efficiency | Python and other languages is that Python is usually interpreted. Interpreted languages tend to perform worse than compiled languages, each command takes up a greater number of machine instructions . |

# 5.3 User Stories

**User Stories**:

| User Type | Functional Requirement (Epic) | User Story Number | User Story / Task | Acceptance criteria | Priority | Release |
|---|---|---|---|---|---|---|
| Customer (Patient) | Registration | USN-1 | As a user, I can gather the details of the patients. | I can access datasets my account Kaggle | High | Sprint-1 |
| | | | As an Analyst, I will check the data set andclean the dataset to create an efficient model. | I can clean the datasets using Cognos Analytics | High | Sprint-1 |
| | | USN-3 | As a user, I can register through Gmail | | Medium | Sprint-1 |
| | Login | USN-4 | As a user, I can log in by entering email & password | | High | Sprint-1 |
| | Forgot Password | USN-5 | As a user, if i forgot my password, by clickina foraot | By entering the OTP sent Ia email. | High | Sprint-1 |
| | | | email, | | | |
| | Data collection | USN-6 | As a user, I can upload the input data set in IBM Cognos | | High | Sprint-1 |

# PROJECT PLANNING & SCHEDULING

# CHAPTER-6

# PROJECT PLANNING & SCHEDULING

## 6.1 Sprint Planning & Estimation Product Backlog, Sprint Schedule, and Estimation:

Use the below template to create product backlog and sprint schedule

| User Type | Sprint | Functional Requirement (Epic) | User Story Number | User Story / Task | Story Points | Priority | Team Members |
|---|---|---|---|---|---|---|---|
| Customer (Patient) | Sprint-1 | Datasets | USN-1 | As a user, I can enter the details of the patients working in our organisation of the detail | 2 | High | S.Vijay Vignesh |
| Analyst | Sprint-1 | | USN-2 | As a Analyst, I will check the data set and clean the dataset to create an efficient model . | 1 | High | B.Karhik Varun |
| | Sprint-1 | | USN-3 | As an Analyst I will also correct the raw data and create a data module | 2 | Medium | R.Kamalesh |
| | Sprint-2 | Cleaning, Exploring data and creating model | USN-4 | As an Analyst I can create a Exploratory data analysis to identify the important factors of patient data set | 2 | High | B.Karhik Varun |
| | Sprint-2 | | USN-5 | As a Data analyst, I create a predicted model by also preparing story | 1 | High | T.Avinash |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | card with using explored data | | | |
| | Sprint-3 | Data Prediction | USN-6 | As a Data analyst, I will create different types of models in explored data to identify suitable | 5 | High | R.Kamalesh |
| | | | | model with effectively and efficiently | | | |
| Admin | Sprint-4 | Creation of deployed data UI | USN-7 | As an Analyst, I will import my analysed model into suitable framework | 2 | High | T.Avinash |
| | Sprint-4 | | USN-8 | | 5 | High | R.Kamalesh |

## 6.2 Sprint Delivery Schedule

### Project Tracker, Velocity & Burndown Chart:

| 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 | 15 | 5 Days | 24 Oct 2022 | 29 Oct 2022 | 15 | 29 Oct 2022 |
| Sprint-2 | 15 | 5 Days | 31 Oct 2022 | 05 Nov 2022 | 15 | 05 Nov 2022 |
| Sprint-3 | 15 | 5 Days | 07 Nov 2022 | 12 Nov 2022 | 15 | 12 Nov 2022 |
| Sprint-4 | 15 | 5 Days | 14 Nov 2022 | 19 Nov 2022 | 15 | 19 Nov 2022 |

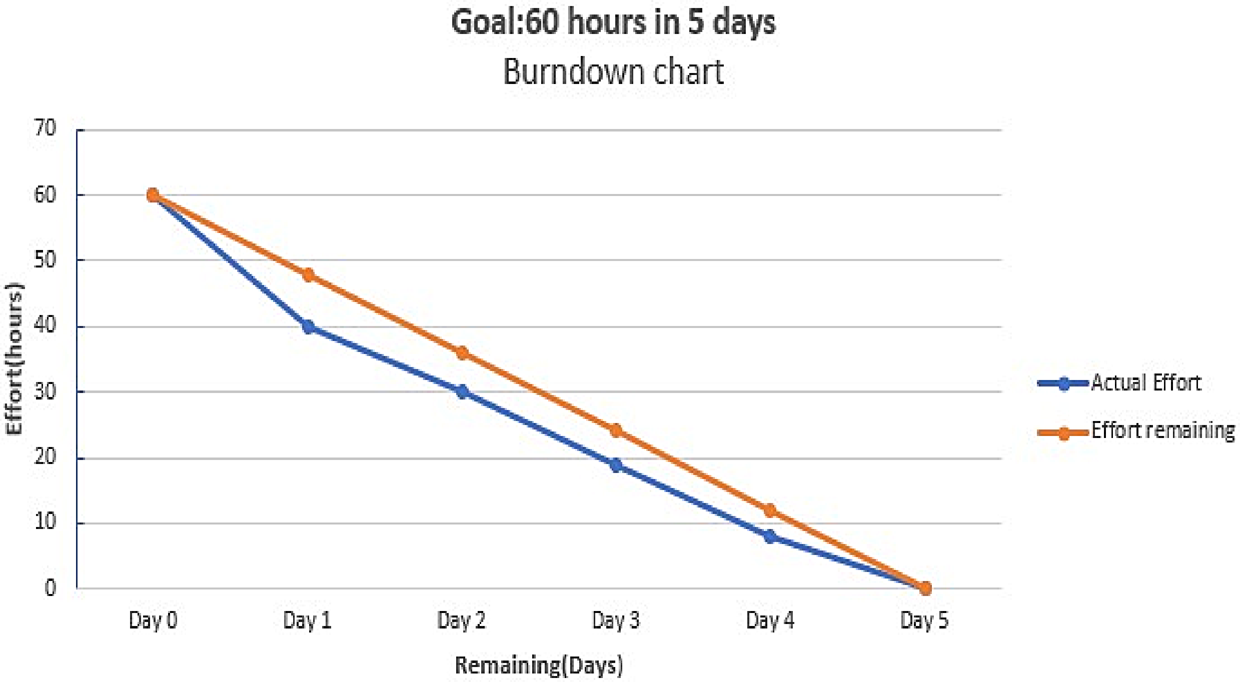
### Velocity:

We have a 5-day sprint duration, and the velocity of the team is 15 (points per sprint). The team’s average velocity (AV) per iteration unit (story points per day)

**Actual Velocity = Sprint Duration/Velocity = 15/5 = 3**

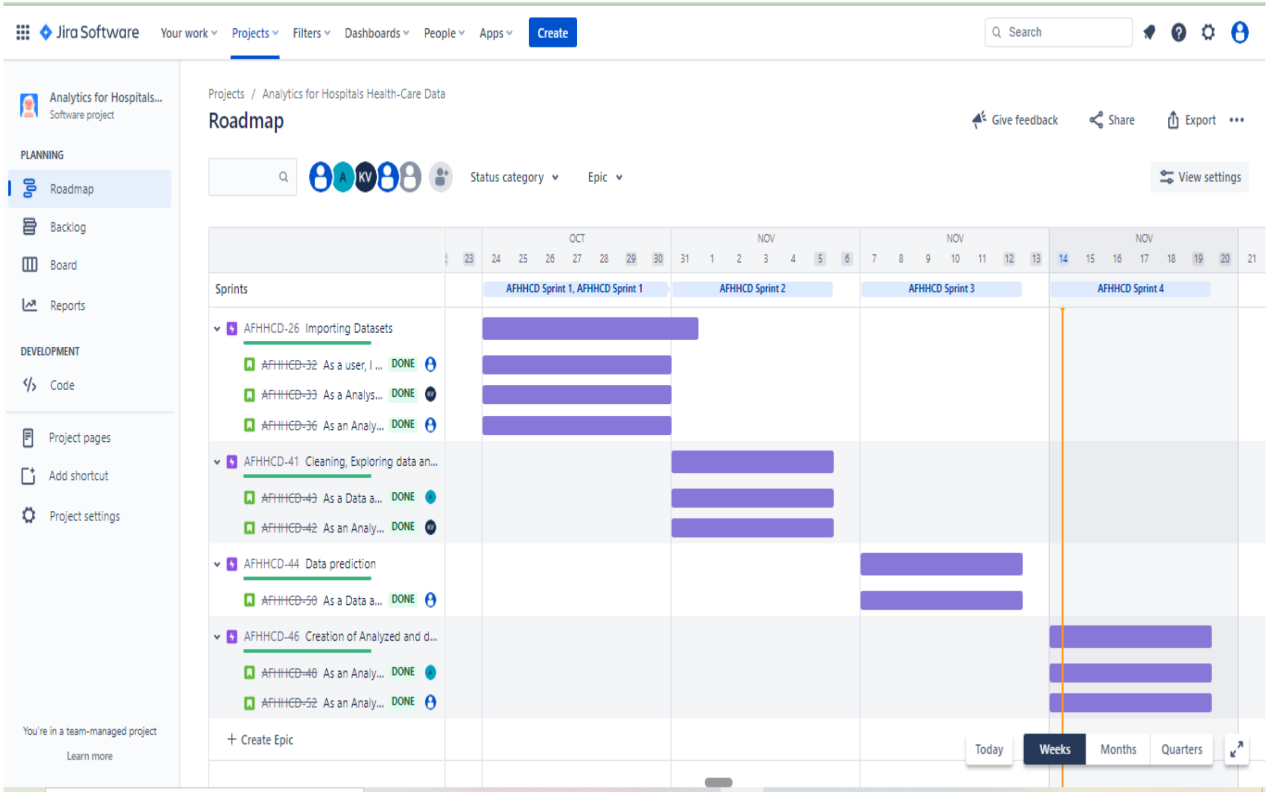
## Burndown Chart:

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.



## 6.3 Reports from JIRA

Jira helps teams plan, assign, track, report, and manage work and brings teams together for everything from agile software development and customer support to start-ups and enterprises. Software teams build better with Jira Software, the #1 tool for agile teams. As a Jira administrator, you can create project categories so your team can view work across related projects in one place. Your team can use categories in advanced search, filters, reports, and more.



Projects / Analytics for Hospitals Health-Care Data

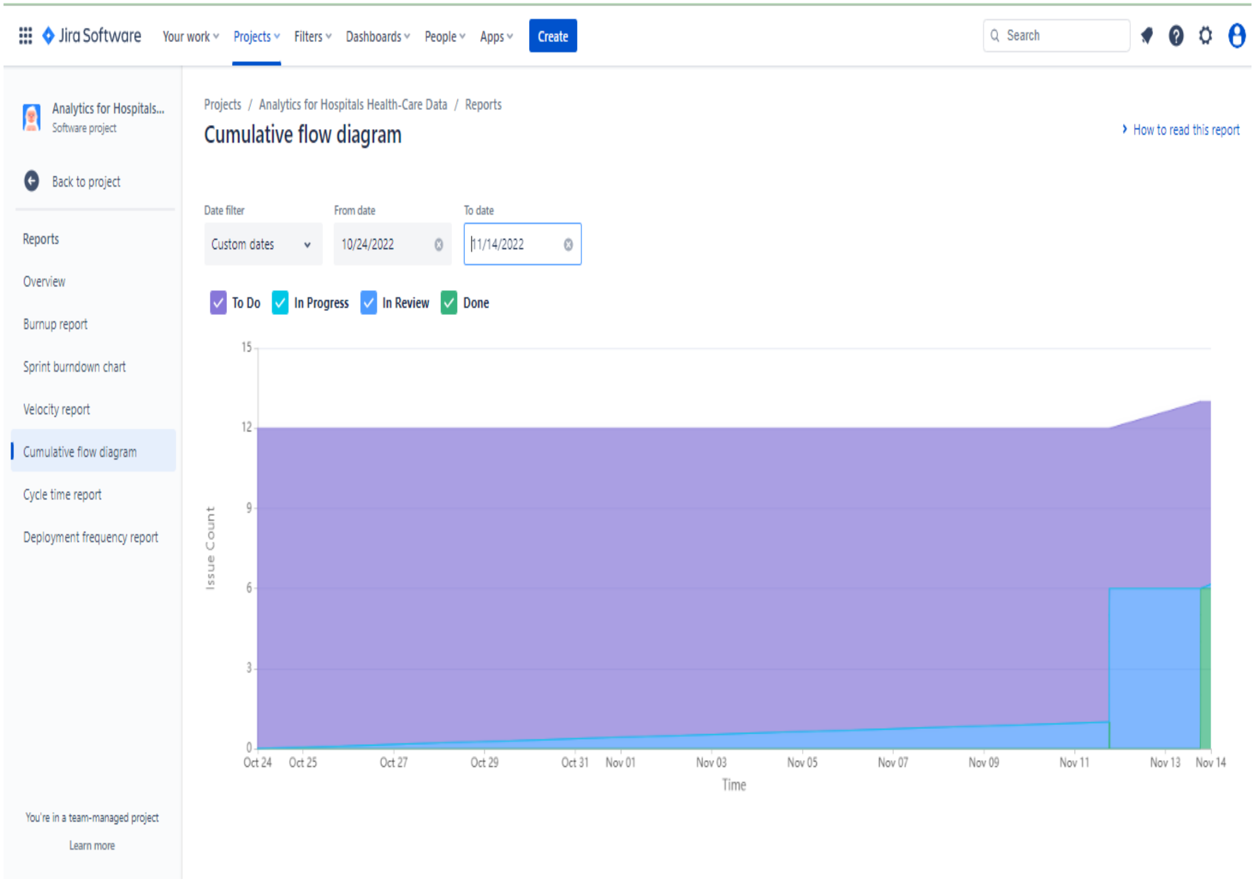
Roadmap

| | OCT | | | | | | | | NOV | | | | | | | NOV | | | | | | | NOV | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| Sprints | | AFHHCD Sprint 1, AFHHCD Sprint 1 | | | | | | | AFHHCD Sprint 2 | | | | | | | AFHHCD Sprint 3 | | | | | | | AFHHCD Sprint 4 | | | | | | | |

- AFHHCD-26 Importing Datasets
  - ~~AFHHCD-32~~ As a user, I ... DONE
  - ~~AFHHCD-33~~ As a Analys... DONE
  - ~~AFHHCD-36~~ As an Analy... DONE
- AFHHCD-41 Cleaning, Exploring data an...
  - ~~AFHHCD-43~~ As a Data a... DONE
  - ~~AFHHCD-42~~ As an Analy... DONE
- AFHHCD-44 Data prediction
  - ~~AFHHCD-50~~ As a Data a... DONE
- AFHHCD-46 Creation of Analyzed and d...
  - ~~AFHHCD-40~~ As an Analy... DONE
  - ~~AFHHCD-52~~ As an Analy... DONE

+ Create Epic

# CUMULATIVE JIRA :



Projects / Analytics for Hospitals Health-Care Data / Reports

Cumulative flow diagram

Date filter: Custom dates

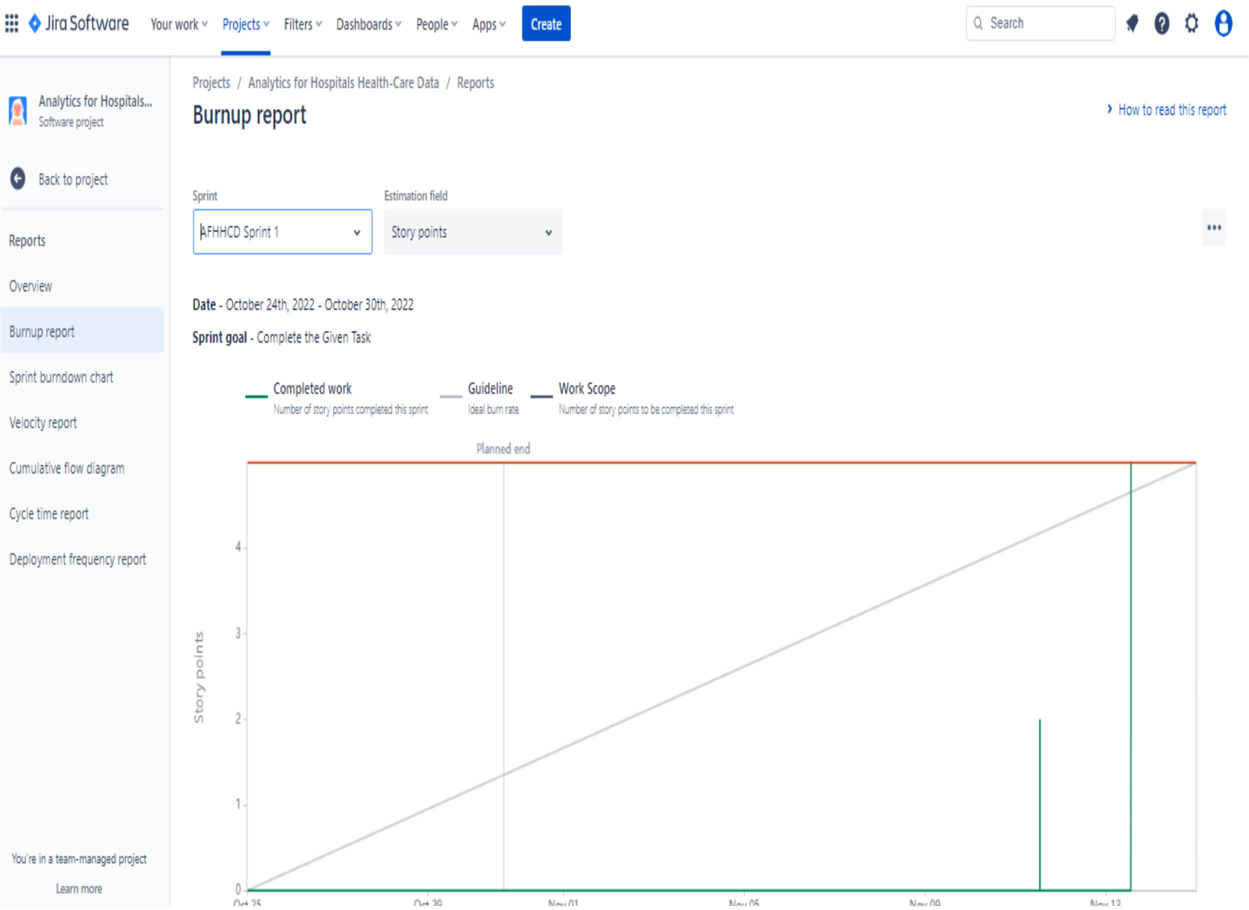
From date: 10/24/2022

To date: 11/14/2022

☑ To Do ☑ In Progress ☑ In Review ☑ Done

# BURNUP REPORT:

# SPRINT-1:



Burnup report

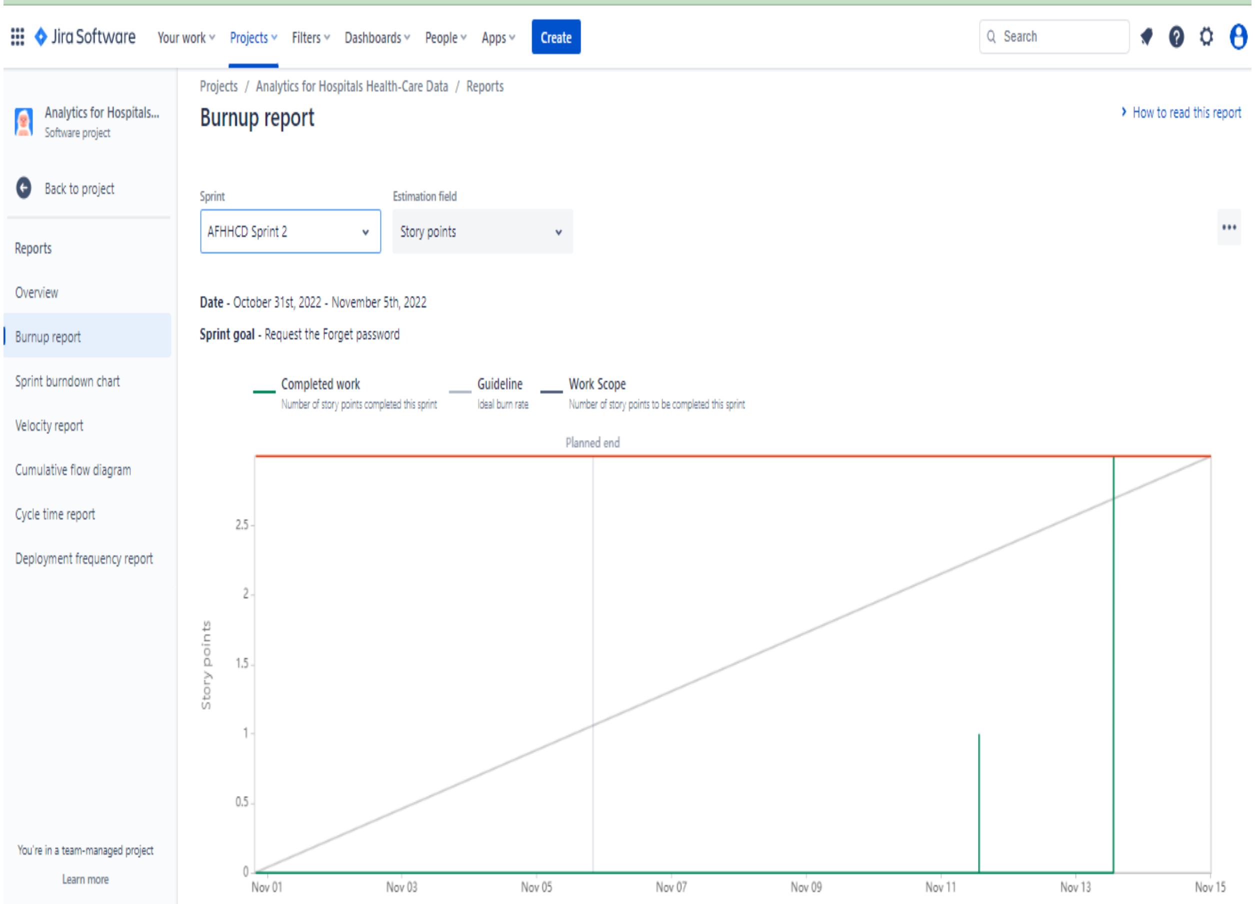
Sprint: AFHHCD Sprint 1

Estimation field: Story points

**Date** - October 24th, 2022 - October 30th, 2022

**Sprint goal** - Complete the Given Task

# SPRINT-2:



Projects / Analytics for Hospitals Health-Care Data / Reports

Burnup report

Sprint: AFHHCD Sprint 2

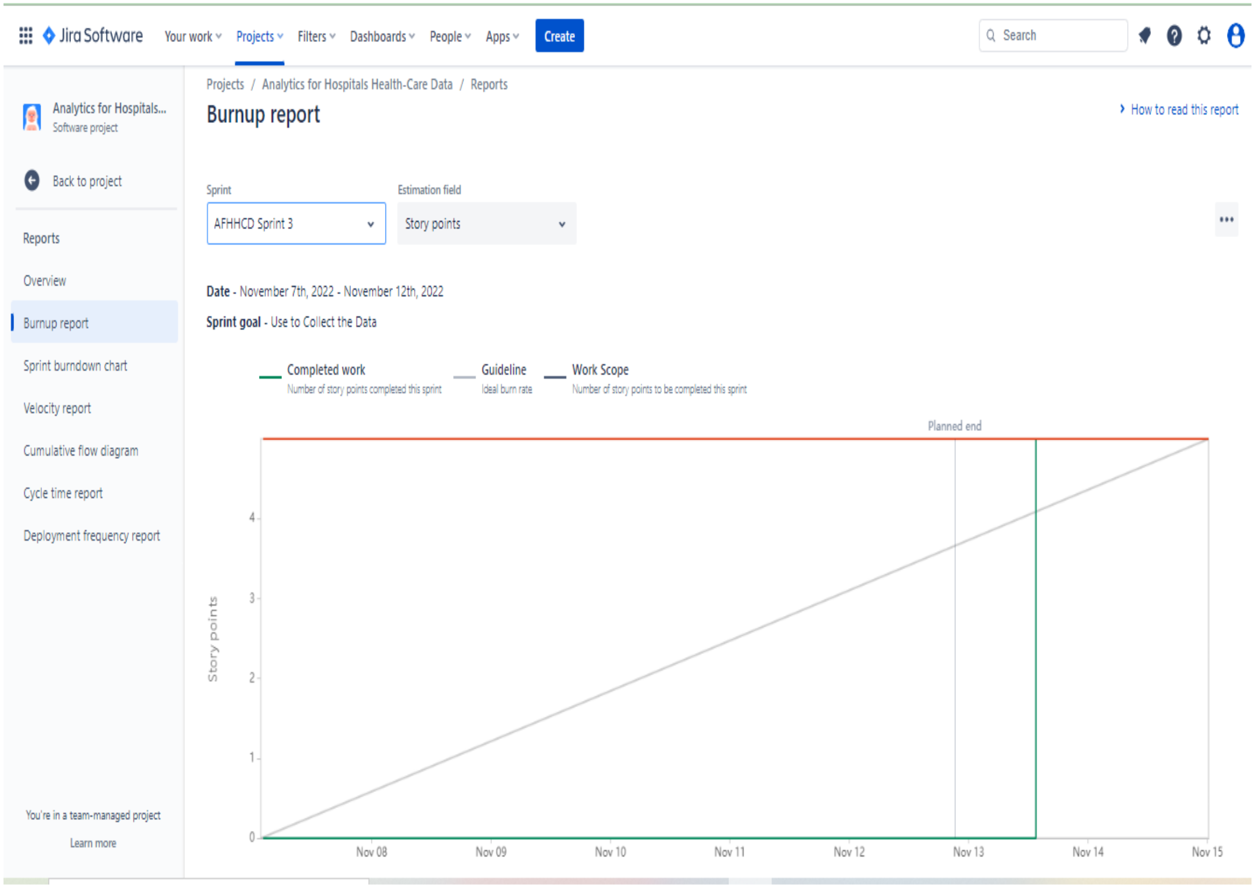
Estimation field: Story points

**Date** - October 31st, 2022 - November 5th, 2022

**Sprint goal** - Request the Forget password

- Completed work — Number of story points completed this sprint
- Guideline — Ideal burn rate
- Work Scope — Number of story points to be completed this sprint

# SPRINT-3:



Projects / Analytics for Hospitals Health-Care Data / Reports

Burnup report

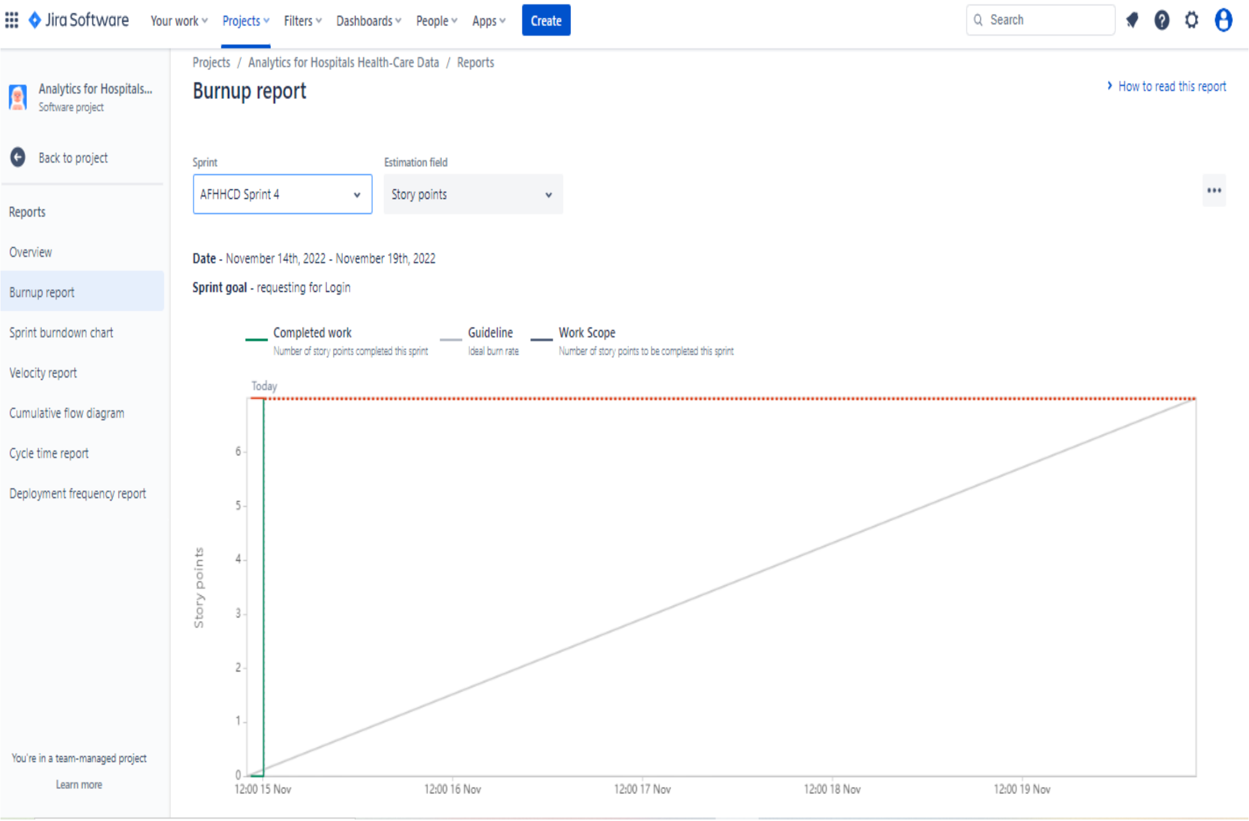
Sprint: AFHHCD Sprint 3

Estimation field: Story points

**Date** - November 7th, 2022 - November 12th, 2022

**Sprint goal** - Use to Collect the Data

# SPRINT-4:



## Burnup report

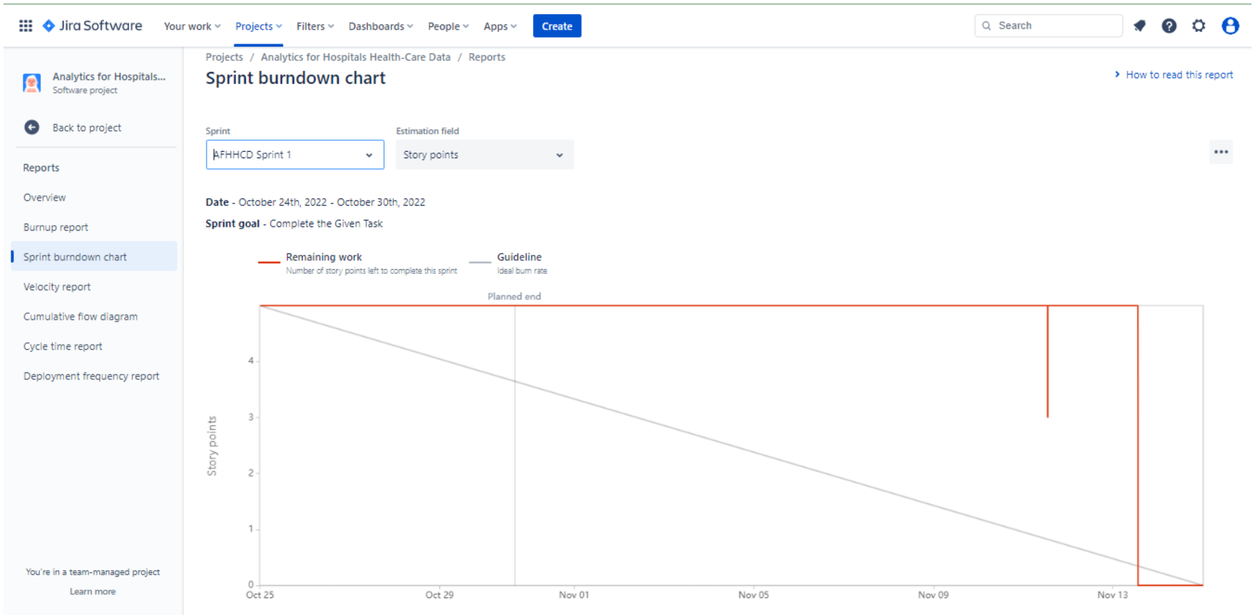
Sprint: AFHHCD Sprint 4

Estimation field: Story points

**Date** - November 14th, 2022 - November 19th, 2022

**Sprint goal** - requesting for Login

# Burndown Chart:
# Burndown Chart Sprint-1:



Sprint burndown chart

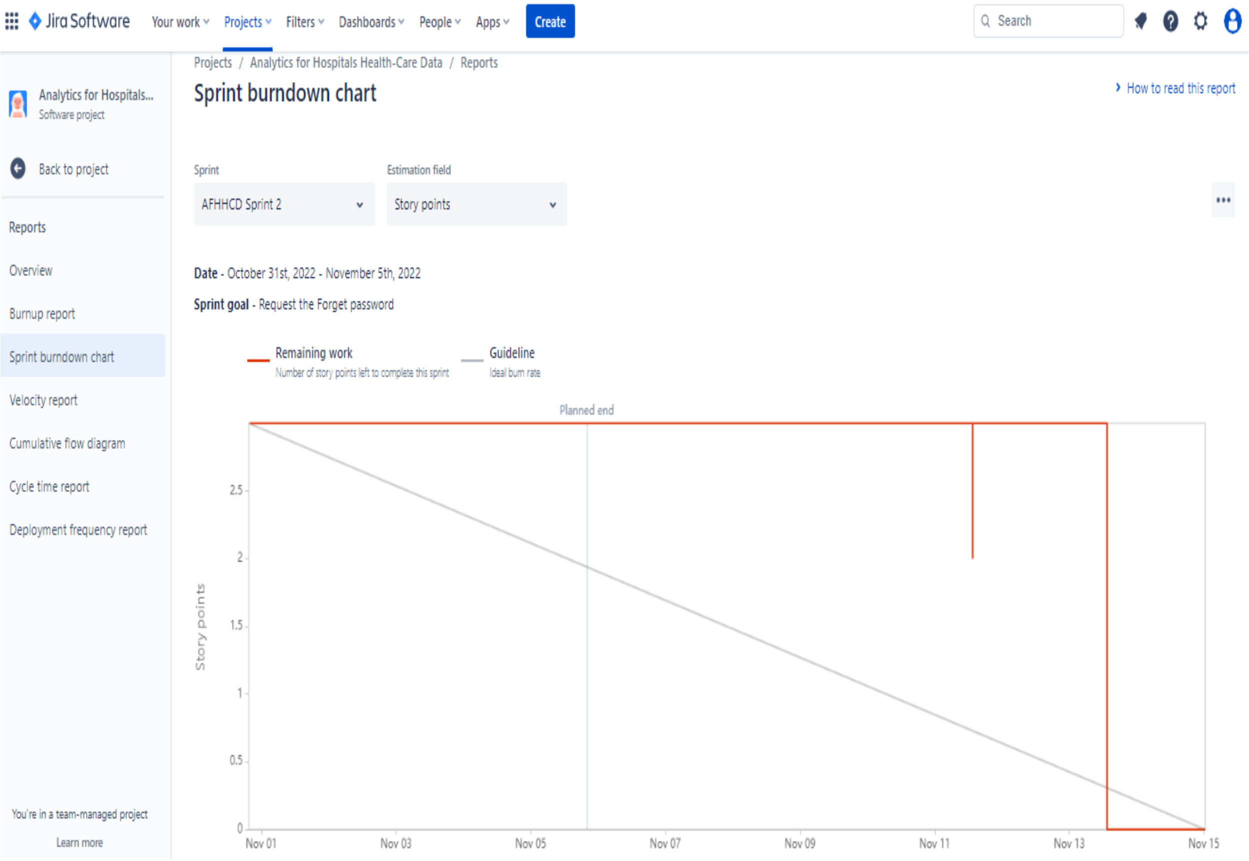
Sprint: AFHHCD Sprint 1

Estimation field: Story points

**Date** - October 24th, 2022 - October 30th, 2022

**Sprint goal** - Complete the Given Task

# Burndown Chart Sprint-2:



Projects / Analytics for Hospitals Health-Care Data / Reports

Sprint burndown chart

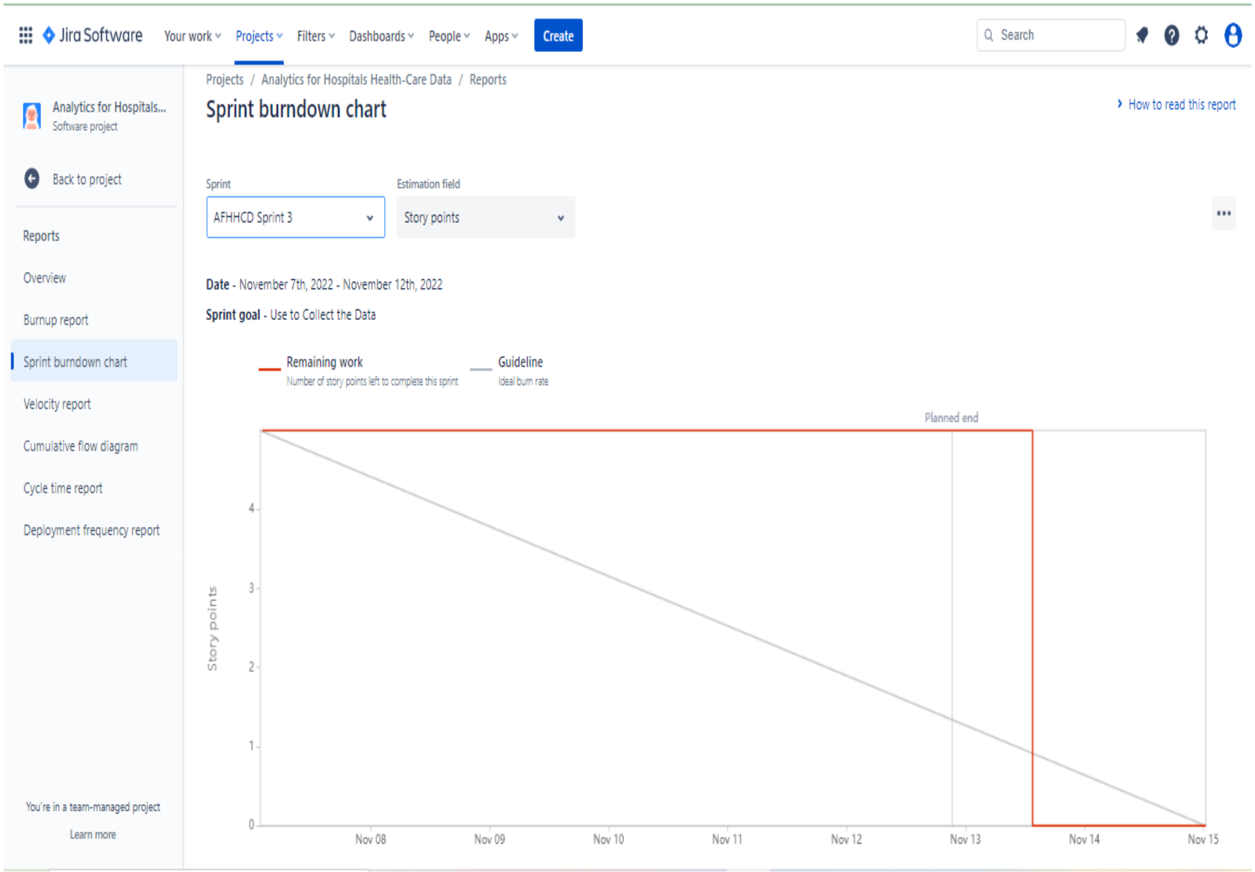
Sprint: AFHHCD Sprint 2

Estimation field: Story points

**Date** - October 31st, 2022 - November 5th, 2022

**Sprint goal** - Request the Forget password

# Burndown Chart Sprint-3:



Projects / Analytics for Hospitals Health-Care Data / Reports

Sprint burndown chart

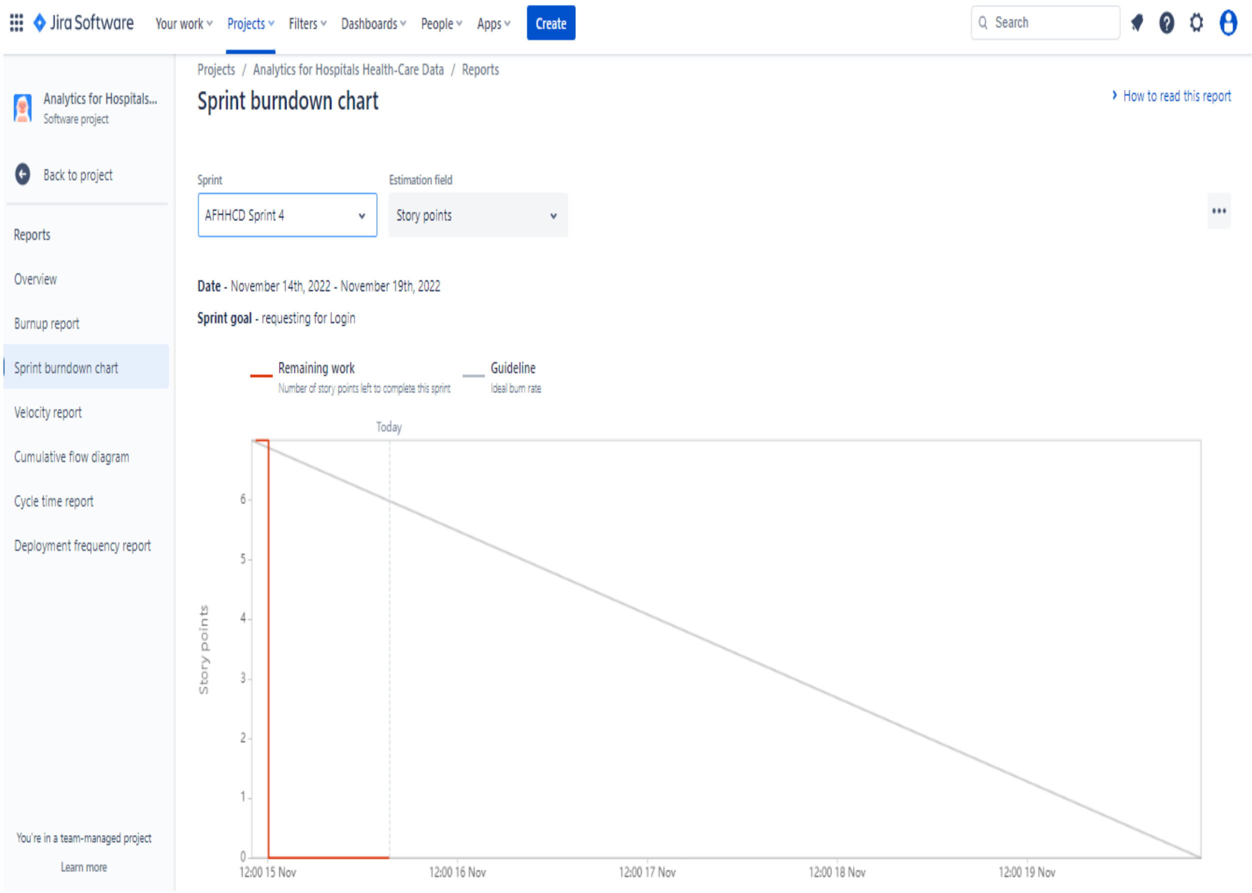
Sprint: AFHHCD Sprint 3

Estimation field: Story points

**Date** - November 7th, 2022 - November 12th, 2022

**Sprint goal** - Use to Collect the Data

# Burndown Chart Sprint-4:



Projects / Analytics for Hospitals Health-Care Data / Reports

Sprint burndown chart

Sprint: AFHHCD Sprint 4

Estimation field: Story points

**Date** - November 14th, 2022 - November 19th, 2022

**Sprint goal** - requesting for Login

# CODING & SOLUTIONING

# CHAPTER-7
# CODING & SOLUTIONING

## 7.1 Random Forest

Random Forest is a supervised learning algorithm. Random forest can be used for both classification and regression problems, by using random forest regressor we can use random forest on regression problems. But we have used random forest on classification in this internship project so we will only consider the classification part.

### 7.1.1 Random Forest pseudocode

• Randomly select “k” features from total “m” features. Where k << m
• Among the “k” features, calculate the node “d” using the best split point. • Split the node into daughter nodes using the best split.
• Repeat 1 to 3 steps until the “l” number of nodes has been reached.
• Build forest by repeating steps 1 to 4 for “n” number times to create “n” number of trees.

### 7.1.2 Random Forest prediction pseudocode

✓ Takes the test features and use the rules of each randomly created decision tree to predict the outcome and stores the predicted outcome (target).
✓ Calculate the votes for each predicted target.
✓ Consider the highly voted predicted target as the final prediction from the random forest algorithm. Code: max_accuracy = 0 for x in range(500): rf_classifier = RandomForestClassifier(random_state=x)

## Code:

```
max_accuracy = 0

for x in range(500):

rf_classifier = RandomForestClassifier(random_state=x)
```

```
rf_classifier.fit(X_train,Y_train)

Y_pred_rf = rf_classifier.predict(X_test)

current_accuracy = round(accuracy_score(Y_pred_rf,Y_test)*100,2)

if(current_accuracy>max_accuracy):

max_accuracy = current_accuracy

best_x = x

print(max_accuracy)

print(best_x)

rf_classifier = RandomForestClassifier(random_state=best_x)

rf_classifier.fit(X_train,Y_train)

Y_pred_rf = rf_classifier.predict(X_test)

Y_pred_rf.shape score_rf = round(accuracy_score(Y_pred_rf,Y_test)*100,2) score_rf
```

## 7.2. K-Nearest Neighbors

We can implement a KNN model by following the below steps:

- Load the data
- Initialize the value of k

- For getting the predicted class, iterate from 1 to total number of training data points

Calculate the distance between test data and each row of training data. Here we will use Euclidean distance as our distance metric since it's the most popular method. The other metrics that can be used are Chebyshev, cosine, etc.

○ Sort the calculated distances in ascending order based on distance values

○ Get top k rows from the sorted array

○ Get the most frequent class of these rows

○Return the predicted class

**Code**:

```
knn_classifier= KNeighborsClassifier(n_neighbors=31,leaf_size=30)

knn_classifier.fit(X_train,Y_train)

Y_pred_knn = knn_classifier.predict(X_test)

score_knn = round(accuracy_score(Y_pred_knn,Y_test)*100,2)

score_knn
```

**7.3 Decision Tree Pseudocode**:

● Place the best attribute of the dataset at the root of the tree.

● Split the training set into subsets. Subsets should be made in such a way that each subset contains data with the same value for an attribute

● Repeat step 1 and step 2 on each subset until you find leaf nodes in all the branches of the tree.

Assumptions while creating a Decision Tree- At the beginning, the whole training set is considered as the root. Feature values are preferred to be categorical. If the values are continuous then they are discretized prior to building the model. Records are distributed recursively on the basis of attribute values. Order to place attributes as root or internal node of the tree is done by using some statistical approach.

**The popular attribute selection measures:**

● Information gain

● Gini index

**Attribute selection method**- A dataset consists of “n” attributes then deciding which attribute to place at the root or at different levels of the tree as internal nodes is a complicated step. By just randomly selecting any node to be the root can’t solve the issue. If we follow a random approach, it may give us bad results with low accuracy. To solve this attribute selection problem, researchers worked and devised some solutions. They suggested using some criterion like information gain, Gini index, etc. These criterions will calculate values for every attribute. The values are sorted, and attributes are placed in the tree by following the order i.e., the attribute with a high value (in case of information gain) is placed at the root. While using information Gain as a criterion, we assume attributes to be categorical, and for Gini index, attributes are assumed to be continuous.

**Gini Index** - Gini Index is a metric to measure how often a randomly chosen element would be incorrectly identified. It means an attribute with a lower Gini index should be preferred.

**Code:**

```
dt_classifier = DecisionTreeClassifier(

max_depth=20,

min_samples_split=2,

min_samples_leaf=1,

min_weight_fraction_leaf=0.00001, max_features='auto',

random_state=46)

dt_classifier.fit(X_train, Y_train)

Y_pred_dt=dt_classifier.predict(X_test)        score_dt        =
round(accuracy_score(Y_pred_dt,Y_test)*100,2)

score_dt
```

### 7.4 Naïve Bayes

**Bayes'Theorem** is stated as:
**P(h|d) = (P(d|h) * P(h)) / P(d)**
**P(h|d)** is the probability of hypothesis h given the data d. This is called the posterior probability.
**P(d|h)** is the probability of data d given that the hypothesis h was true.
**P(h)** is the probability of hypothesis h being true (regardless of the data). This is called the prior probability of h.
**P(d)** is the probability of the data (regardless of the hypothesis).

We are interested in calculating the posterior probability of P(h|d) from the prior probability p(h) with P(D) and P(d|h). After calculating the posterior probability for a number of different hypotheses, we will select the hypothesis with the highest probability. This is the maximum probable hypothesis and may formally be called the (MAP) hypothesis.

This can be written as:

**MAP(h) = max(P(h|d)) or**
**MAP(h) = max((P(d|h) * P(h)) / P(d)) or**
**MAP(h) = max(P(d|h) * P(h))**

The P(d) is a normalizing term which allows us to calculate the probability. We can drop it when we are interested in the most probable hypothesis as it is constant and only used to normalize. Back to classification, if we have an even number of instances in each class in our training data, then the probability of each class (e.g. P(h)) will be equal. Again, this would be a constant term in our equation, and we could drop it so that we end up with:

**MAP(h) = max(P(d|h))**

Naive Bayes is a classification algorithm for binary (two-class) and multi-class classification problems. The technique is easiest to understand when described using binary or categorical input values. It is called Naive Bayes or Idiot Bayes because the calculation of the probabilities for each hypothesis are simplified to make their calculation tractable. Rather than attempting to calculate the values of each attribute value P (d1, d2, d3|h), they are assumed to be conditionally independent given the target value and calculated as P(d1|h) * P(d2|H) and so on. This is a very strong assumption that is most unlikely in real data, i.e. that the attributes do not interact. Nevertheless, the approach performs surprisingly well on data where this assumption does not hold.

**MAP(h) = max(P(d|h) * P(h))**

**Gaussian Naïve Bayes**:

**mean(x) = 1/n * sum(x)**

Where n is the number of instances and x are the values for an input variable in your training data. We can calculate the standard deviation using the following equation:

**standard deviation(x) = sqrt (1/n * sum(xi-mean(x)^2))**

This is the square root of the average squared difference of each value of x from the mean value of x, where n is the number of instances, sqrt() is the square root function, sum() is the sum function, xi is a specific value of the x variable for the i'th instance and mean(x) is described above, and ^2 is the square. Gaussian PDF with a new input for the variable, and in return the Gaussian PDF will provide an estimate of the probability of that new input value for that class.

**pdf (x, mean, sd) = (1 / (sqrt (2 * PI) * sd)) * exp (-((x-mean^2)/(2*sd^2)))**

Where pdf(x) is the Gaussian Probability Density Function (PDF), sqrt () is the square root, mean and sd are the mean and standard deviation calculated above, Pi is the numerical constant, exp () is the numerical constant e or Euler's number raised to power and x is the input value for the input variable.

**Code:**

```
 nb_classifier = GaussianNB( var_smoothing=1e-50)

nb_classifier.fit(X_train,Y_train)
nb_classifier.predict(X_test)

Y_pred_nb = nb_classifier.predict(X_test)

score_nb = round(accuracy_score(Y_pred_nb,Y_test)*100,2)

score_nb
```

## 7.6 Libraries used

Python has a vast reserve of inbuilt standard libraries which includes areas like web services tools, string operation, data analysis, and machine learning, etc. The complex programming tasks can be dealt with ease using these inbuilt libraries as it reduces the size of code with many inbuilt functions that do the job pretty well for its user.

### 7.6.1 Data Visualization

- Matplotlib: Matplotlib is a cross-platform, data visualization and graphical plotting library for Python and its numerical mathematics extension NumPy, a big data numerical handling resource.
  - pyplot
  - rcParams
  - rainbow
- Seaborn: Seaborn is an open-source Python library built on top of matplotlib. It is used for data visualization and exploratory data analysis. Seaborn works easily with dataframes and the Pandas library. The graphs created can also be customized easily.

### 7.6.2 Data Manipulation

- NumPy: The NumPy library in python is used for scientific computing and array manipulation. It can perform different operations such as indexing of an array, sequencing, and slicing, etc.

● Pandas: The Pandas library in python is used for structuring, manipulating, and organizing data in a tabular structure called the data frame which is further used for data analysis.

● Scikit-learn: ○ sklearn.model_selection ■ train_test_split ○ sklearn.preprocessing ■ StandardScaler ■ LabelEncoder

### 7.6.3 Data Modeling

● Scikit-learn: Scikit-learn is one of the most useful libraries that python offers. It has various statistical learning algorithms such as regression models (linear regression, logistic regression), SVM's, random forest for classification tasks and k-means for clustering, etc.

○ sklearn.ensemble.RandomForestClassifier

○ sklearn.neighbors.KNeighborsClassifier

○ sklearn.tree.DecisionTreeClassifier

○ sklearn.naive_bayes.GaussianNB

### 7.6.4 Data Validation

● **Scikit-learn-metrics:** The sklearn.metrics module implements several loss, score, and utility functions to measure classification performance. sklearn.metrics - log_loss, roc_auc_score, precision_score, f1_score, recall_score, roc_curve, auc, plot_roc_curve, classification_report, confusion_matrix, accuracy_score, fbeta_score, matthews_corrcoef

● **Mlxtend**: Mlxtend (machine learning extensions) is a Python library of useful tools for day-to-day data science tasks. mlxtend.plotting - plot_confusion_matrix

# TESTING

# CHAPTER-8

# TESTING

## 8. Testing

### 8.1 Testing and Validations

Validation is a complex process with many possible variations and options, so specifics vary from database to database, but the general outline is:

• **Requirement Gathering**

o The Sponsor decides what the database is required to do based on regulations, company needs, and any other important factors.

o The requirements are documented and approved.

• **System Testing**

o Procedures to test the requirements are created and documented.

o The version of the database that will be used for validation is set up.

o The Sponsor approves the test procedures.

o The tests are performed and documented.

o Any needed changes are made. This may require another, shorter round of testing and documentation.

• **System Release**

o The validation documentation is finalized.

o The database is put into production. .

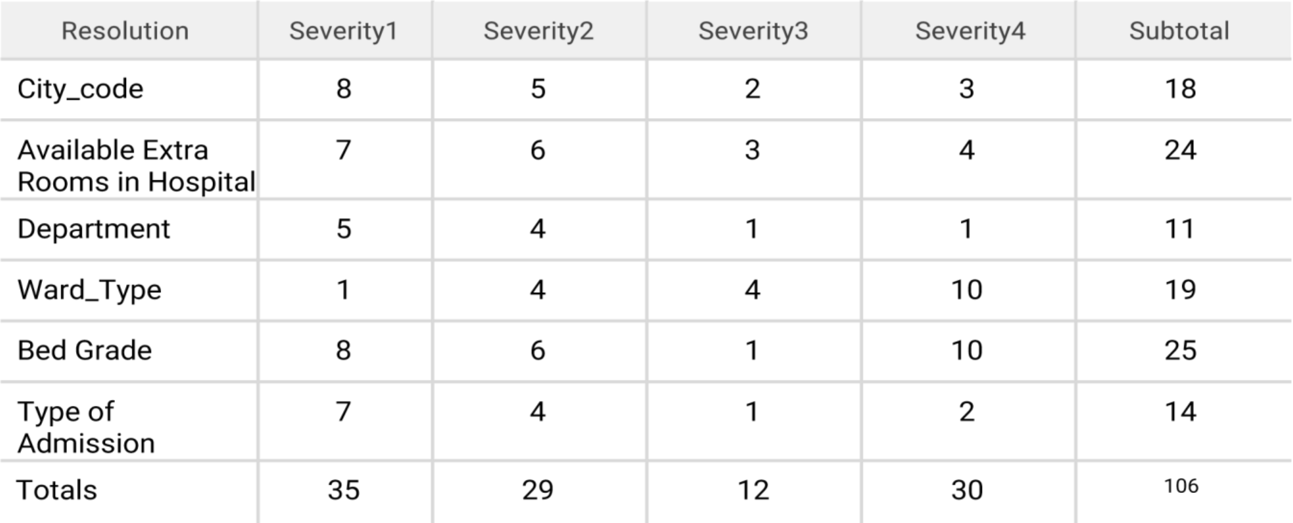
## 8.1.1 User Acceptance Testing

### 1.Purpose of Document

The purpose oft his documentation to briefly explain the test coverageandopenissuesof the[VisualizingandPredictingHeartDiseases]projectat the timeofthe releaseto UserAcceptanceTesting(UAT).

### 2.DefectAnalysis

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



| Resolution | Severity1 | Severity2 | Severity3 | Severity4 | Subtotal |
|---|---|---|---|---|---|
| City_code | 8 | 5 | 2 | 3 | 18 |
| Available Extra Rooms in Hospital | 7 | 6 | 3 | 4 | 24 |
| Department | 5 | 4 | 1 | 1 | 11 |
| Ward_Type | 1 | 4 | 4 | 10 | 19 |
| Bed Grade | 8 | 6 | 1 | 10 | 25 |
| Type of Admission | 7 | 4 | 1 | 2 | 14 |
| Totals | 35 | 29 | 12 | 30 | 106 |

### 3.TestCaseAnalysis

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

| Section | Total Cases | Not Tested | Fail | Pass |
|---|---|---|---|---|
| City_code | 22 | 0 | 0 | 22 |
| Available ExtraRooms in Hospital | 31 | 0 | 0 | 31 |
| Department | 4 | 0 | 0 | 4 |
| BedGrade | 9 | 0 | 0 | 9 |
| TypeofAdmission | 2 | 0 | 0 | 2 |

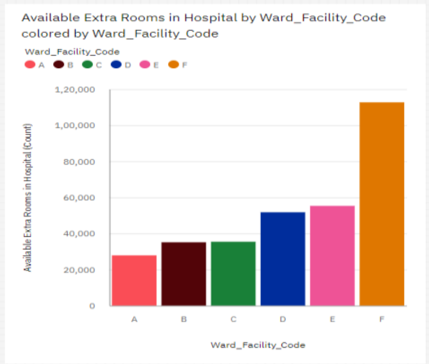
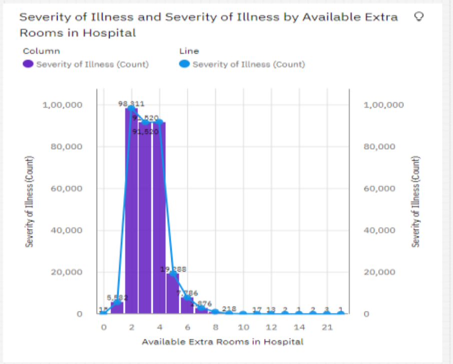
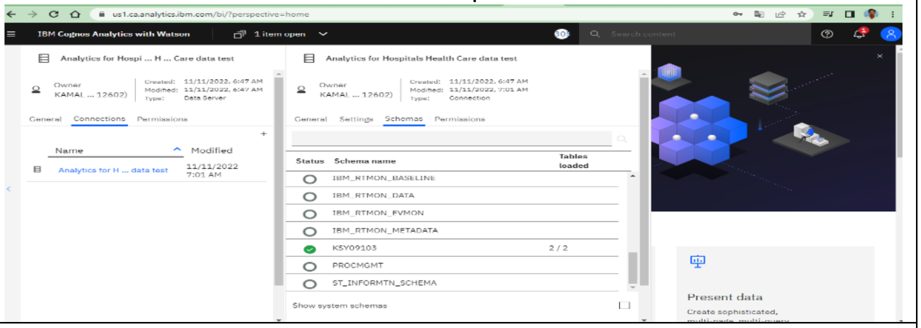
## 8.1.2 Test Cases Report

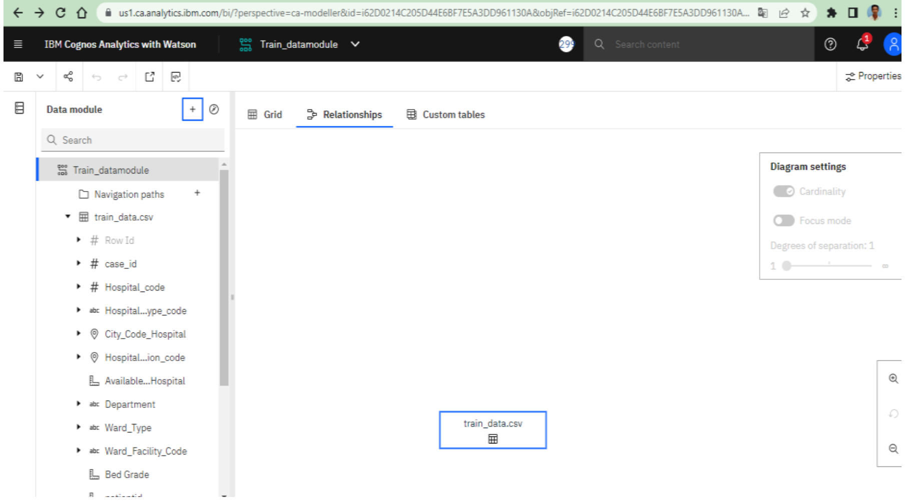
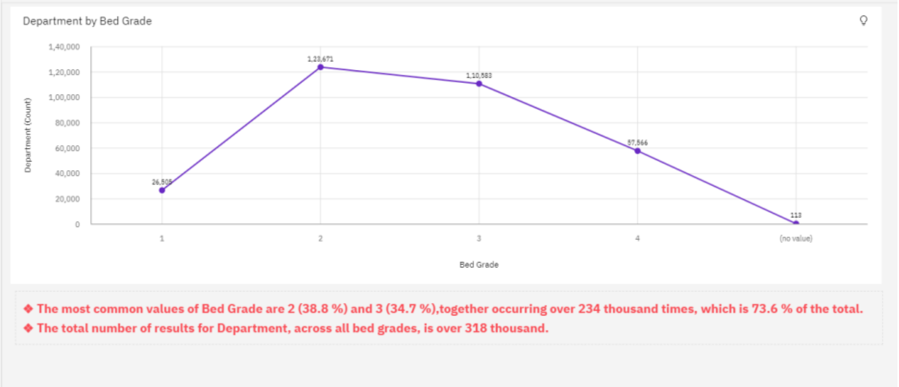
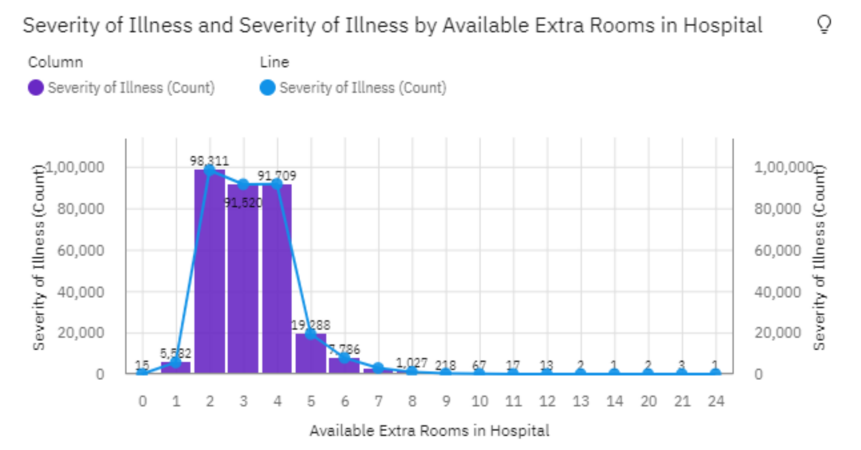
| Test case ID | Feature Type | Component | Test Scenario | Pre-Requisite | Steps To Execute | Test Data | Expected Result | Actual Result | Status | Commnets | TC for Automation(Y/N) | BUG ID | Executed By |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| City_Code | Dashboard/ report, report | Cognos Analytics | Verify the dataset for accurate performance | A quality dataset | 1.Upload the dataset<br>2.Explore the data<br>3.Create dashboard/ Report ,Story | https://github.com/IBM-EPBL/IBM-Project-456221660731319/blob/main/Final%20Deliverable/Datasets/test_data.csv | Accurate Prediction | Working as expected | Pass | Cognos analytics to accurate predict of patients City_Code | yes | high | VIJAY VIGNESH .S |
| Avaliable Extra Room in Hospital | Dashboard/ report, report | Cognos Analytics | Verify the dataset for accurate performance | A quality dataset | 1.Upload the dataset<br>2.Explore the data<br>3.Create dashboard/ Report ,Story | https://github.com/IBM-EPBL/IBM-Project-456221660731319/blob/main/Final%20Deliverable/Datasets/test_data.csv | Accurate Prediction | Working as expected | fail | Cognos analytics to accurate predict of patients Avslisble Extra Room in Hospital | no | low | AVINASH .T |
| Department | Dashboard/ report, report | Cognos Analytics | Verify the dataset for accurate performance | A quality dataset | 1.Upload the dataset<br>2.Explore the data<br>3.Create dashboard/ Report ,Story | https://github.com/IBM-EPBL/IBM-Project-456221660731319/blob/main/Final%20Deliverable/Datasets/test_data.csv | Accurate Prediction | Working as expected | Pass | some data not accuracy | yes | high | KRATHIK VARUN .B |
| Ward_Type | Dashboard/ report, report | Cognos Analytics | Verify the dataset for accurate performance | A quality dataset | 1.Upload the dataset<br>2.Explore the data<br>3.Create dashboard/ Report ,Stor | https://github.com/IBM-EPBL/IBM-Project-456221660731319/blob/main/Final%20Deliverable/Datasets/test_data.csv | Accurate Prediction | Working as expected | Pass | Cognos analytics to accurate predict of patients Ward_Type | yes | high | KAMALESH .R |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | *y* | | | | | | | | |
| *Bed Grade* | *Dashboard/ report, report* | *Cognos Analytics* | *Verify the dataset for accurate performance* | *A quality dataset* | *1.Upload the dataset*<br>*2.Explore the data*<br>*3.Create dashboard/ Report ,Story* | [*https://github.com/IBM-EPBL/IBM-Project-456221660731319/blob/main/Final%20Deliverable/Datasets/test_data.csv*](https://github.com/IBM-EPBL/IBM-Project-456221660731319/blob/main/Final%20Deliverable/Datasets/test_data.csv) | *Accurate Prediction* | *Working as expected* | *fail* | *Cognos analytics to accurate predict of patients Bed Grade* | *no* | *low* | *VIJAY VIGNESH .S* |
| *Type of Admission* | *Dashboard/ report, report* | *Cognos Analytics* | *Verify the dataset for accurate performance* | *A quality dataset* | *1.Upload the dataset*<br>*2.Explore the data*<br>*3.Create dashboard/ Report ,Story* | [*https://github.com/IBM-EPBL/IBM-Project-456221660731319/blob/main/Final%20Deliverable/Datasets/test_data.csv*](https://github.com/IBM-EPBL/IBM-Project-456221660731319/blob/main/Final%20Deliverable/Datasets/test_data.csv) | *Accurate Prediction* | *Working as expected* | *Pass* | *Cognos analytics to accurate predict of Type of Admission* | *yes* | *high* | *KAMALESH .R* |

### 8.1.3 Performance Testing

Project team shall fill the following information in model performance testing template.

| S.No | Parameter | Screenshot / Values |
|---|---|---|
| 1. | Dashboard designs | No of Visualizations / Graphs –11 dashboard tabs with 1-2 visualizations in each dashboard<br><br> <br><br> |
| 2. | Data Responsiveness | It hides certain aspects of the visualization if the size is limited, to maximize the space that is available to display data.<br>• There was two different datasets with the common column and full outer join was done by that common column.<br>• There was another dataset with various continuous values , those values was grouped as common. |
| 3. | Amount Data to Rendered (DB2 Metrics) | There are some relevant datasets are uploaded in the IBM DB2.<br><br>us1.ca.analytics.ibm.com/bi/?perspective=home<br>IBM Cognos Analytics with Watson<br>1 item open<br>Search content<br>Analytics for Hospi ... H ... Care data test<br>Owner KAMAL ... 12602)<br>Created: 11/11/2022, 6:47 AM<br>Modified: 11/11/2022, 6:47 AM<br>Type: Data Server<br>General Connections Permissions<br>Name Modified<br>Analytics for H ... data test 11/11/2022 7:01 AM<br>Analytics for Hospitals Health Care data test<br>Owner KAMAL ... 12602)<br>Created: 11/11/2022, 6:47 AM<br>Modified: 11/11/2022, 7:01 AM<br>Type: Connection<br>General Settings Schemas Permissions<br>Status Schema name Tables loaded<br>○ IBM_RTMON_BASELINE<br>○ IBM_RTMON_DATA<br>○ IBM_RTMON_FVMON<br>○ IBM_RTMON_METADATA<br>✓ KSY09103 2 / 2<br>○ PROCMGMT<br>○ ST_INFORMTN_SCHEMA<br>Show system schemas □<br>Present data<br>Create sophisticated, |
| 4. | Utilization of Data Filters | If any similar datasets in the provided area this is reduced by joining the common column of those two dataset. i.e. Cardinality is used. |

| | | |
|---|---|---|
| | |  us1.ca.analytics.ibm.com/bi/?perspective=ca-modeller&id=i62D0214C205D44E6BF7E5A3DD961130A&objRef=i62D0214C205D44E6BF7E5A3DD961130A...<br>IBM Cognos Analytics with Watson<br>Train_datamodule<br>Search content<br>Properties<br>Data module<br>Search<br>Train_datamodule<br>Navigation paths<br>train_data.csv<br>Row Id<br>case_id<br>Hospital_code<br>Hospital...ype_code<br>City_Code_Hospital<br>Hospital...ion_code<br>Available...Hospital<br>Department<br>Ward_Type<br>Ward_Facility_Code<br>Bed Grade<br>Grid<br>Relationships<br>Custom tables<br>Diagram settings<br>Cardinality<br>Focus mode<br>Degrees of separation: 1<br>train_data.csv |
| 5. | Effective User Story | No of Scene Added – 10 stories with 1-2 visualizations in each story<br> <br>❖ The most common values of Bed Grade are 2 (38.8 %) and 3 (34.7 %),together occurring over 234 thousand times, which is 73.6 % of the total.<br>❖ The total number of results for Department, across all bed grades, is over 318 thousand. |
| 6. | Descriptive Reports | No of Visualizations / Graphs – 2 reports with 3 – 5 visualization in each report<br> |

## 8.2 Testing Levels

### 8.2.1 Functional Testing:

This type of testing is done against the functional requirements of the project
. Types:

Unit testing: Each unit /module of the project is individually tested to check for bugs. If any bugs found by the testing team, it is reported to the developer for fixing.

Integration testing: All the units are now integrated as one single unit and checked for bugs. This also checks if all the modules are working properly with each other.

System testing: This testing checks for operating system compatibility. It includes both functional and non functional requirements.

Sanity testing: It ensures change in the code doesn’t affect the working of the project.

Smoke testing: this type of testing is a set of small tests designed for each build. Interface testing: Testing of the interface and its proper functioning.

Regression testing: Testing the software repetitively when a new requirement is added, when bug fixed etc.

Beta/Acceptance testing: User level testing to obtain user feedback on the product.

### 8.2.2 Non-Functional Testing:

This type of testing is mainly concerned with the non-functional requirements such as performance of the system under various scenarios. Performance testing: Checks for speed, stability and reliability of the software, hardware or even the network of the system under test.

Compatibility testing: This type of testing checks for compatibility of the system with different operating systems, different networks etc. Localization testing: This checks for the localized version of the product mainly concerned with UI.

Security testing: Checks if the software has vulnerabilities and if any, fix them.

Reliability testing: Checks for the reliability of the software

Stress testing: This testing checks the performance of the system when it is exposed to different stress levels.

Usability testing: Type of testing checks the easily the software is being used by the customers.

Compliance testing: Type of testing to determine the compliance of a system with internal or external standards

**Reliability**

The structure must be reliable and strong in giving the functionalities. The movements must be made unmistakable by the structure when a customer has revealed a couple of enhancements. The progressions made by the Programmer must be Project pioneer and in addition the Test designer.

• **Maintainability**

The system watching and upkeep should be fundamental and focus in its approach. There should not be an excess of occupations running on diverse machines such that it gets hard to screen whether the employments are running without lapses.

• **Performance**

The framework will be utilized by numerous representatives all the while. Since the system will be encouraged on a single web server with a lone database server outside of anyone's ability to see, execution transforms into a significant concern. The structure should not capitulate when various customers would use everything the while. It should allow brisk accessibility to each and every piece of its customers. For instance, if two test specialists are all the while attempting to report the vicinity of a bug, then there ought not to be any irregularity at the same time.

• **Portability** T
he framework should to be effectively versatile to another framework. This is obliged when the web server, which s facilitating the framework gets adhered because of a few issues, which requires the framework to be taken to another framework.

• **Scalability**
The framework should be sufficiently adaptable to include new functionalities at a later stage. There should be a run of the mill channel, which can oblige the new functionalities.

• **Flexibility**
Flexibility is the capacity of a framework to adjust to changing situations and circumstances, and to adapt to changes to business approaches and rules. An adaptable framework is one that is anything but difficult to reconfigure.

### 8.3 White Box Testing

White Box Testing is defined as the testing of a software solution's internal structure, design, and coding. In this type of testing, the code is visible to the tester. It focuses primarily on verifying the flow of inputs and outputs through the application, improving design and usability, strengthening security. White box testing is also known as Clear Box testing, Open Box testing, Structural testing, Transparent Box testing, Code-Based testing, and Glass Box testing. It is usually performed by developers.

It is one of two parts of the "Box Testing" approach to software testing. Its counterpart, Blackbox testing, involves testing from an external or enduser type perspective. On the other hand, Whitebox testing is based on the inner workings of an application and revolves around internal testing.

The term "WhiteBox" was used because of the see-through box concept. The clear box or WhiteBox name symbolizes the ability to see through the software's outer shell (or "box") into its inner workings. Likewise, the "black box" in "Black Box Testing" symbolizes not being able to see the inner workings of the software so that only the end-user experience can be tested.

## 8.4 Different Stages of Testing

### 8.4.1 Unit Testing

UNIT TESTING is a level of software testing where individual units/ components of software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output. In procedural programming, a unit may be an individual program, function, procedure, etc. In object-oriented programming, the smallest unit is a method, which may belong to a base/ super class, abstract class or derived/ child class. (Some treat a module of an application as a unit. This is to be discouraged as there will probably be many individual units within that module.) Unit testing frameworks, drivers, stubs, and mock/ fake objects are used to assist in unit testing.

#### Benefits

Unit testing increases confidence in changing/ maintaining code. If good unit tests are written and if they are run every time any code is changed, we will be able to promptly catch any defects introduced due to the change. Also, if codes are already made less interdependent to make unit testing possible,the unintended impact of changes to any code is less. Codes are more reusable. In order to make unit testing possible, codes need to be modular. This means that codes are easier to reuse. Development is faster. How? If you do not have unit testing in place, you write your code and perform that fuzzy ‘developer test’ (You set some breakpoints, fire up the GUI, provide a few inputs that hopefully hit your code and hope that you are all set.) But, if you have unit testing in place, you write the test, write the code and run the test. Writing tests takes time but the time is compensated by the less amount of time it takes to run the tests; You need not fire up the GUI and provide all those inputs. And, of course, unit tests are more reliable than ‘developer tests’. Development is faster in the long run too. How? The effort required to find and fix defects found during unit testing is very less in comparison to the effort required to fix defects found during system testing or acceptance testing.

### 8.4.2 Integration Testing

INTEGRATION TESTING is a level of software testing where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units. Test drivers and test stubs are used to assist in Integration Testing

Tasks
Integration Test Plan

o Prepare

o Review

o Rework

o Baseline Integration Test Cases/Scripts

o Prepare

o Review

o Rework

o Baseline Integration Test

### 8.4.3 System Testing

SYSTEM TESTING is a level of software testing where a complete and integrated software is tested. The purpose of this test is to evaluate the system’s compliance with the specified requirements.

system testing: The process of testing an integrated system to verify that it meets specified requirements

### 8.4.4 Acceptance Testing

ACCEPTANCE TESTING is a level of software testing where a system is tested for acceptability. The purpose of this test is to evaluate the system's compliance with the business requirements and assess whether it is acceptable for delivery.

**TESTING HEALTH CARE DATA:**

Testing is the process used to help identify the correctness, completeness, security and quality of the developed computer software. Testing is the process of technical investigation and includes the process of executing a program or application with the intent of finding errors. In the training process, our model learns to associate a particular input (i.e. features) to the corresponding output (tag) based on the test samples used for training. Input features and tags (e.g. 1-normal 2-health disease) are fed into the machine learning algorithm to generate a model. A comparative analysis of different classifiers was performed for the classification of the Health dataset in order to correctly classify and predict Health disease cases with minimal attributes.

| Input | Expected Output | Actual Output |
|---|---|---|
| Data Visualization | Various visual representations of the data to understand more about the relationship between various features. | Pass |
| Data Processing | Convert some categorical variables into dummy variables and scale all the values before training the Machine Learning models. | Pass |
| Dataset | Split the dataset into training and testing datasets. | Pass |
| Training dataset | Train the model using the training dataset. | Pass |
| Testing dataset | Tests if the model is accurate based on the output of the testing dataset. | Pass |

Training and Subsequent testing

| Input | Expected Output | Actual Output |
|---|---|---|
| No Health cate data failure | Should be labeled as 1 (no health care data failure) and should show output as “The patient is not likely to have health disease”. | Pass |
| Health care data analysis failure | Should be labeled as 2 (health care data failure ) and should show output as “The patient is likely to have health care data failure”. | Pass |

## 8.5 Model Evaluation

The most important evaluation metrics for this problem domain are Accuracy, Sensitivity, Specificity, Precision, F1-measure, Log Loss, ROC and Mathew correlation coefficient.

• Accuracy: which refers to how close a measurement is to the true value and can be calculated using the following formula:

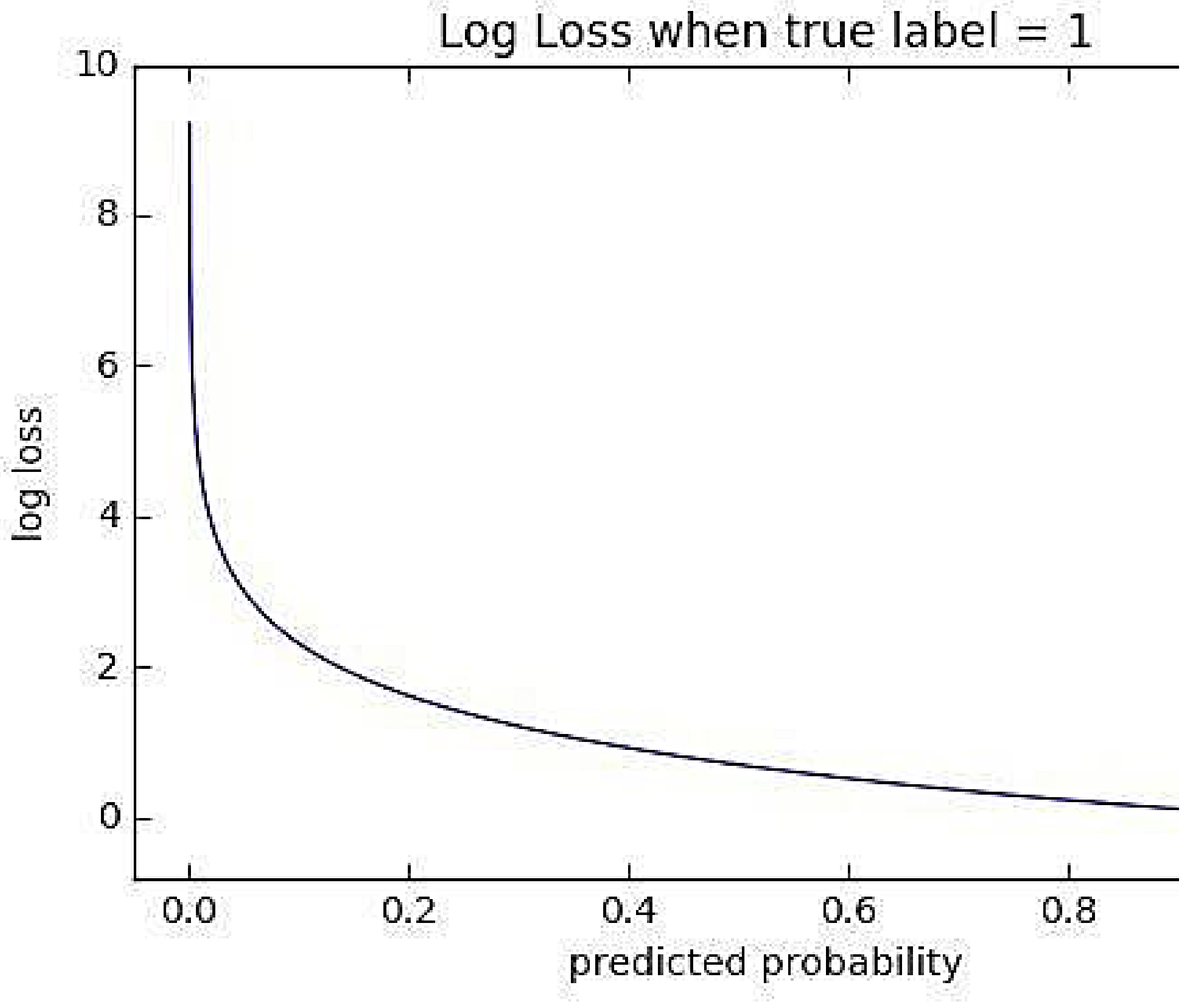
• Precision: which is how consistent results are when measurements are repeated and can be calculated using the following formula: Precision = True Positive / (True Positive + False Positive)

• Sensitivity: Sensitivity is a measure of the proportion of actual positive cases that gotpredicted as positive (or true positive). Sensitivity is also termed as Recall. Sensitivity = True Positive / (True Positive + False Negative)

• Specificity: Specificity is defined as the proportion of actual negatives, which got predicted asthe negative (or true negative). Specificity = True Negative / (True Negative + False Positive) • Mathew Correlation coefficient

(MCC): The Matthews correlation coefficient (MCC), instead, is a more reliable statisticalrate which produces a high score only if the prediction obtained good results in all ofthe four confusion matrix categories (true positives, false negatives, true negatives, and false positives), proportionally both to the size of positive elements and the size of negative elements in the dataset.

• Logic loss Logarithmic loss measures the performance of a classification model where the prediction input is a probability value between 0 and 1. The goal of our machine learning models is to minimize this value. A perfect model would have a log loss of 0. Log loss increases as the predicted probability diverges from the actual label. So, predicting a probability of .012 when the actual observation label is 1 would be bad and result in a high log loss.



**Log Loss Graph**

• F1 Score F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. Intuitively it is not as easy to understand as accuracy, but F1 score is usually more useful than accuracy, especially if you have an uneven class distribution. Accuracy works best if false positives and false negatives have similar cost.

**F1 Score = 2(Recall Precision) / (Recall + Precision)**

**• ROC Curve**

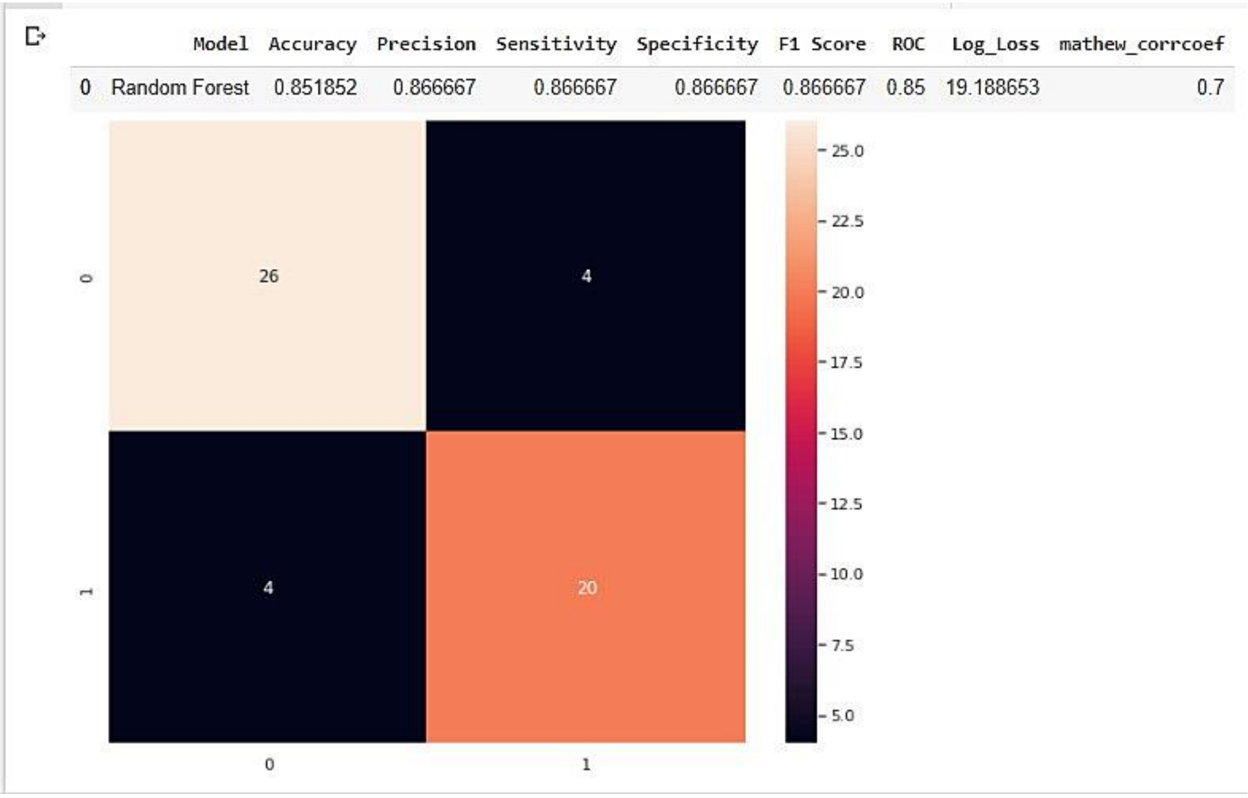
An ROC curve (receiver operating characteristic curve) is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters: True Positive Rate & False Positive Rate.

### 8.5.1 Random Forest Classifier

**Code:**

```
y_pred_rfe = rf_classifier.predict(X_test)

plt.figure(figsize=(10, 8))

 CM=confusion_matrix(Y_test,y_pred_rfe)

sns.heatmap(CM, annot=True)

TN = CM[0][0] FN = CM[1][0] TP = CM[1][1]

FP = CM[0][1] specificity = TN/(TN+FP)
```

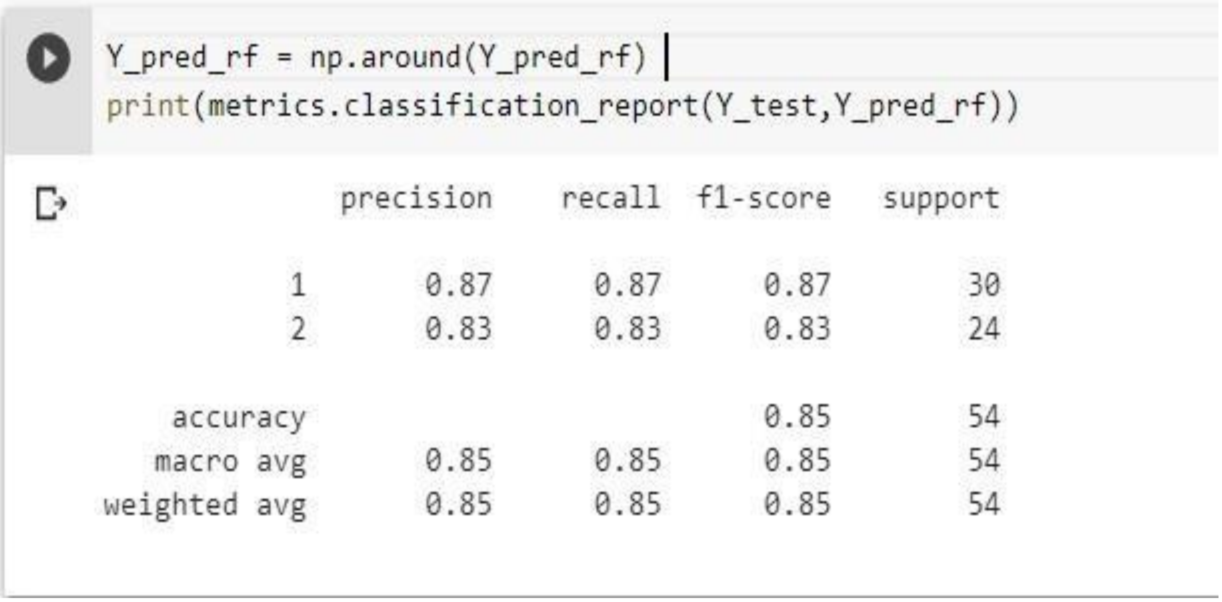
```
loss_log = log_loss(Y_test, y_pred_rfe)

acc= accuracy_score(Y_test, y_pred_rfe)

 roc=roc_auc_score(Y_test, y_pred_rfe)

prec = precision_score(Y_test, y_pred_rfe)

rec = recall_score(Y_test, y_pred_rfe)

f1 = f1_score(Y_test, y_pred_rfe)

 mathew = matthews_corrcoef(Y_test, y_pred_rfe)

model_results =pd.DataFrame([['Random Forest',acc, prec,rec,specificity, f1,roc, loss_log,mathew]], columns = ['Model', 'Accuracy','Precision', 'Sensitivity','Specificity', 'F1 Score','ROC','Log_Loss','mathew_corrcoef'])

model_results
```



| | Model | Accuracy | Precision | Sensitivity | Specificity | F1 Score | ROC | Log_Loss | mathew_corrcoef |
|---|---|---|---|---|---|---|---|---|---|
| 0 | Random Forest | 0.851852 | 0.866667 | 0.866667 | 0.866667 | 0.866667 | 0.85 | 19.188653 | 0.7 |

**Random Forest Confusion Matrix**

```
Y_pred_rf = np.around(Y_pred_rf)

print(metrics.classification_report(Y_test,Y_pred_rf))
```



```
Y_pred_rf = np.around(Y_pred_rf)
print(metrics.classification_report(Y_test,Y_pred_rf))
```

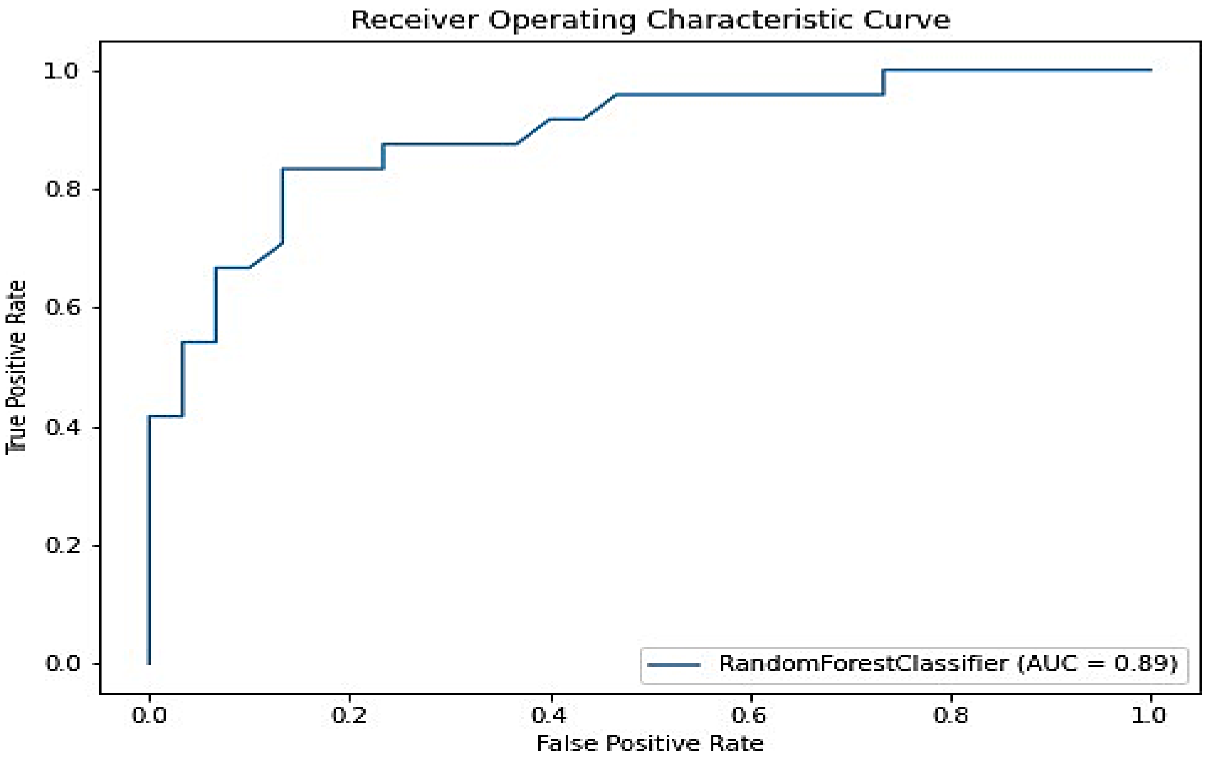
```
              precision    recall  f1-score   support

           1       0.87      0.87      0.87        30
           2       0.83      0.83      0.83        24

    accuracy                           0.85        54
   macro avg       0.85      0.85      0.85        54
weighted avg       0.85      0.85      0.85        54
```

**Random Forest Classification Report**

plot_roc_curve(rf_classifier,X_test,Y_test)

plt.xlabel('False Positive Rate') plt.ylabel('True Positive Rate')

plt.title('Receiver Operating Characteristic Curve')

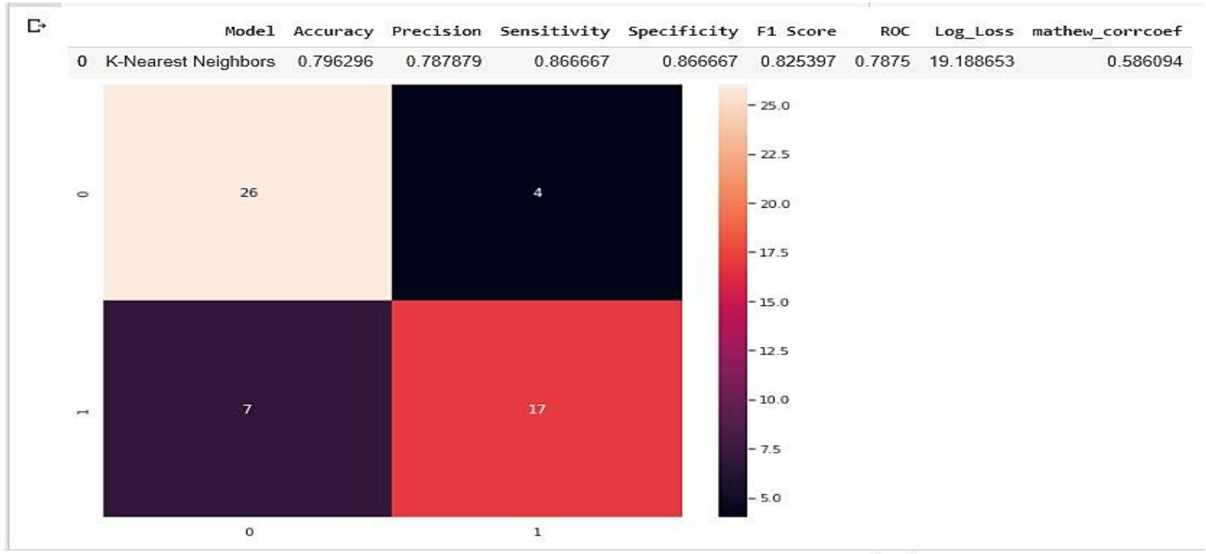


**Random Forest ROC Curve**

### 8.5.2 K-Nearest Neighbors Classifier

```
y_pred_knne = knn_classifier.predict(X_test)

plt.figure(figsize=(10, 8))

CM=confusion_matrix(Y_test,y_pred_knne) sns.heatmap(CM, annot=True)

TN = CM[0][0]

FN = CM[1][0]

TP = CM[1][1]

FP = CM[0][1]

specificity = TN/(TN+FP)

loss_log = log_loss(Y_test, y_pred_knne)

acc= accuracy_score(Y_test, y_pred_knne)
```

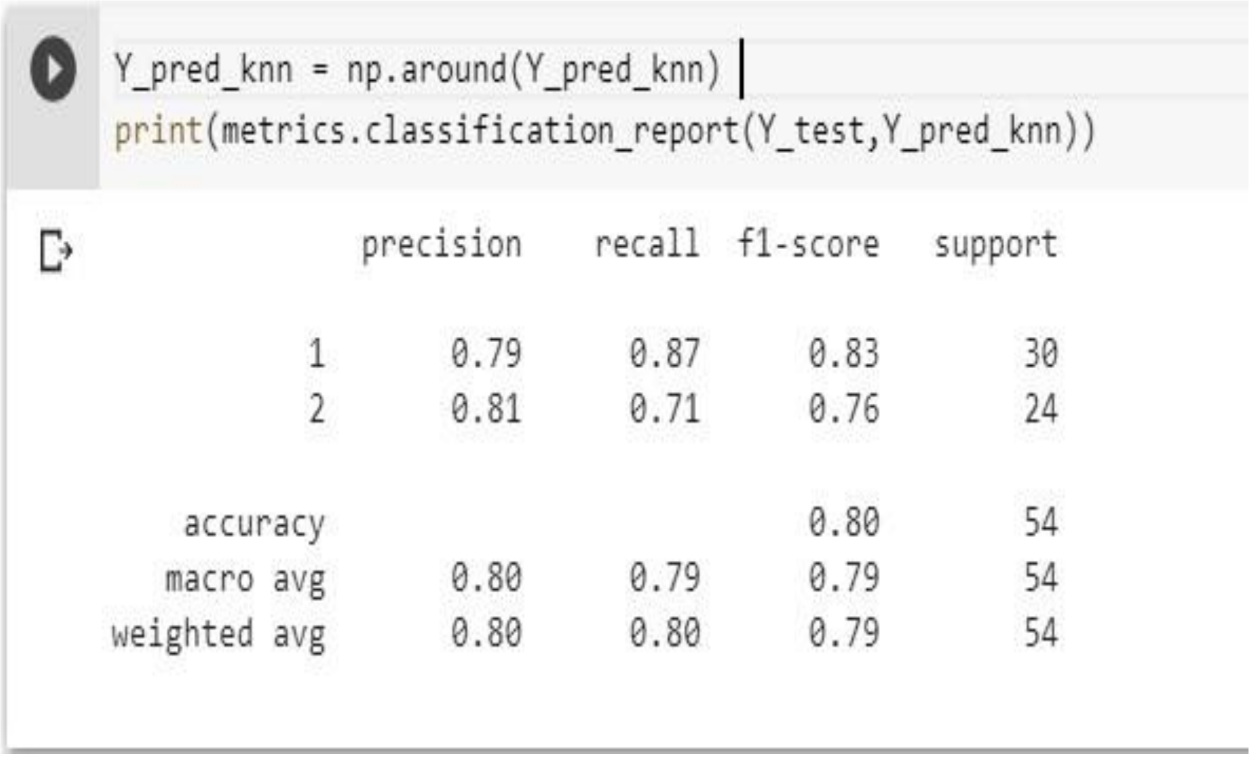
```
roc=roc_auc_score(Y_test, y_pred_knne)

prec = precision_score(Y_test, y_pred_knne)

 rec = recall_score(Y_test, y_pred_knne)

f1 = f1_score(Y_test, y_pred_knne)

mathew = matthews_corrcoef(Y_test, y_pred_knne)

model_results =pd.DataFrame([['K-Nearest Neighbors ',acc, prec,rec,specificity, f1,roc,
loss_log,mathew]], columns = ['Model', 'Accuracy','Precision', 'Sensitivity','Specificity', 'F1
Score','ROC','Log_Loss','mathew_corrcoef'])

model_results
```



```
                 Model  Accuracy  Precision  Sensitivity  Specificity  F1 Score     ROC   Log_Loss  mathew_corrcoef
0  K-Nearest Neighbors  0.796296   0.787879     0.866667     0.866667  0.825397  0.7875  19.188653         0.586094
```

**K-Nearest Neighbors Confusion Matrix**

```
Y_pred_knn = np.around(Y_pred_knn)

print(metrics.classification_report(Y_test,Y_pred_knn))
```

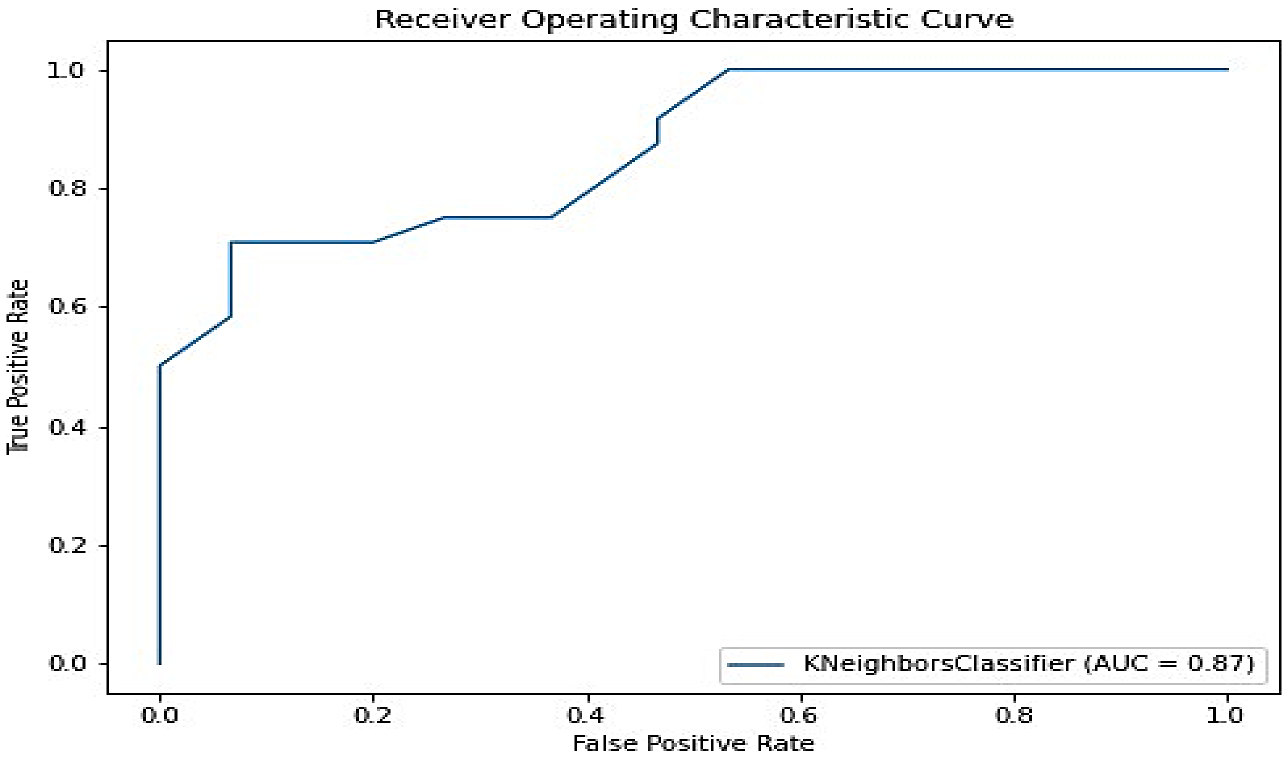


```
Y_pred_knn = np.around(Y_pred_knn)
print(metrics.classification_report(Y_test,Y_pred_knn))
```

```
              precision    recall  f1-score   support

           1       0.79      0.87      0.83        30
           2       0.81      0.71      0.76        24

    accuracy                           0.80        54
   macro avg       0.80      0.79      0.79        54
weighted avg       0.80      0.80      0.79        54
```

**K-Nearest Neighbors Classification Report**

```
plot_roc_curve(knn_classifier,X_test,Y_test)

plt.xlabel('False Positive Rate')

plt.ylabel('True Positive Rate')

plt.title('Receiver Operating Characteristic Curve')
```

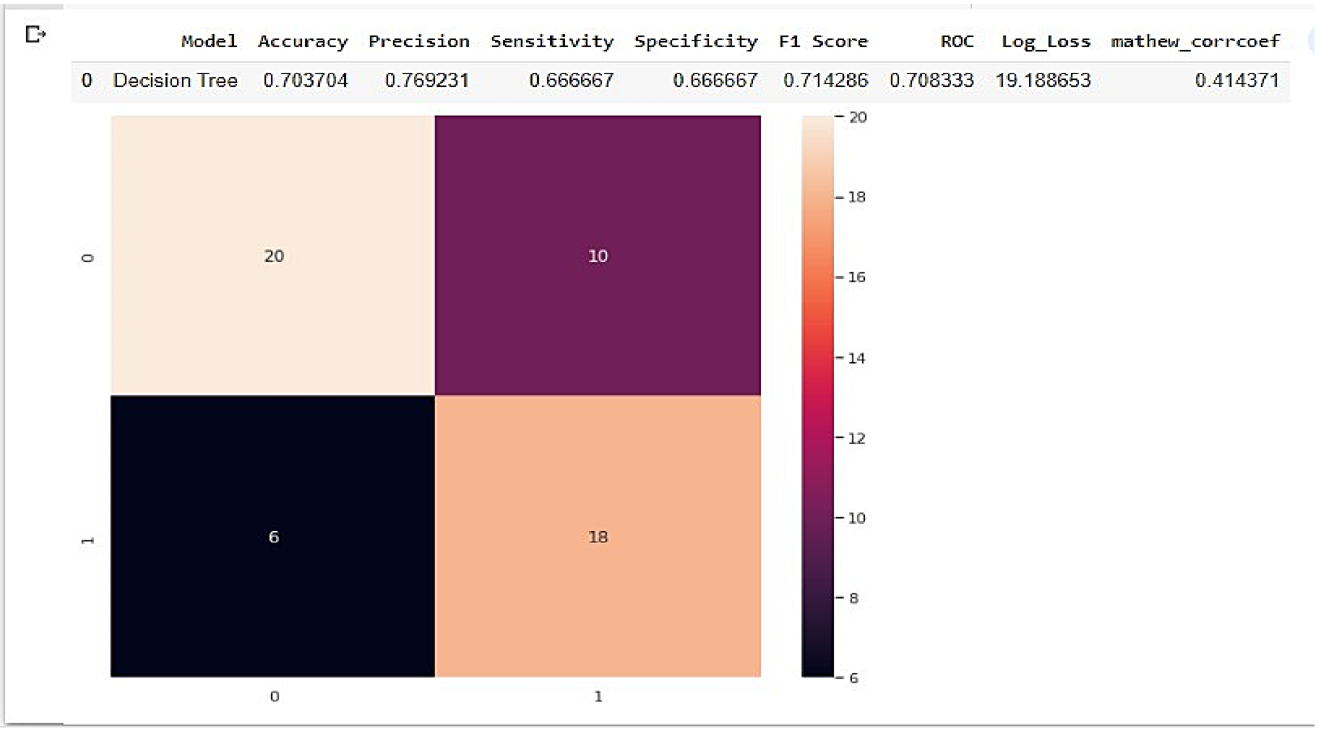


**K-Nearest Neighbors ROC Curve**

### 8.5.3 Decision Tree Classifier

```
 y_pred_dte = dt_classifier.predict(X_test)

plt.figure(figsize=(10, 8))

CM=confusion_matrix(Y_test,y_pred_dte)

sns.heatmap(CM, annot=True)

TN = CM[0][0] FN = CM[1][0] TP = CM[1][1]

FP = CM[0][1]

specificity = TN/(TN+FP)

 loss_log = log_loss(Y_test, y_pred_dte)

acc= accuracy_score(Y_test, y_pred_dte)

 roc=roc_auc_score(Y_test, y_pred_dte)

 prec = precision_score(Y_test, y_pred_dte)
```

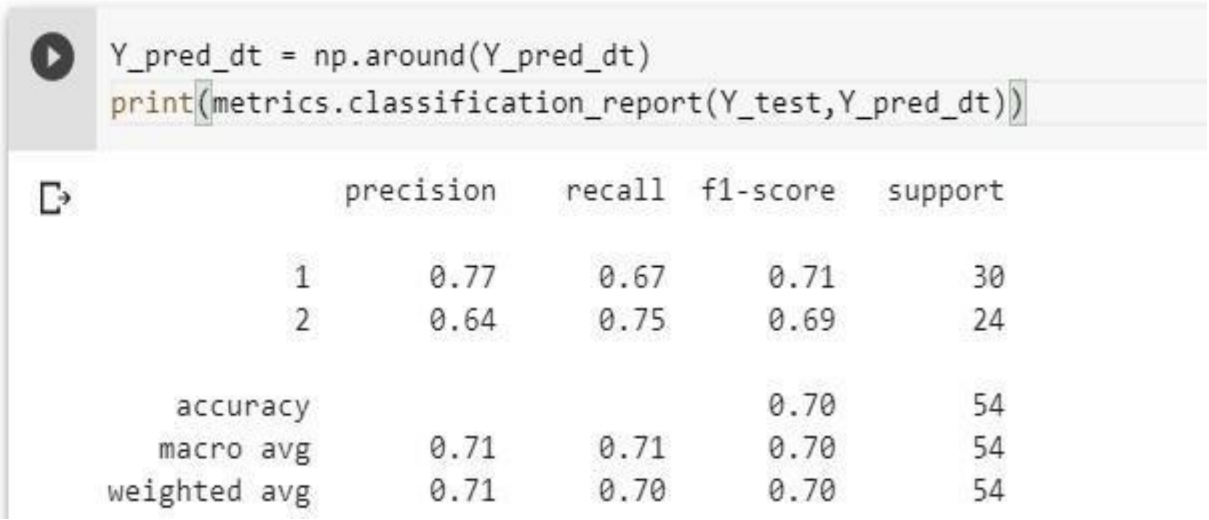
```
rec = recall_score(Y_test, y_pred_dte)

f1 = f1_score(Y_test, y_pred_dte)

mathew = matthews_corrcoef(Y_test, y_pred_dte)

model_results =pd.DataFrame([['Decision Tree',acc, prec,rec,specificity, f1,roc,
loss_log,mathew]], columns = ['Model', 'Accuracy','Precision', 'Sensitivity','Specificity', 'F1
Score','ROC','Log_Loss','mathew_corrcoef'])

model_results
```



|   | Model | Accuracy | Precision | Sensitivity | Specificity | F1 Score | ROC | Log_Loss | mathew_corrcoef |
|---|---|---|---|---|---|---|---|---|---|
| 0 | Decision Tree | 0.703704 | 0.769231 | 0.666667 | 0.666667 | 0.714286 | 0.708333 | 19.188653 | 0.414371 |

**Decision Tree Confusion Matrix**

```
Y_pred_dt = np.around(Y_pred_dt)

print(metrics.classification_report(Y_test,Y_pred_dt))
```

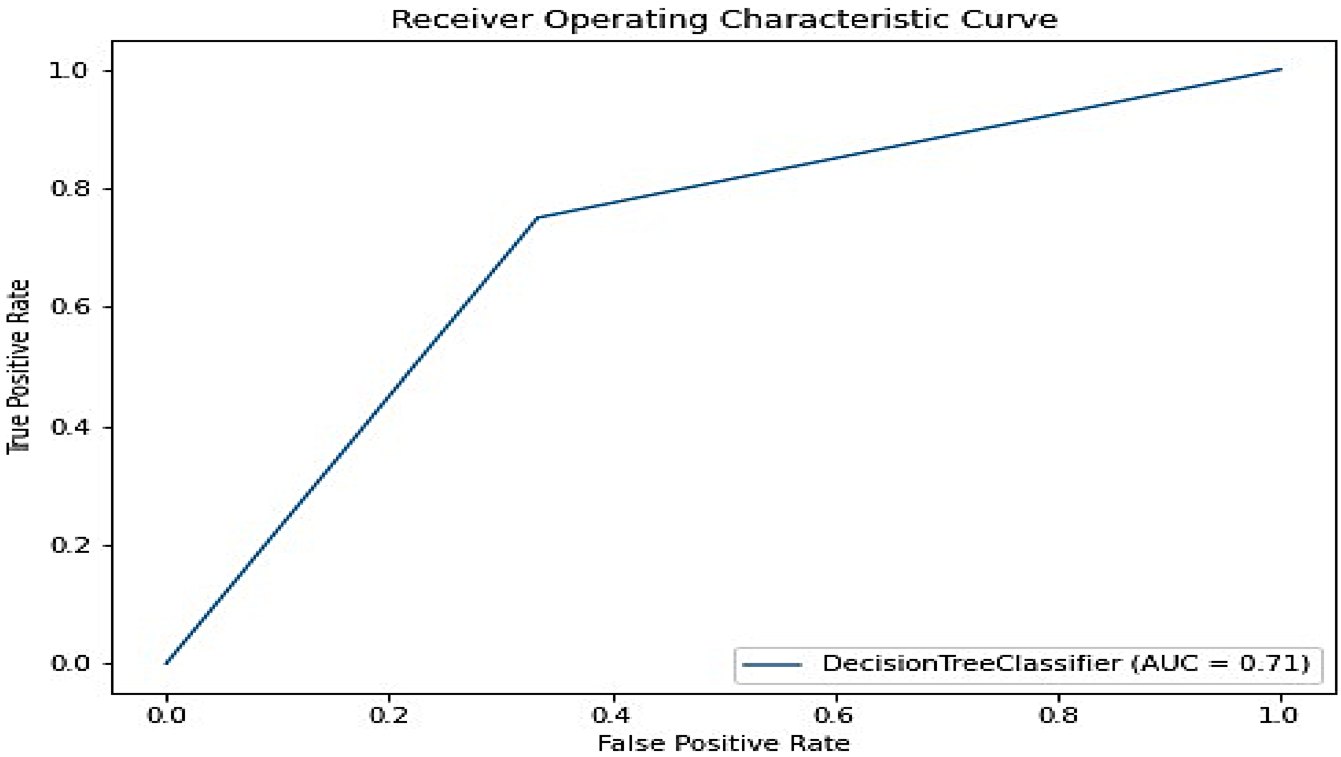


```
Y_pred_dt = np.around(Y_pred_dt)
print(metrics.classification_report(Y_test,Y_pred_dt))
```

```
              precision    recall  f1-score   support

           1       0.77      0.67      0.71        30
           2       0.64      0.75      0.69        24

    accuracy                           0.70        54
   macro avg       0.71      0.71      0.70        54
weighted avg       0.71      0.70      0.70        54
```

**Decision Tree Classification Report**

```
plot_roc_curve(dt_classifier,X_test,Y_test)

plt.xlabel('False Positive Rate')

 plt.ylabel('True Positive Rate')

plt.title('Receiver Operating Characteristic Curve')
```



**Decision Tree ROC Curve**

### 8.5.4 Naive Bayes Classifier

y_pred_nbe = nb_classifier.predict(X_test)

plt.figure(figsize=(10, 8))

CM=confusion_matrix(Y_test,y_pred_nbe)

sns.heatmap(CM, annot=True)

TN = CM[0][0] FN = CM[1][0]

TP = CM[1][1] FP = CM[0][1]

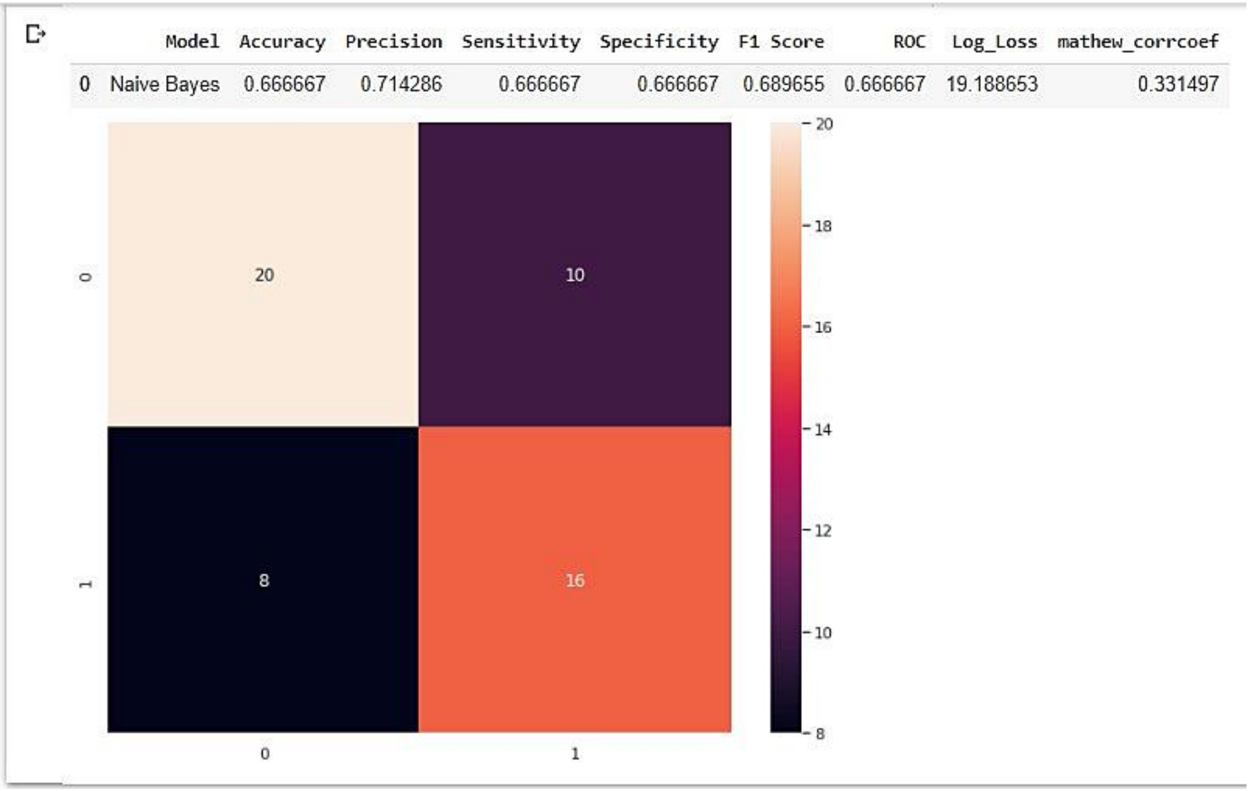
specificity = TN/(TN+FP) l

oss_log = log_loss(Y_test, y_pred_nbe)

acc= accuracy_score(Y_test, y_pred_nbe) roc=roc_auc_score(Y_test, y_pred_nbe)

prec = precision_score(Y_test, y_pred_nbe)

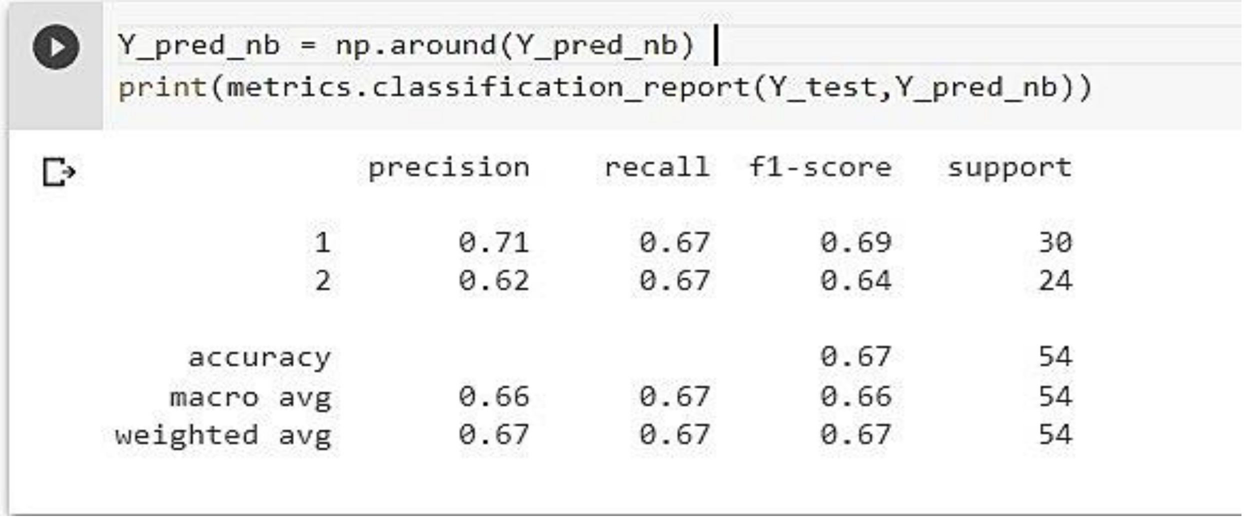
```
 rec = recall_score(Y_test, y_pred_nbe)

f1 = f1_score(Y_test, y_pred_nbe)

 mathew = matthews_corrcoef(Y_test, y_pred_nbe)

model_results =pd.DataFrame([['Naive Bayes ',acc, prec,rec,specificity, f1,roc, loss_log,mathew]],
columns = ['Model', 'Accuracy','Precision', 'Sensitivity','Specificity', 'F1
Score','ROC','Log_Loss','mathew_corrcoef'])

model_results
```



|  | Model | Accuracy | Precision | Sensitivity | Specificity | F1 Score | ROC | Log_Loss | mathew_corrcoef |
|---|---|---|---|---|---|---|---|---|---|
| 0 | Naive Bayes | 0.666667 | 0.714286 | 0.666667 | 0.666667 | 0.689655 | 0.666667 | 19.188653 | 0.331497 |

**Naive Bayes Confusion Matrix**

```
Y_pred_nb = np.around(Y_pred_nb)

print(metrics.classification_report(Y_test,Y_pred_nb))
```

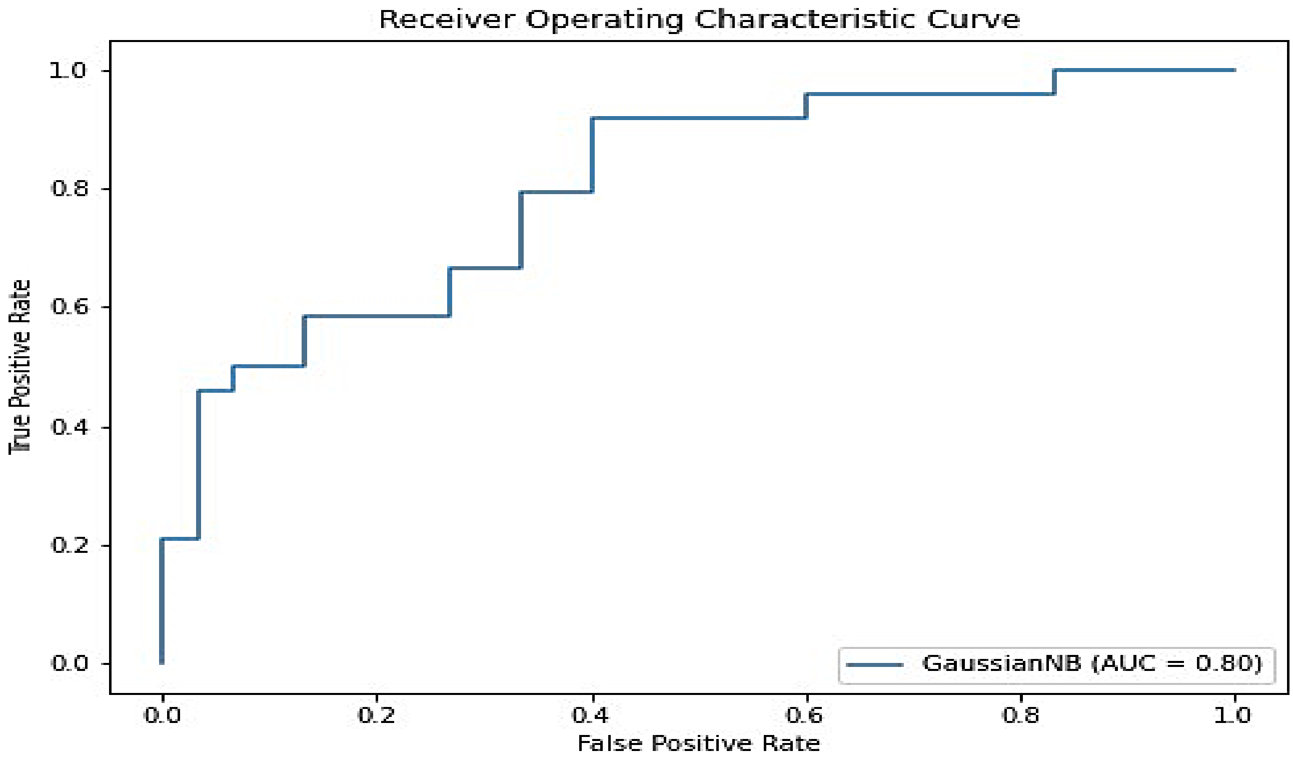


```python
Y_pred_nb = np.around(Y_pred_nb)
print(metrics.classification_report(Y_test,Y_pred_nb))
```

```
              precision    recall  f1-score   support

           1       0.71      0.67      0.69        30
           2       0.62      0.67      0.64        24

    accuracy                           0.67        54
   macro avg       0.66      0.67      0.66        54
weighted avg       0.67      0.67      0.67        54
```

**Naive Bayes Classification Report**

```python
plot_roc_curve(nb_classifier,X_test,Y_test)

plt.xlabel('False Positive Rate')

plt.ylabel('True Positive Rate')

plt.title('Receiver Operating Characteristic Curve')
```



**Naive Bayes ROC Curve**

# RESULTS

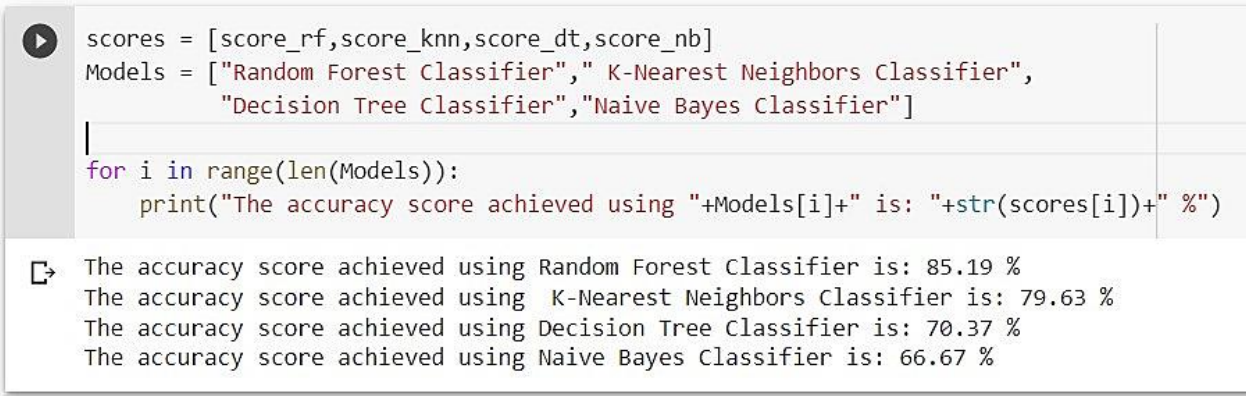
# CHAPTER-9
# RESULTS

## 9.1 Performance Metrics

### Final Result

| Model | Accuracy | Precision | Sensitivity | Specificity | F1 Score | ROC | Log_Loss | Mathew_correcoef |
|---|---|---|---|---|---|---|---|---|
| Random Forest | 0.8519 | 0.8667 | 0.8667 | 0.8667 | 0.8667 | 0.85 | 19.1886 | 0.7 |
| KNN | 0.7963 | 0.7879 | 0.8667 | 0.8667 | 0.8254 | 0.7875 | 19.1886 | 0.5861 |
| Decision Tree | 0.7037 | 0.7692 | 0.6667 | 0.6667 | 0.7142 | 0.7083 | 19.1886 | 0.4144 |
| Naive Bayes | 0.6667 | 0.7143 | 0.6667 | 0.6667 | 0.6896 | 0.6667 | 19.1886 | 0.3315 |

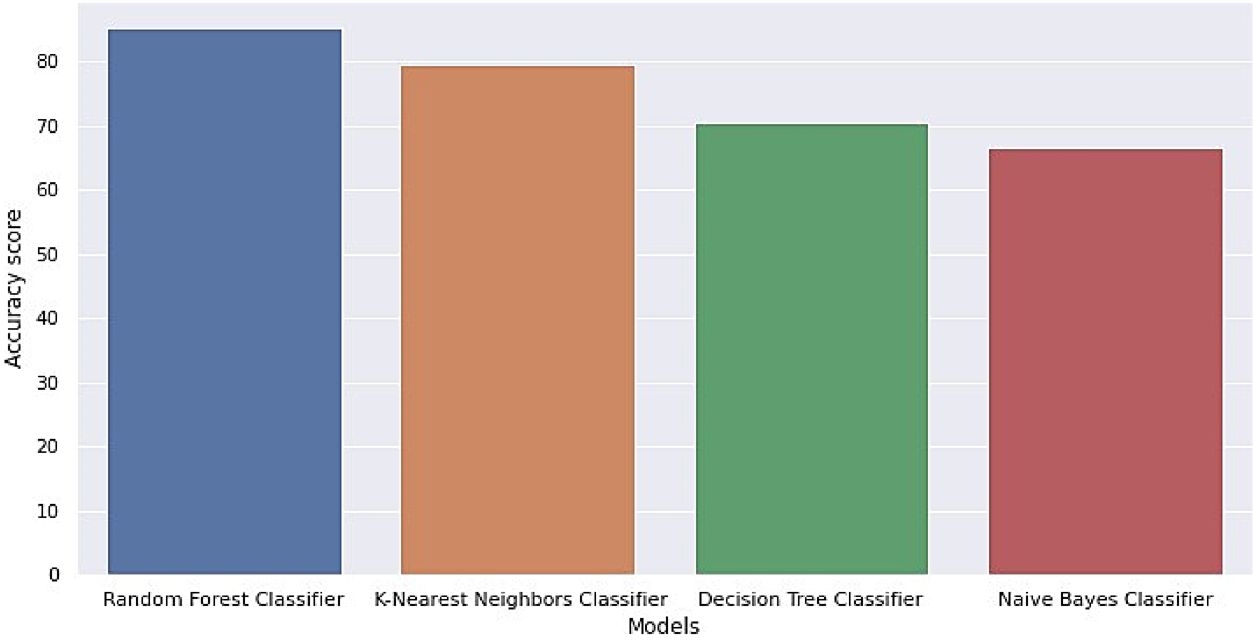
## Final Accuracy Score



```
scores = [score_rf,score_knn,score_dt,score_nb]
Models = ["Random Forest Classifier"," K-Nearest Neighbors Classifier",
          "Decision Tree Classifier","Naive Bayes Classifier"]

for i in range(len(Models)):
    print("The accuracy score achieved using "+Models[i]+" is: "+str(scores[i])+" %")
```

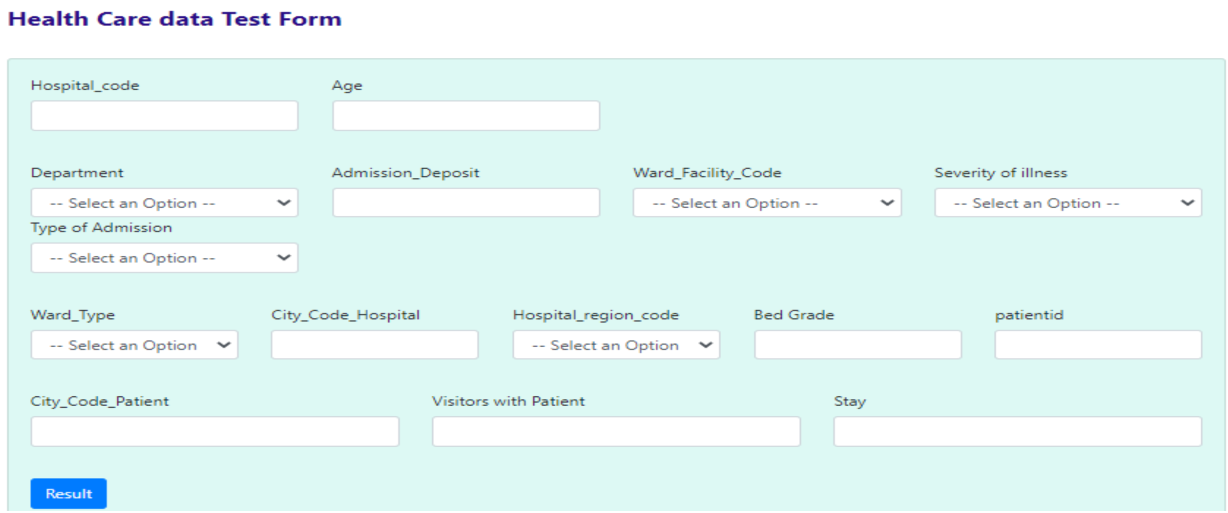
```
The accuracy score achieved using Random Forest Classifier is: 85.19 %
The accuracy score achieved using  K-Nearest Neighbors Classifier is: 79.63 %
The accuracy score achieved using Decision Tree Classifier is: 70.37 %
The accuracy score achieved using Naive Bayes Classifier is: 66.67 %
```

## Accuracy Score Bar Graph

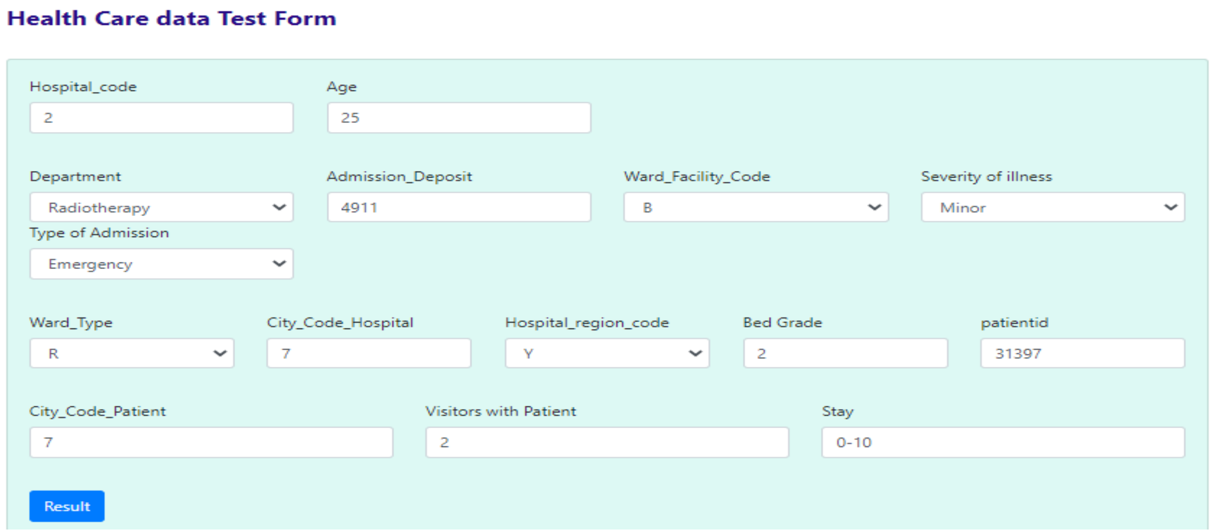


# Snapshots:

## Health-care data Test Form:



## Health-care data Test Form:



Hospital_code: 2

Age: 25

Department: Radiotherapy

Admission_Deposit: 4911

Ward_Facility_Code: B

Severity of illness: Minor

Type of Admission: Emergency

Ward_Type: R

City_Code_Hospital: 7

Hospital_region_code: Y

Bed Grade: 2

patientid: 31397

City_Code_Patient: 7

Visitors with Patient: 2

Stay: 0-10

# CHAPTER-10

# ADVANTAGES & DISADVANTAGES

## 10.1 ADVANTAGES

✓ User can search for doctor’s help at any point of time.

✓ User can talk about their heart disease and get instant diagnosis.

✓ Doctors get more clients online.

✓ Very useful in case of emergency

## 10.2 DISADVANTAGES

- ✓ This would be the primary usage of predictive analytics in healthcare – diagnosing and treating a disease before it causes larger problems.
- ✓ As this new industry matures, the disadvantages are likely to be outweighed by the advantages, presenting a new standard for care
- ✓ Risk related to alteration of data that may be used to make wrong healthcare decisions

# CHAPTER-11

# CONCLUSION

## 11.1 CONCLUSION

In conclusion, the barriers to healthcare access are not only moral, but they can be financial, or based on policies as well. While it may be easy to say that everyone should have the right to free healthcare, the situation is not that simple.

The book mentions that “The United States lags far behind many other postindustrial societies in adopting social policies…” People in European countries, for example, can enjoy the benefit of paid parental leave and guaranteed childcare, while citizens in the US still have to fight over days off and vacation time. As stated on a previous page, many European countries have adopted many forms of social policy- they have given their citizen Universal Healthcare. We, in the US, do not have that. The Unites States has been reluctant to make use of universal social policies, instead relying on formal policies such as health insurance companies. The American Medical Association, as mentioned in the textbook in chapter 16, has been “vigorously opposed (to the) establishment of a national health care system such as those founded in most other countries” because it will destroy insurance companies and make the government the body in charge of healthcare. As a result of our fractured healthcare system, we now have what the textbook refers to, also in chapter 16, Health and Medicine, as a “health disparity” between the population.

# CHAPTER-12

# FUTURE SCOPE

## 12.1 FUTURE SCOPE

Each new wave of technological innovation has produced an exponential growth of data volume for health care organizations. Increase in data volume has led to an increase in the need to capture and process it. Along with the lack of a streamlined process, consolidation within the health care industry has exacerbated access and availability problems. Government compliance and coding requirements have contributed to make this a “perfect storm” for analytics in the health care industry.

Furthermore, organizations view problems around widespread implementation and adoption as insurmountable obstacles. Fortunately, industry stakeholders are starting to see solutions and advancements that may produce a sea-change

# CHAPTER-13

# APPENDIX

## 13. APPENDIX

**Python:** Python is an interpreted, high-level, general purpose programming language created by Guido Van Rossum and first released in 1991, Python's design philosophy emphasizes code Readability with its notable use of significant White space. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects. Python is dynamically typed and garbage collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming.

**Sklearn:** Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modelling including classification, regression, clustering and dimensionality reduction via a consistent interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib.

**Numpy:** NumPy is a library for the python programming language, adding support for large, multi- dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.

The ancestor of NumPy, Numeric, was originally created by Jim with contributions from several other developers.

**Librosa:** Librosa is a Python package for music and audio analysis. Librosa is basically used when we work with audio data like in music generation (using LSTMs), Automatic Speech Recognition. It provides the building blocks necessary to create the music information retrieval systems. Librosa helps to visualize the audio signals and also do the feature extractions in it using different signal processing techniques.

**Matplotlib:** Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an objectoriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK. There is also a procedural "pylab" interface based on a statemachine (like OpenGL), designed to closely resemble that of MATLAB, though its use is discouraged.

**Seaborn:** Seaborn is a Python data visualization library based on matplotlib. It provides a highlevel interface for drawing attractive and informative statistical graphics. Seaborn is a library in Python predominantly used for making statistical graphics. Seaborn is a data visualization library built on top of matplotlib and closely integrated with pandas data structures in Python. Visualization is the central part of Seaborn which helps in exploration and understanding of data.

**SciPy:** SciPy contains modules for optimization, linearalgebra, integration, interpolation, special functions, FFT, signal and imageprocessing, ODE solvers and other tasks common in science and engineering. SciPy is also a family of conferences for users and developers of these tools: SciPy (in the

United States), EuroSciPy (in Europe) and SciPy.in (in India). Enthought originated the SciPy conference in the United States and continues to sponsor many of the international conferences as well as host the SciPy website. SciPy is a scientific computation library that uses NumPy underneath. It provides more utility functions for optimization, stats and signal process

# Analytics for Hospitals Health-Care data Source code:

## Code for Checking null values:

```
#importing datasets using pandas

import pandas as pd

#using read_csv function

d= pd.read_csv("D:/dataset/train_data.csv"):

# get it by using info() keyword
d.info()

#purpose of isnull() to checking null values in the data set

d.isnull()

d.isnull().sum()
```

### Code for performing Exploratory Data Analysis:

```
import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

%matplotlib inline

df= pd.read_csv("D:/dataset/train_data.csv")

df
```

```
#use of head() function to read the values in first 5 rows
df.head()
#use of tail() function to read the values in last 5 rows
df.tail()

df.info()
#use of dtypes to represent type of data
df.dtypes
#use of shape() to represent total rows and columns in 2D manner.
df.shape
df.isnull() .sum() .sum()
#check for null values
df.isnull()
#describe the values
df.describe()
#correlation values
df.corr()
#change null values by mean
df['Bed Grade'].fillna(df['Bed Grade'].mean(),inplace=True)
df['Bed Grade'].isnull() .sum()

df.isnull().sum()
```

```
#change the null value into mean value
df["City_Code_Patient"].fillna(df["City_Code_Patient"].mean(),inplace=True)

df["City_Code_Patient"].isnull().sum()

df.cov()



sns.heatmap (df.corr(),annot=True)

plt.title("correlation Matrix")

plt.show()



df["Admission_Deposit"].hist(bins=10)

plt.title("Histogram for Admission_Deposit ")

plt.show()



df["Ward_Type"].hist(bins=10)

plt.title("Histogram for Ward_Type ")

plt.show()



df["patientid"].hist(bins=100)

plt.title("Histogram for patientid ")

plt.show()
```

## Model creation of dataset:

```
import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

np.set_printoptions(suppress=True)

import warnings

warnings.filterwarnings('ignore')


#load data
d1 = pd.read_csv('/content/drive/My Drive/Healthcare_Data/sample_sub.csv')

d2 = pd.read_csv('/content/drive/My Drive/Healthcare_Data/train_data_dictionary.csv')

test = pd.read_csv('/content/drive/My Drive/Healthcare_Data/test_data.csv')

train = pd.read_csv('/content/drive/My Drive/Healthcare_Data/train_data.csv')


from google.colab import drive

drive.mount('/content/drive')
```

```
train.head()


train.info()

train.Stay.unique()


# NA values in train dataset :

train.isnull().sum().sort_values(ascending = False)


# NA values in test dataset :
test.isnull().sum().sort_values(ascending = False)
# Dimension of train dataset

train.shape
# Dimension of test dataset
test.shape



# Number of distinct observations in train dataset
for i in train.columns:

    print(i, ':', train[i].nunique())


# Number of distinct observations in test dataset
for i in test.columns:
    print(i, ':', test[i].nunique())
```

# Data Preparation :

```
#Replacing NA values in Bed Grade Column for both Train and Test datssets

train['Bed Grade'].fillna(train['Bed Grade'].mode()[0], inplace = True)

test['Bed Grade'].fillna(test['Bed Grade'].mode()[0], inplace = True)

#Replacing NA values in  Column for both Train and Test datssets

train['City_Code_Patient'].fillna(train['City_Code_Patient'].mode()[0], inplace = True)

test['City_Code_Patient'].fillna(test['City_Code_Patient'].mode()[0], inplace = True)

# Label Encoding Stay column in train dataset

from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()

train['Stay'] = le.fit_transform(train['Stay'].astype('str'))

train.head()


#Imputing dummy Stay column in test datset to concatenate with train dataset
test['Stay'] = -1

df = pd.concat([train, test])

df.shape


#Label Encoding all the columns in Train and test datasets
for i in ['Hospital_type_code', 'Hospital_region_code', 'Department',
        'Ward_Type', 'Ward_Facility_Code', 'Type of Admission', 'Severity of Illness', 'Age']:
```

```
    le = LabelEncoder()

    df[i] = le.fit_transform(df[i].astype(str))

#Spearating Train and Test Datasets

train = df[df['Stay']!=-1]

test = df[df['Stay']==-1]
```

**Feature Engineering**

```
def get_countid_enocde(train, test, cols, name):

  temp = train.groupby(cols)['case_id'].count().reset_index().rename(columns = {'case_id': name})

  temp2 = test.groupby(cols)['case_id'].count().reset_index().rename(columns = {'case_id': name})

  train = pd.merge(train, temp, how='left', on= cols)

  test = pd.merge(test,temp2, how='left', on= cols)

  train[name] = train[name].astype('float')

  test[name] = test[name].astype('float')

  train[name].fillna(np.median(temp[name]), inplace = True)

  test[name].fillna(np.median(temp2[name]), inplace = True)

  return train, test

train, test = get_countid_enocde(train, test, ['patientid'], name = 'count_id_patient')

train, test = get_countid_enocde(train, test,
                                 ['patientid', 'Hospital_region_code'], name = 'count_id_patient_hospitalCode')

train, test = get_countid_enocde(train, test,
                          ['patientid', 'Ward_Facility_Code'], name =
'count_id_patient_wardfacilityCode')


# Droping duplicate columns

test1 = test.drop(['Stay', 'patientid', 'Hospital_region_code', 'Ward_Facility_Code'], axis
=1)

train1 = train.drop(['case_id', 'patientid', 'Hospital_region_code', 'Ward_Facility_Code'],
axis =1)

# Splitting train data for Naive Bayes and XGBoost
X1 = train1.drop('Stay', axis =1)

y1 = train1['Stay']

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X1, y1, test_size =0.20, random_state
=100)
```

**Models**
**Naive Bayes**

```
from sklearn.naive_bayes import GaussianNB

target = y_train.values

features = X_train.values

classifier_nb = GaussianNB()

model_nb = classifier_nb.fit(features, target)

prediction_nb = model_nb.predict(X_test)

from sklearn.metrics import accuracy_score

acc_score_nb = accuracy_score(prediction_nb,y_test)

print("Acurracy:", acc_score_nb*100)
```

## Neural Network

```
# Segregation of features and target variable
X = train.drop('Stay', axis =1)


y = train['Stay']


print(X.columns)


z = test.drop('Stay', axis = 1)


print(z.columns)



# Data Scaling
from sklearn import preprocessing


X_scale = preprocessing.scale(X)


X_scale.shape



X_train, X_test, y_train, y_test = train_test_split(X_scale, y, test_size =0.20, random_state
=100)

import keras

from keras.models import Sequential
from keras.layers import Dense

import tensorflow as tf

from keras.utils import to_categorical

#Sparse Matrix
a = to_categorical(y_train)

b = to_categorical(y_test)

model = Sequential()

model.add(Dense(64, activation='relu', input_shape = (254750, 20)))

model.add(Dense(128, activation='relu'))

model.add(Dense(256, activation='relu'))

model.add(Dense(512, activation='relu'))

model.add(Dense(512, activation='relu'))

model.add(Dense(11, activation='softmax'))

model.summary()


model.compile(optimizer= 'SGD',
          loss='categorical_crossentropy',
          metrics=['accuracy'])
```

## Predictions

```
# Naive Bayes
pred_nb = classifier_nb.predict(test1.iloc[:,1:])

result_nb = pd.DataFrame(pred_nb, columns=['Stay'])

result_nb['case_id'] = test1['case_id']

result_nb = result_nb[['case_id', 'Stay']]
```

```
result_nb['Stay'] = result_nb['Stay'].replace({0:'0-10', 1: '11-20', 2: '21-30', 3:'31-40', 4: '41-50', 5: '51-60', 6: '61-70', 7: '71-80', 8: '81-90', 9: '91-100', 10: 'More than 100 Days'})
```

```
result_nb.head()
```

```
# Neural Network
test_scale = preprocessing.scale(z)
```

```
test_scale.shape
```

## Results

*Naive Bayes*

```
print(result_nb.groupby('Stay')['case_id'].nunique())
```

**Health-care_data-Classifier.html**:

```
<html>

<head>

<!-- Bootstrap CSS -->

<link rel="stylesheet"

href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" integrity="sha384JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN0hoP+VmmDGMN5t9UJ0Z" crossorigin="anonymous">

<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js" integrity="sha3849/reFTGAW83EW2RDu2S0VKaIzap3H66lZH81PoYlFhbGU+6BZp6G7niu735Sk7lN" crossorigin="anonymous"></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js" integrity="sha384B4gt1jrGC7Jh4AgTPSdUtOBvfO8shuf57BaghqFfPlYxofvL8/KUEfYiJOMMV+rV" crossorigin="anonymous"></script>

<title>Health Care Data</title>

</head>

<body>

<!-- Java Script -->

<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js" integrity="sha3849/reFTGAW83EW2RDu2S0VKaIzap3H66lZH81PoYlFhbGU+6BZp6G7niu735Sk7lN" crossorigin="anonymous"></script>
```

```
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js" integrity="sha384B4gt1jrGC7Jh4AgTPSdUtOBvfO8shuf57BaghqFfPlYxofvL8/KUEfYiJOMMV+rV" crossorigin="anonymous"></script>

<!-- Navbar-->

<nav class="navbar navbar-dark" style="background-color: rgb(13, 102, 87);">

<span class="navbar-brand mb-0 h1">Health Care data Test</span>

</nav>

<div class="container">

<br>

<!--Form-->

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

<fieldset>

<legend style="color: rgb(41, 15, 134);"><b>Health  Care data Test Form</b></legend><br>

 <div class="card card-body" style="background-color: rgb(194 245 236 / 56%);">

  <div class="form-group  row">

 <div class="col-sm-3">

<label for="hospital_code">Hospital_code</label>

<input type="number" class="form-control" id=hospital_Code" name="hospital_code" required>

<label for="age">Age</label>

<input type="number" class="form-control" id="age" name="age" required>

</div>

</div>
```

```
<br>
<div class="form-group  row">
<div class="col-sm">
<label for="dp">Department</label>
<select class="form-control" id="dp" name = "dp" required>
<option disabled selected value> -- Select an Option -- </option>
<option value = "1">Surgery</option>
<option value = "2">TB & Chest disease</option>
<option value = "3">Radiotherapy</option>
<option value = "4">Anesthesia</option>
<option value = "5">Gynecology</option>
</select>

<label for="admission_types">Type of Admission</label>
<select class="form-control" id="toa" name = "toa" required>
<option disabled selected value> -- Select an Option -- </option>
<option value = "1">Emergency</option>
 <option value = "2">Trauma</option>
 <option value = "3">Urgent</option>

</select>
</div>
<div class="col-sm">
```

```
<label for="admission_deposit">Admission_Deposit</label>

<input type="number" class="form-control" id="admission_deposit"
name="admission_deposit"  required>

</div>

<div class="col-sm">

<label for="ward_code">Ward_Facility_Code</label>

<select class="form-control" id="ward_code" name="ward_code" required>

<option disabled selected value> -- Select an Option -- </option>

<option value = "1">A</option>

<option value = "2">B</option>

<option value = "3">C</option>

<option value = "4">D</option>

<option value = "5">E</option>

<option value = "6">F</option>

</select>

</div>

<div class="col-sm">

<label for="illness">Severity of illness</label>

<select class="form-control" id="illness" name="illness" required>

<option disabled selected value> -- Select an Option -- </option>

<option value = "1">Extreme</option>

<option value = "2">Minor</option>

 <option value = "3">Moderate</option>
```

```
</select>
</div>
</div>
<br>
<div class="form-group row">
<div class="col-sm">
<label for="ward_type">Ward_Type</label>
<select class="form-control" id="ward_type" name="ward_type" required>
<option disabled selected value> -- Select an Option -- </option>
<option value = "1">P</option>
<option value = "2">Q</option>
<option value = "3">R</option>
<option value = "4">S</option>
<option value = "5">T</option>
<option value = "6">U</option>
</div>
<div class="col-sm">
<label for="city_code">City_Code_Hospital</label>
<input type="number" class="form-control" id="city_code" name="city_code" required>
</div>
<div class="col-sm">
<label for="hospital_region_code">Hospital_region_code</label>
```

```
<select class="form-control" id="hospital_region_code" name="hospital_region_code" required>
<option disabled selected value> -- Select an Option -- </option>
<option value = "1">X</option>
<option value = "2">Y</option>
<option value = "3">Z</option>
</select>
</div>
 <div class="col-sm">
<label for="bed_grade">Bed Grade</label>
<input type="number" step="any" class="form-control" id="bed_grade" name="bed_grade" required>
 </div>
 <div class="col-sm">
<label for="patient_id">patientid</label>
<input type="number" step="any" class="form-control" id="patient_id" name="patient_id" required>
</div>
</div>
<br>
<div class="form-group  row">
<div class="col-sm">
<label for="city_code_patient">City_Code_Patient</label>
```

```
<input type="number" step="any" class="form-control" id="city_code_patient"
name="city_code_patient" required>
</div>
<div class="col-sm">
<label for="visitor_patient">Visitors with Patient</label>
<input type="number" step="any" class="form-control" id="visitor_patient"
name="visitor_patient" required>
</div>
<div class="col-sm">
<label for="stay">Stay</label>
<input type="text" step="any" class="form-control" id="stay" name="stay" required>
</div>
</div>
 <br>
<div class="form-group">
<input class="btn btn-primary" type="submit" value="Result">
</div>

<!--Prediction Result-->
<div id ="result">
<strong style="color:red">{{result}}</strong>
</div>
</div>
```

```
</fieldset>

</form>

</div>

</body>

</html>
```

**Web App Code for Embedding Dashboard of Analytics for Hospitals Health care data:**

**embedded cognos dashboard.html**

```
<!DOCTYPE html>

<html lang="en">

<head>

<title>Analytics for health-care data</title>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css" rel="stylesheet">

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js"></script>

 </head>

<body>

<div class="container-fluid p-5 bg-primary text-white text-center">

 <h1> Analytics for Hospitals Health Care-data </h1>

<p>Health-care data dashboard</p>

<p><iframe
src="https://us1.ca.analytics.ibm.com/bi/?perspective=dashboard&amp;p
```

```
athRef=.my_folders%2FHospitals%2Bhealth%2Bcare%2Fhealthcare_data_d
ashboard&amp;closeWindowOnLastView=true&amp;ui_appbar=false&amp;
ui_navbar=false&amp;shareMode=embedded&amp;action=view&amp;mod
e=dashboard&amp;subView=model000001846be9cc8e_00000000"
width="320" height="200" frameborder="0" gesture="media"
allow="encrypted-media" allowfullscreen=""></iframe>

</p>

</div>

</body>


</html>
```

## Web App Code for Embedding Story of Analytics for Hospitals Health care data:

### embedded cognos story.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Analytics for health-care data </title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css" rel="stylesheet">
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<div class="container-fluid p-5 bg-primary text-white text-center"> <h1>
Analytics for Hospitals Health Care-data</h1>
<p>Story of Health-care data</p>
<p><iframe
src="https://us1.ca.analytics.ibm.com/bi/?perspective=story&amp;pathRef
=.my_folders%2FHospitals%2Bhealth%2Bcare%2FHospital_data_story&am
```

```
p;closeWindowOnLastView=true&amp;ui_appbar=false&amp;ui_navbar=fal
se&amp;shareMode=embedded&amp;action=view&amp;sceneId=model00
00018456abc5e0_00000001&amp;sceneTime=0" width="320" height="200"
frameborder="0" gesture="media" allow="encrypted-media"
allowfullscreen=""></iframe></p>
</div>
</body>
</html>
```

**GITHUB LINK:**

**https://github.com/IBM-EPBL/IBM-Project-45622-1660731319.git**

**PROJECT DEMO LINK:**

**https://drive.google.com/file/d/1sanp3JQhem4RH4WEGHNINMgg2XwGOhTn/view?usp=share_link**