Analytics for Hospital's Health Care Data

IBM–DOCUMENTATION

UNDER THE GUIDANCE OF

Industry Mentor(s) Name : Shivam Shivare, Hari Prabu

Faculty Mentor(s) Name : Sophana Jennifer

TEAM ID : PNT2022TMID36912

SUBMITTED BY:

ENITHA M	211319104009
DIVYA DHARSHINI S	211319104007
BHAGYA LAKSHMI T	211319104005
LOGESHWARI S	211319104012

COMPUTER SCIENCE AND ENGINEERING

P B COLLEGE OF ENGINEERING

ANNA UNIVERSITY: 2019 – 2023

1. INTRODUCTION

1.1 Project Overview :

Predictive analytics is an increasingly important tool in the healthcare field since modern machine learning (ML) methods can use large amounts of available data to predict individual outcomes for patients. For example, ML predictions can help healthcare providers determine the likelihood of disease, aid in the diagnosis, recommend treatment, and predict future wellness. For this project, we chose to focus on a more logistical metric of healthcare, hospital **length-of-stay** (LOS). LOS is defined as the time between hospital admission and discharge measured in days. This incentivizes hospitals to identify patients of high LOS risk at the time of admission. Once identified, patients with high LOS risk can have their treatment plan optimized to minimize LOS. Another benefit is that prior knowledge of LOS can aid in logistics such as room and bed allocation planning.

1.2 Purpose :

The purpose is to accurately predict using the given dataset, we plan to create various graphs and charts to highlight the insights and visualizations. a) Length of stay for each Case of Patients. b) Stay by patient ID using Column Chart. c) Severity of illness by Patient-ID using Tree Map. d) Age, Department wise patient using Table. e) Room availability by Pie Chart. f) Dashboard Creation. g) Department wise no. of Admissions by Waterfall Chart

By using this application in addition to improving patient safety and lowering costs, reducing LOS can release capacity in the system (including beds and staff time) and improve throughput, enabling the hospital to serve more patients.Predicting the LOS of patients is important because it helps hospitals to more effectively manage its resources and patients. Suggesting a cost efficient hospital with bed availability and to get well soon.To receive good care from the hospital, to predict the number of days of stay and to predict the amount of time to get a bed. Getting discharged or cured before the estimated time.For this purpose our objective is to accurately predict the Length of Stay for each patient on a case by case basis so that the Hospitals can use this information for optimal resource allocation and better functioning. The optimal resource allocation will help in better care of patients.

2. LITERATURE SURVEY :

2.1 Existing problem :

The recent Covid-19 Pandemic has raised alarms over one of the most overlooked areas to focus on Healthcare Management.From which doctors and hospital management were not able to handle multiple numbers of patients at the same time. And due to the lack of proper treatment, the patient's conditions used to get worse.

2.2 References :

	PAPER	AUTHOR	YEAR	METHOD AND ALGORITHM	ACCURACY
1	Machine Learning model for predicting the length of stay in the intensive care unit for covid – 19 patients in the eastern province of Saudi AraKawtbia	Dina A. Alabbad, Abdullah M. Almuhaideb, Shikah J. Alsunaidi, her S. Alqudhaihai, Fatimah A. Alamoudi, Maha K. Alhobaishi, Naimah A. Alaqeel, Mohammed S. Alshahrani	2022	To predict the length of stay of a patient. Here we employed four algorithm Random Forest(RF), Gradient Boosting(GB), Extreme Gradient Boosting (XG Boost) and Ensemble Models. Through these experiments, the prediction is done in this algorithm Random Forest gives the highest accuracy when compared to other methods.	94.16%

2	Time-to-event modelling for hospital length of stay prediction for covid – 19 patients	Y. Wen, M.F. Rahman, Y.Zhuang et al, Michael Pokojovy, Honglun Xu, Peter McCaffrey, Alexander Vo, Eric Walser, Scott Moen, Tzu-Liang (Bill) Tseng	2022	This study uses a technique called time-to event modelling which is also known as survival analysis. It uses algorithms like Logistic Regression, Random forests, Support Vector Machines and Decision tree-based methods. Survival analysis is a branch of statistics concerned with analyzing time-to-event data and predicting the probability of the occurrence of an event. The event could be any format	70%
3	Robust length of a stay prediction model for indoor patients	Ayesha Siddiqa, Syed Abbas Zilqurnian Naqvi, Ahsan Naeem, Allah Ditta, Hani Alquahayz, Muhammad Adnan Khan	2021	The length of stay of patients with different diseases is identified. So that the hospital can manage the available resources and new patients get entries for their prompt treatment. Here they use algorithms such as Ridge Regression(RR), Decision Tree Regression	92%

				Extreme Gradient Boosting Regression (XGBR) and Random Forest Regression (RFR). The process like a Raw dataset is processed then exploring the data, Machine learning modelling, performance measuring and selection of robust model based on the performance.	
4	Predicting length of stay in hospital intensive care unit using general admission features	Merhan A. Abd- Elrazek, Ahmed A. Eltahawi, Mohamed H. Abd Elaziz, Mohamed N. Abd- Elwhab	2021	This paper is based on the length of stay of patients in the ICU. Here the data is pre-processed and the dataset is divided into Fold cross-validation. ML techniques used are Neural Networks(NN), Classification Tree(CT), Tree Baggies(TB), Random Forest(RF), Fuzzy Logic(FL), Support Vector Machine(SVM), KNN, Regression Tree(RT) and Naive Bayes(NB).	92%

				Proposed techniques are data acquisition, data preprocessing, data transformation, training and testing	
5	Predicting inpatient length of stay at hospitals using python + big data	Vishal Tien	2020	In this study, the paper describes creating a model that can predict the length of stay for patients upon admission to a hospital. The algorithms used are Logistic Regression, Boosted Decision Tree and Random forest. In this, the APR DRG code, a classification system that classifies patients according to the reason for admission, the severity of illness and the risk of mortality and the APR severity of illness score are the most important feature In predicting the patient's length of stay.	70%

2.3 Problem Statement Definition :

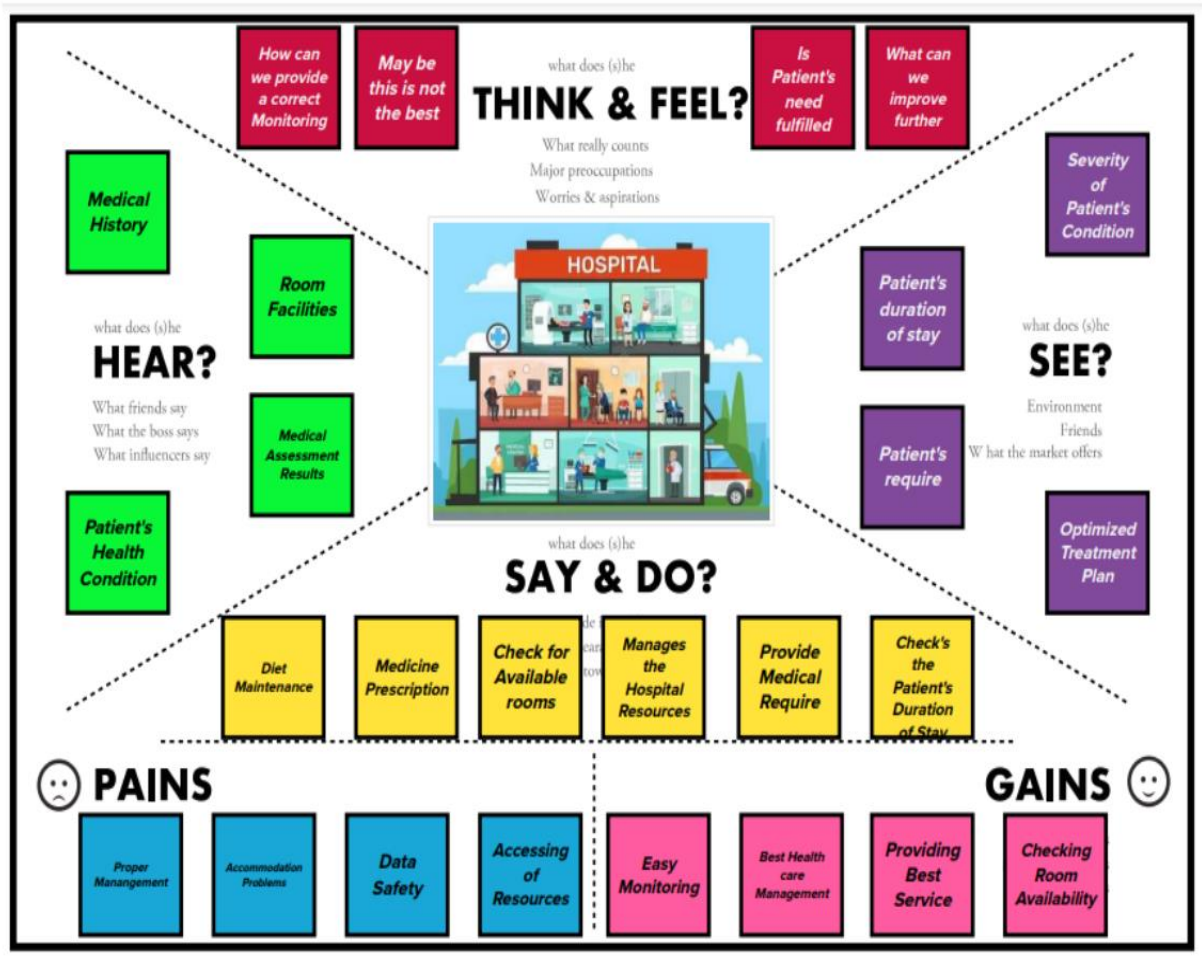
The hospitals need a way to accurately predict the Length of stay for each patient at the time of admission so that the patients with high LOS have their treatment plan optimized on minimum LOS and also the hospital resources such as rooms and beds are efficiently utilized. IN AN ELABORATE MANNER: The recent Covid-19 Pandemic has raised alarms over one of the most overlooked areas to focus on: Healthcare Management. In earlier days, doctors and hospital management were not able to handle multiple numbers of patients at the same time. And due to the lack of proper treatment, the patient's conditions used to get worse. With the help of the application of Data Science in healthcare, it has now become possible to detect the symptoms of a disease at a very early stage. Also, with the advent of various innovative tools and technologies, doctors are able to monitor patients' conditions from remote locations. Data science is an interdisciplinary field that extracts knowledge and insights from structural and unstructured data, using scientific methods, data mining techniques, machine learning algorithms, and big data. The healthcare industry generates large datasets of useful information on patient demography, treatment plans, results of medical examinations, insurance, etc. While healthcare 9 management has various use cases for using data science, patient length of stay is one critical parameter to observe and predict if one wants to improve the efficiency of healthcare management in a hospital. This parameter helps hospitals to identify patients of high LOS risk (patients who will stay longer) at the time of admission. Once identified, patients with high LOS risk can have their treatment plan optimized to minimize LOS

and lower the chance of staff/visitor infection. Also, prior knowledge of LOS can aid in logistics such as room and bed allocation planning. Suppose you have been hired as a Data Scientist of Health Man – a not-for-profit organization dedicated to managing the functioning of Hospitals in a professional and optimal manner.

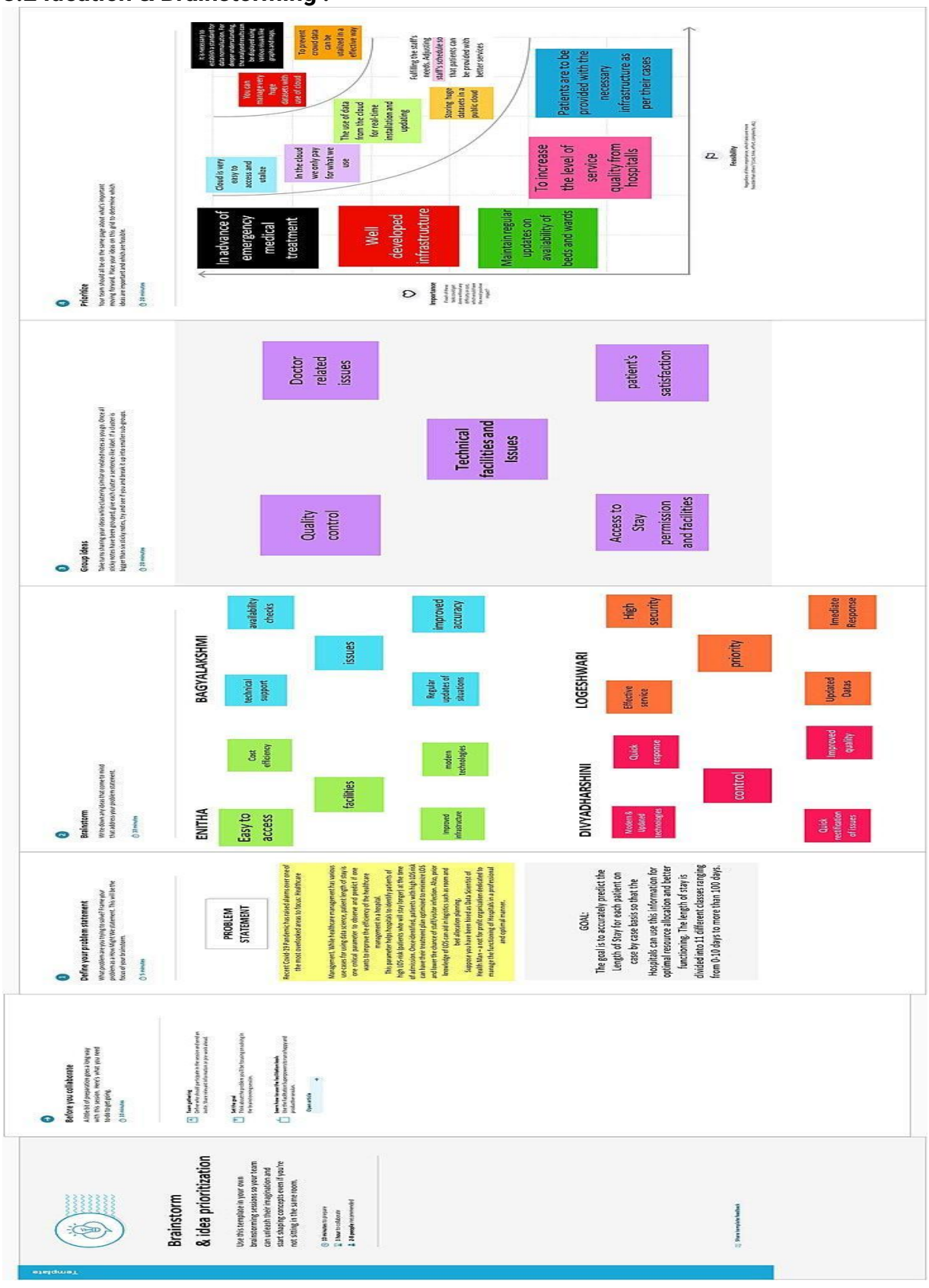
3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas :

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviors and attitudes.It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.



3.2 Ideation & Brainstorming :



3.3 Proposed Solution :

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	During the covid-19 pandemic, we have faced one of the most difficult times of our life. Everyone seeks to survive the great disaster. At the time of the pandemic, none get to know which hospital has vacant beds(free beds) to admit themselves or others infected by covid-19. This situation made the death rate higher.
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, improve efficiencies, and so on.
3.	Novelty / Uniqueness	Healthcare data frequently resides in several locations. The Collected data should be stored in a central system (like centralized storage). This data becomes accessible and usable when it is combined into a single, the central system, such as an enterprise data warehouse (EDW). Uniqueness of our project is that we can able to use data for different things such as which medicine is more effective and for understanding

		the behavioral pattern of a particular disease.
4.	Social Impact / Customer Satisfaction	The application has a drug information a system that accounts for the drug history of the beneficiaries. The system provides up-to-date, accurate medication profiles for improved health planning, evaluation and research. It also includes a comprehensive Drug Utilization Review (DUR) and flags potential interactions with patients medication profile.
5.	Business Model (Revenue Model)	Providers (hospitals) can access the model/application through a subscription service. The minimum subscription period will bean year
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 centralized healthcare services.

3.4 Problem Solution fit :

<table>
<tr>
<td><u>1. Customer Segments</u>
The term includes the Healthcare facilities and services, Patients and Hospital management.</td>
<td><u>6. Customer Limitation</u>
There is no assurity of the most effective allocation or utilization of available resources.</td>
<td><u>5. Available Solution</u>
Text mining , Information retrieval, Information extraction.</td>
</tr>
<tr>
<td><u>2. Problems/Pains</u>
> The length of stay of COVID patients are monitored.
> Demand for the medicine.
> Proper standard of information handling.</td>
<td><u>9. Problem</u>
> Prediction of the occuring situation.
> Efficient less type of calculation.</td>
<td><u>7. Behaviour</u>
Data tracking with that given available methodologies such as text mining and information retrieval and information extraction.</td>
</tr>
<tr>
<td><u>3. Triggers to Act</u>
During the alarming of the pandemic period and the given prevailing emergency situation.</td>
<td rowspan="2"><u>10. Your Solution</u>
The solution is usage of predictive analysis of which is powered by the artificial intelligence which is then used in the analytics technique.</td>
<td><u>8. Channels of Behaviour</u>
1. Online: Usage of data exploration.</td>
</tr>
<tr>
<td><u>4. Emotions</u>
Tensed and perplexed mind set to get rectified from the pandemic period.</td>
<td>2. Offline: Preparing the dataset on the COVID patients.</td>
</tr>
</table>

4. REQUIREMENT ANALYSIS

4.1 Functional requirement :

Following square measure the purposeful needs of the projected resolution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form
FR-2	User Confirmation	Confirmation via Email or Confirmation via OTP
FR-3	User Login	Login exploitation the given credentials
FR-4	Dataset	Upload the dataset to the dashboard
FR-5	Analysis	Data is pre-processed and clean. Once cleanup the exploration method is dole out
FR-6	Prediction	Machine learning rule is employed for prediction
FR-7	Visualization	Visualization of the prediction is shown within the dashboard created exploitation IBM Cognos Analytics

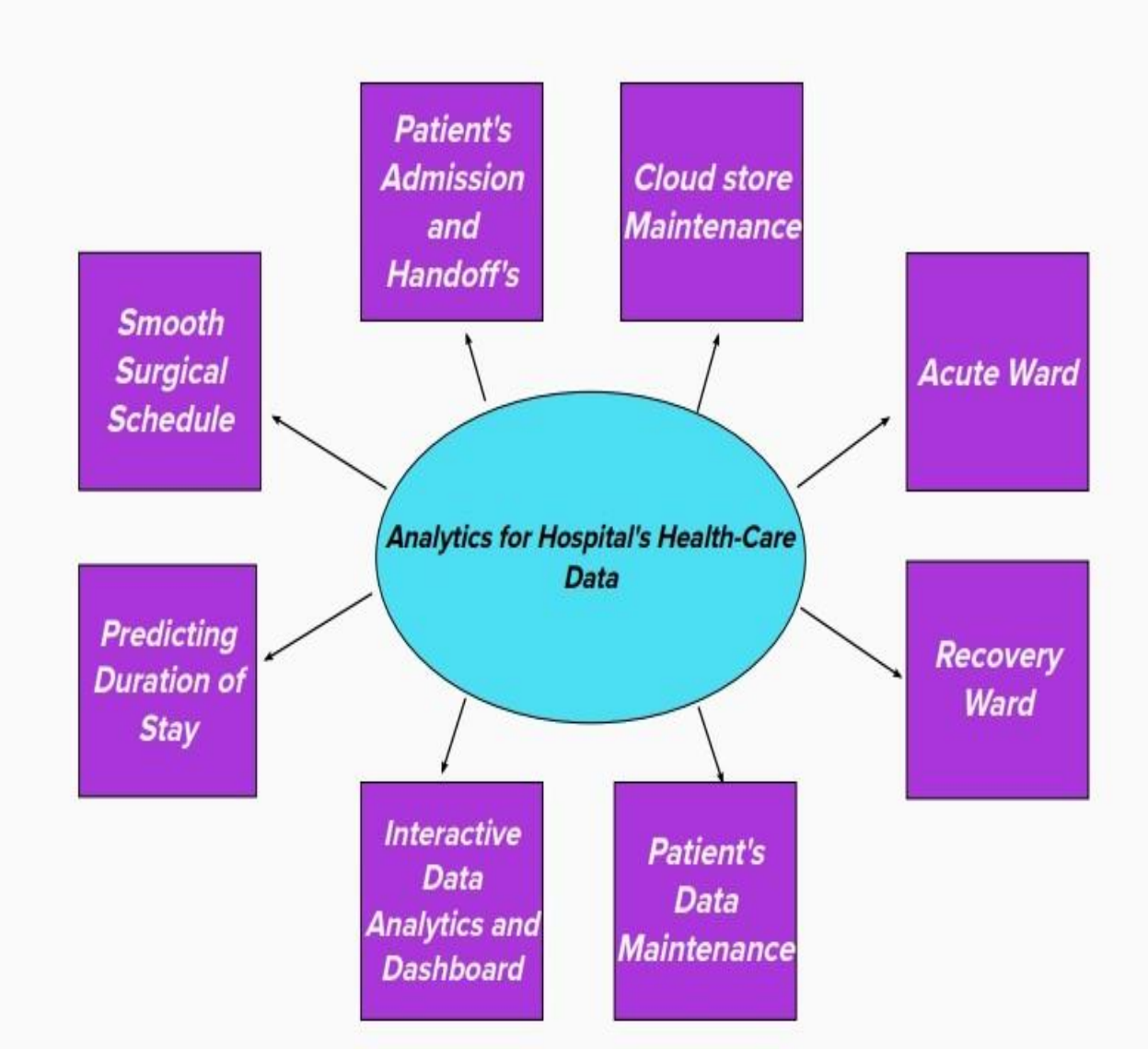
4.2 Non-Functional requirements :

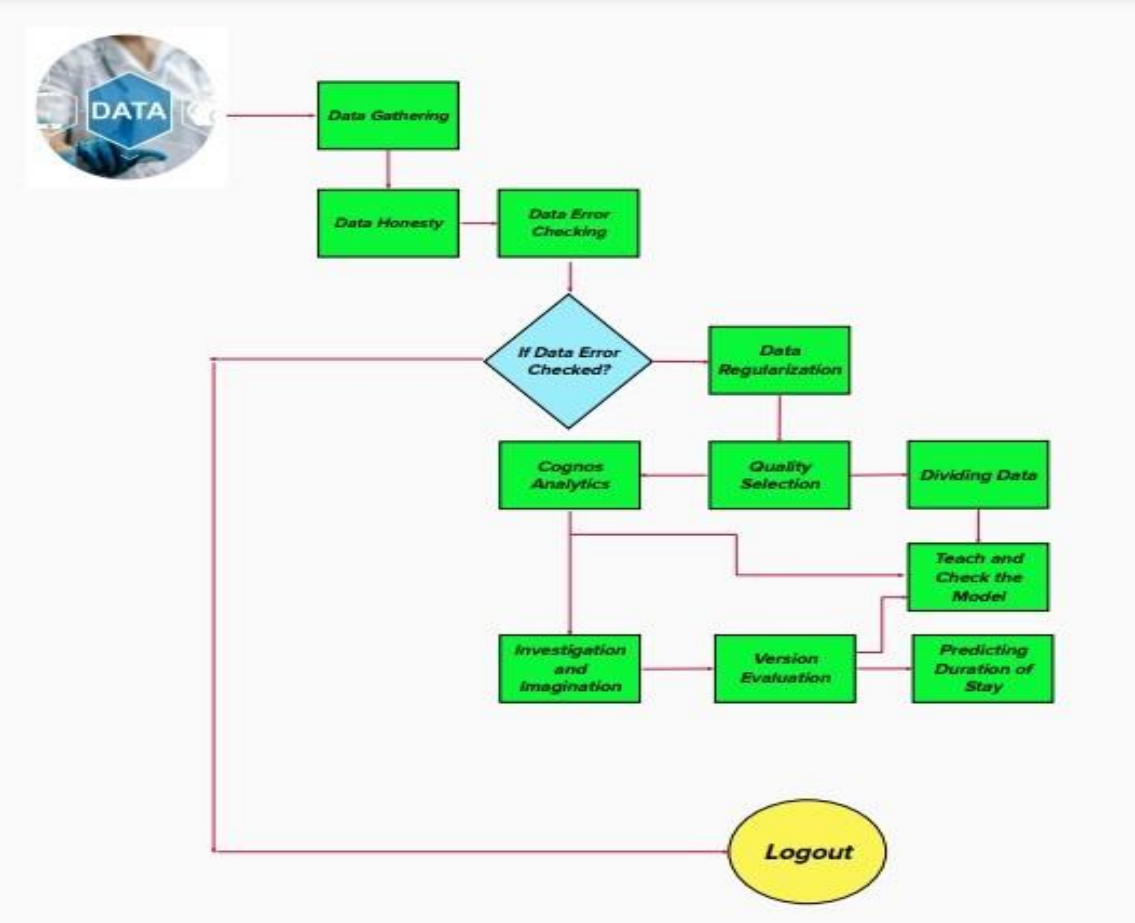
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Dashboards square measure created to show the length of stay prediction in a visual manner. So, the predictions are often simply understood
NFR-2	**Security**	The dataset uploaded to the dashboard can't be downloaded or accessed by external sources.
NFR-3	**Reliability**	A dashboard created once the prediction method can be additional reliable and shows the result clearly and effectively.
NFR-4	**Performance**	the prediction has additional accuracy.

NFR-5	**Availability**	Predicted information are going to be obtainable a few time once the prediction.
NFR-6	**Scalability**	This system can predict the length of stay of all kinds of patients

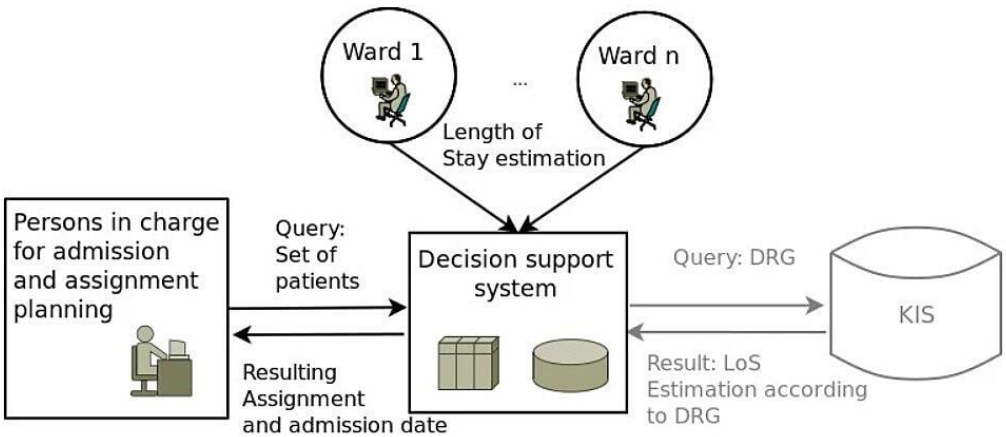
5. PROJECT DESIGN

5.1 Data Flow Diagrams :





5.2 Solution Architecture :



Technical Architecture :

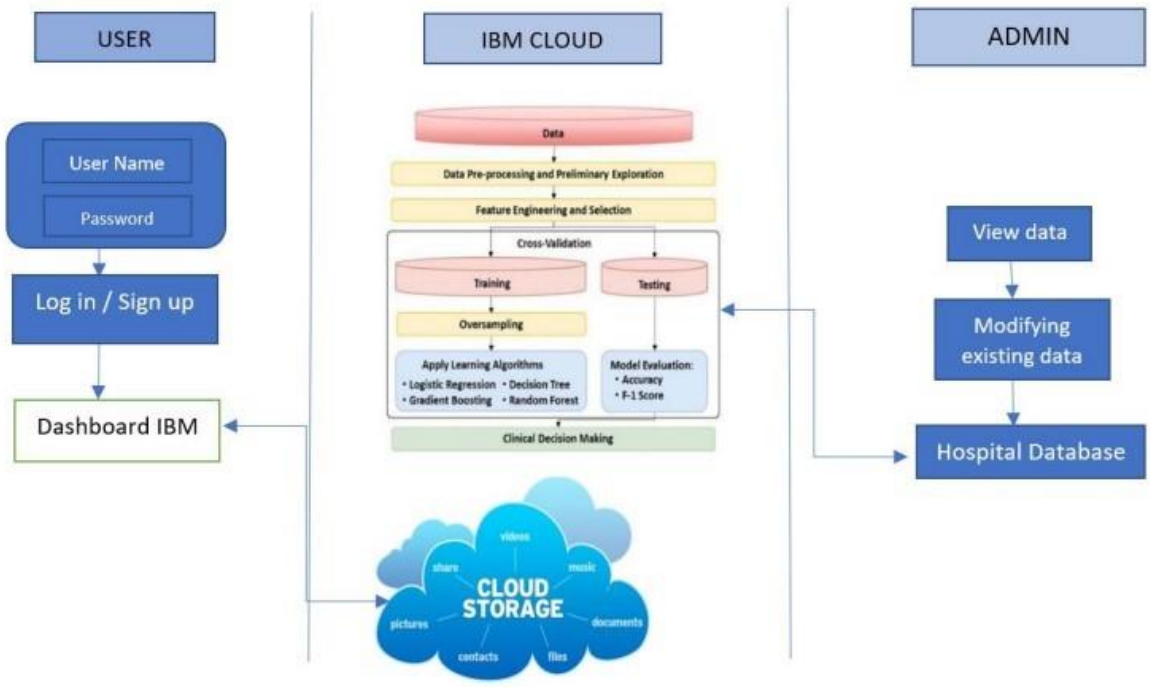


Table-1 : Components & Technologies:

S.NO	Component	Description	Technology
1.	User Interface	The User can interact through the site with their convenient communicative device	HTML, CSS, JavaScript / Angular Js / React Js etc…
2.	Application Logic-1	Collecting the data	CSV File
3.	Application Logic-2	Exploring and Visualizing the data	IBM Cognos Analytics
4.	Database	Data model on the available data	AI or ML
5.	Cloud Database	Storing the Patients data in an cloud environment	IBM Cloud

6.	External API-1	Purpose of External API used in the application	Adhara API, etc..
7.	Machine Learning Model	Predictive analysis on the data model	Regression Model
8.	Infrastructure (Server / Cloud)	Cloud environment for analytics	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	Python
2.	Security Implementations	Admin and User Authorization or Management Authorization	Encryption,IAM Controls,OWASP,etc...
3.	Availability	The data exploring and visualizing is a timely work hence the system should be more available	Cognos Analytics
4.	Performance	The accurate calculation of data is the well-known performance of the system	Predicting analysis using specific ideas

5.3 User Stories :

User Type	Functional Requirement (Epic)	User Story Number	User Story/Task	Acceptance Criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user,I can sign in for the application entering my email,password	I can access my account dashboard	High	Sprint-1
			As a user, I can sign up for the application through google		High	Sprint-1
		USN-2	As a user, I will receive confirmation email	I can click confirmation email	High	Sprint-1
		USN-3	As a user, I can register for the application through Instagram	I can access the dashboard with Instagram Login	Low	Sprint-2
		USN-4	As a user, I can register for the application to know the length of stay of patients	I can access patient's data based on their severeness	High	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering user id & password		High	Sprint-1

		USN-6	As a user I can explore the all the details regarding to hospital in my dashboard		Medium	Sprint-2
Administrationtor	Updating data	USN-7	As a user I can collect the data of all the patients from their attendees and I can store it	I can check the gathered data and can store it	High	Sprint-1
		USN-8	As a Administrator I can categorize the patients based on their risk factors		High	Sprint-1
Customer (Web User)	Accessing the resources	USN-9	As a user I can get all the information in the dashboard	These resources cannot be accessed by others but only me	High	Sprint-1
Customer tools	Tools	USN-10	As a user I can explore the data through data visualization tools like Cognos analytic		High	Sprint-2

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	User Registration	USN-1	Registration through a form.	10	High	Bhagyalakshmi T Divyadharshini S
Sprint-1	User Confirmation	USN-2	Confirmation via E-mail Or Confirmation via OTP	10	High	Enitha M Logeshwari S
Sprint-2	User Login	USN-3	Login exploitation of the given credentials.	10	High	Enitha M Logeshwari S
Sprint-2	Dataset	USN-4	Upload the data set to the dashboard.	10	High	Bhagyalakshmi T Divyadharshini S
Sprint-3	Analysis	USN-5	Data is pre-processed and clean. Once clean up the exploration method is dole out.	20	High	Bhagyalakshmi T Divyadharshin S Enitha M Logeshwari S
Sprint-4	Prediction	USN-6	The machine learning rule is employed for the prediction.	10	High	Bhagyalakshmi T Divyadharshin S
Sprint-4	Visualization	USN-7	Visualization of prediction is shown within the dashboard created exploitation IBM Cognos analytics	10	High	Enitha M Logeshwari S

6.2 Sprint Delivery Schedule :

Project Tracker:

Sprint	Total	Duration	Sprint	Sprint End	Story	Sprint

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

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10}$$

Burn Down Chart:

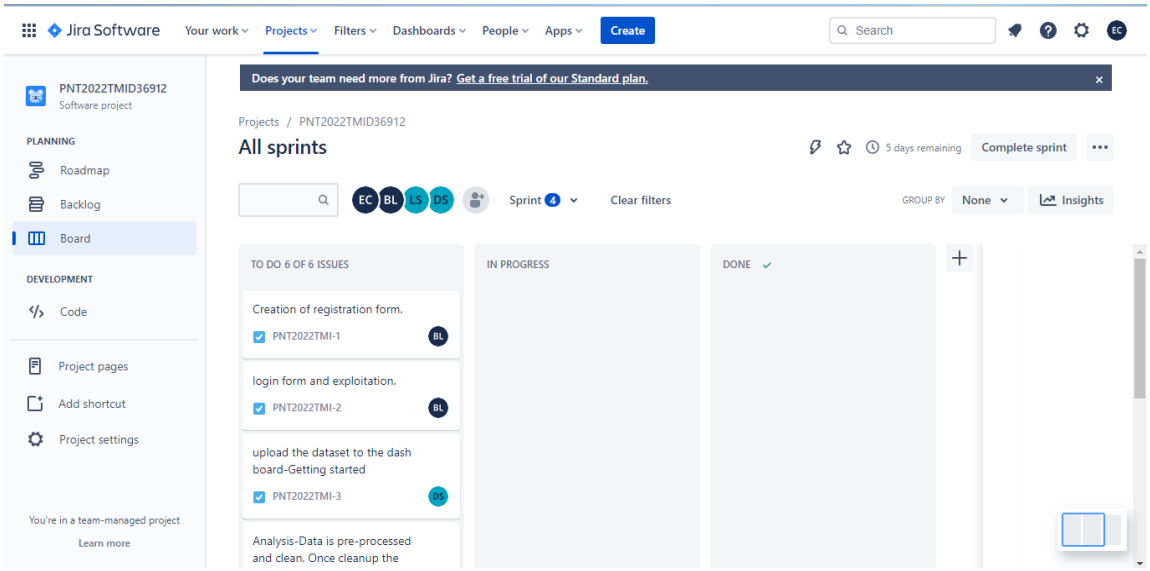
A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.

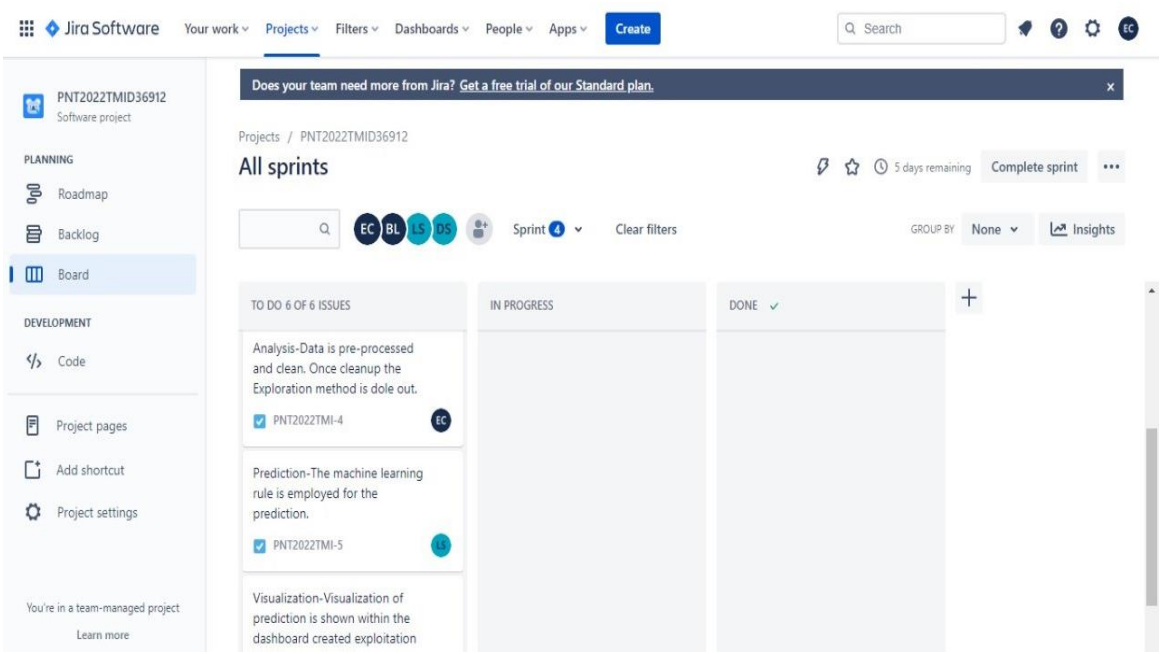
SPRINT	DATE	ESTIMATED EFFORT	ACTUAL EFFORT
SPRINT-1	24-OCT-22	20	20
	25-0CT-22	19	20
	26-OCT-22	18	19
	27-OCT-22	17	19
	28-OCT-22	17	18
	29-OCT-22	16	17
	30-OCT-22	15	15
SPRINT-2	31-OCT-22	14	13
	01-NOV-22	13	12
	02-NOV-22	12	11
	03-NOV-22	11	11
	04-NOV-22	11	11
	05-NOV-22	10	9

	06-NOV-22	9	8
SPRINT-3	07-NOV-22	8	7
	08-NOV-22	7	6
	09-NOV-22	6	6
	10-NOV-22	5	5
	11-NOV-22	5	5
	12-NOV-22	5	5
	13-NOV-22	4	4
SPRINT-4	14-NOV-22	4	4
	15-NOV-22	3	3
	16-NOV-22	2	2
	17-NOV-22	2	2
	18-NOV-22	1	1
	19-NOV-22	1	1

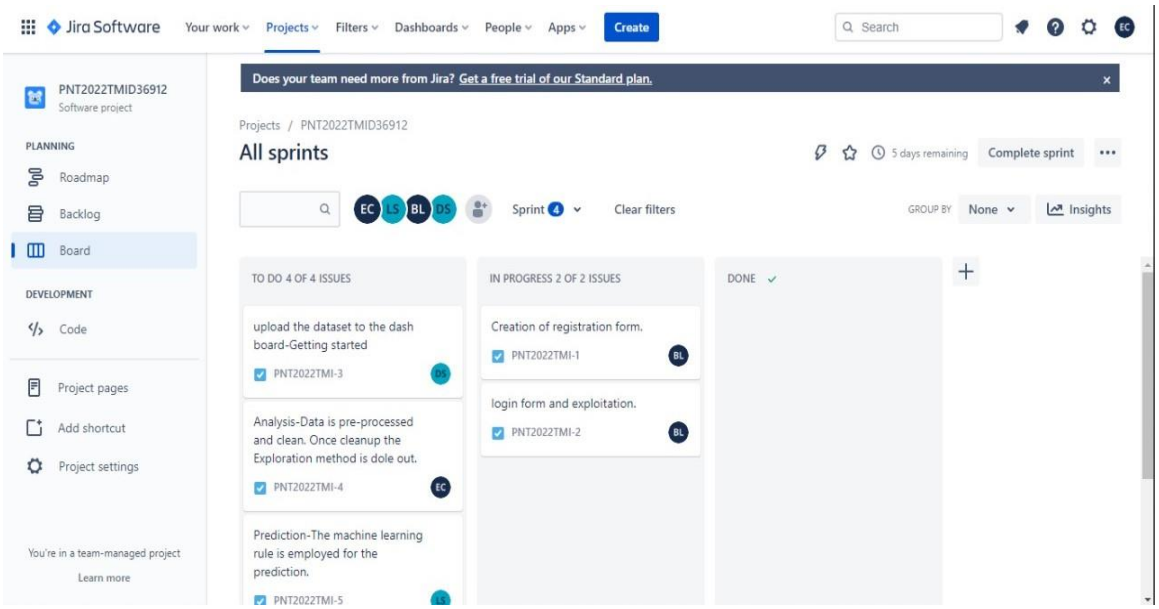
6.3 Reports from JIRA

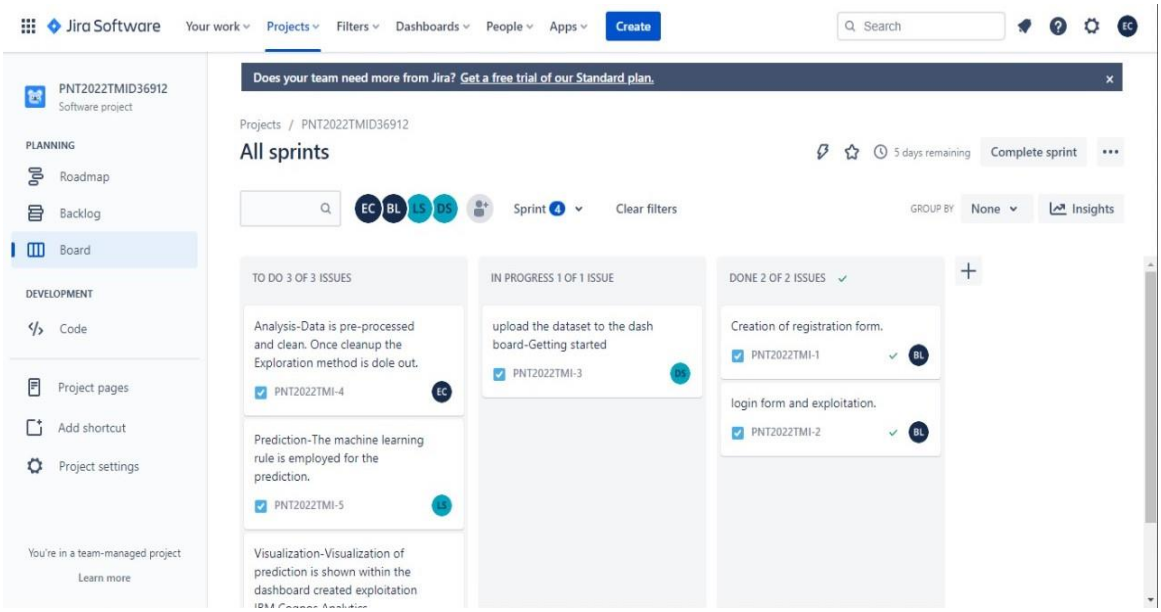
JIRA – SETUP OF SPRINTS:



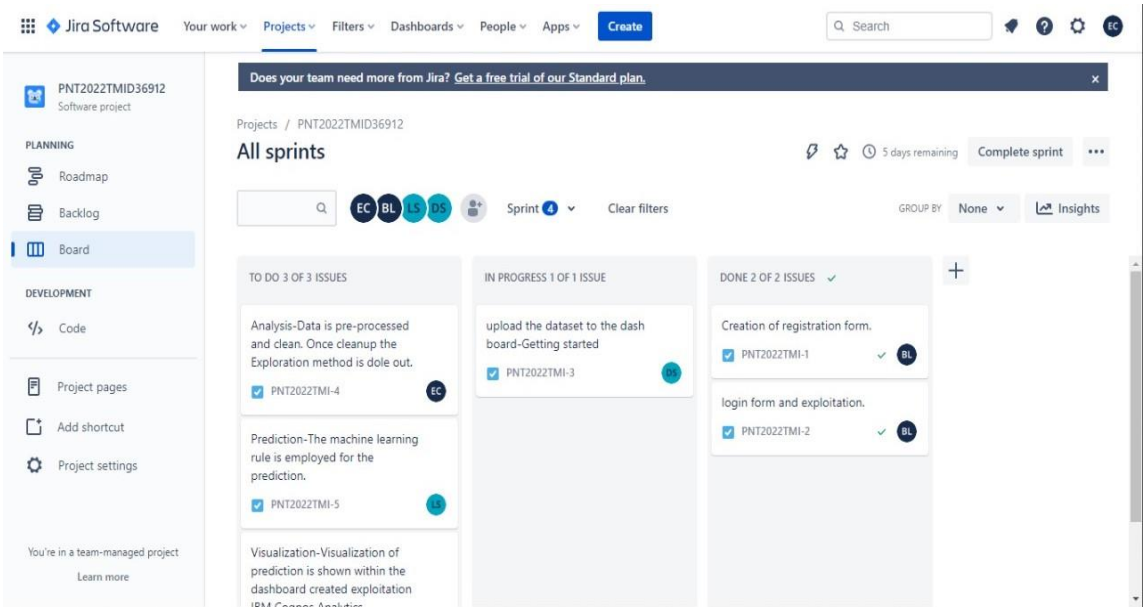


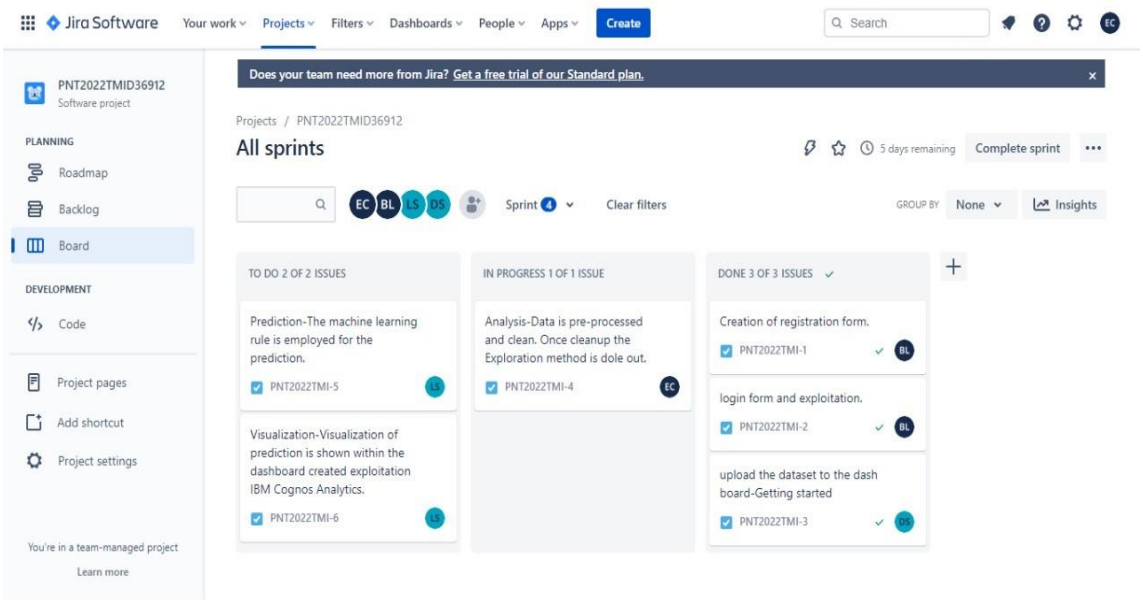
SPRINT-1 PROGRESS:



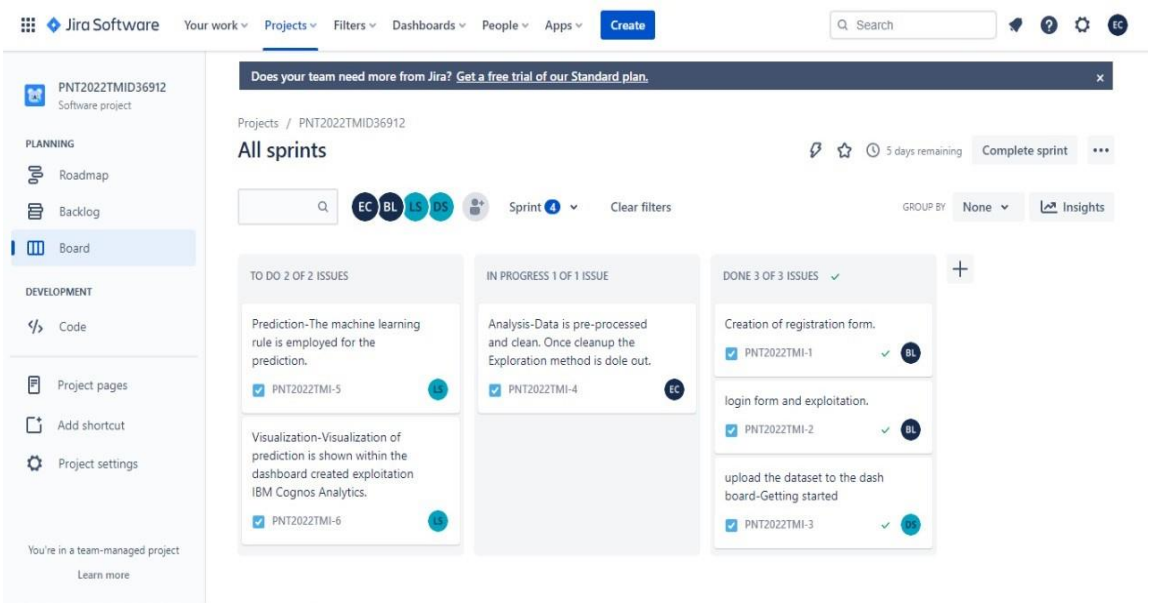


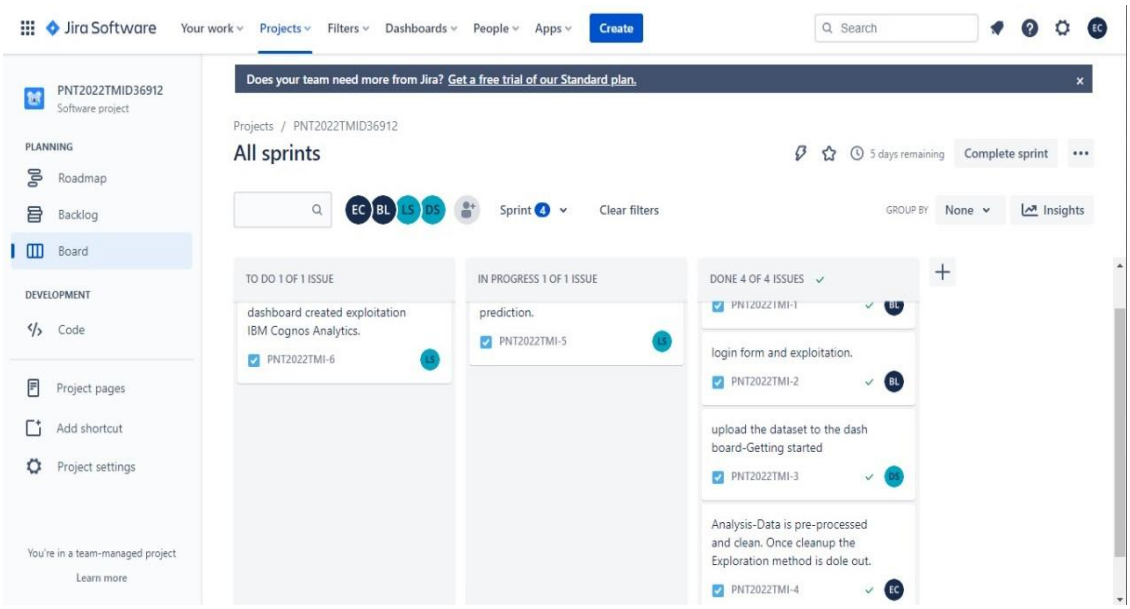
SPRINT-2 PROGRESS:



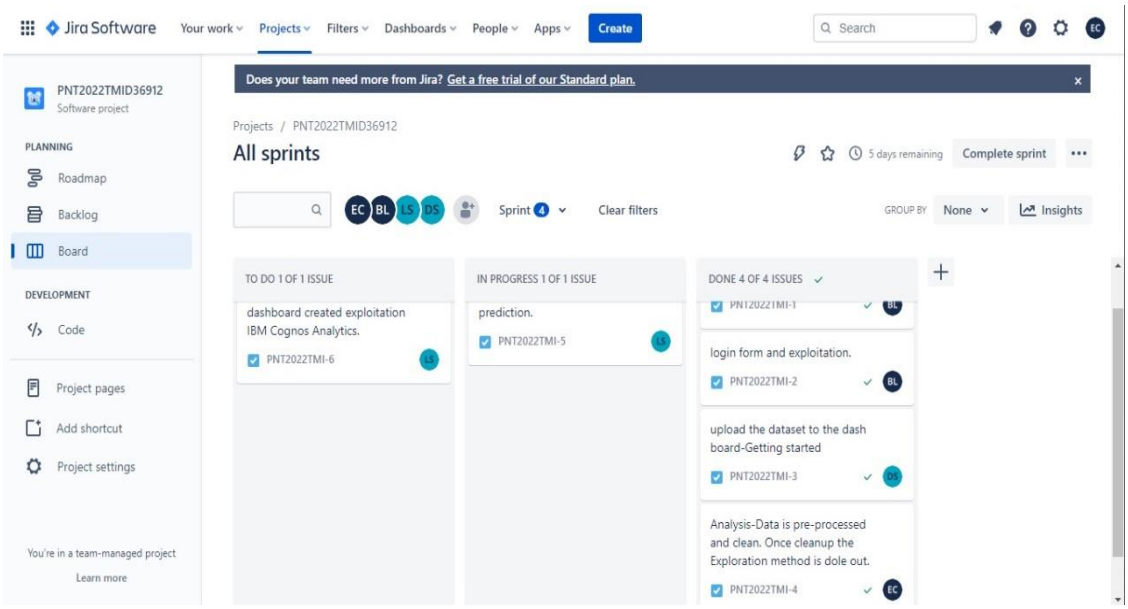


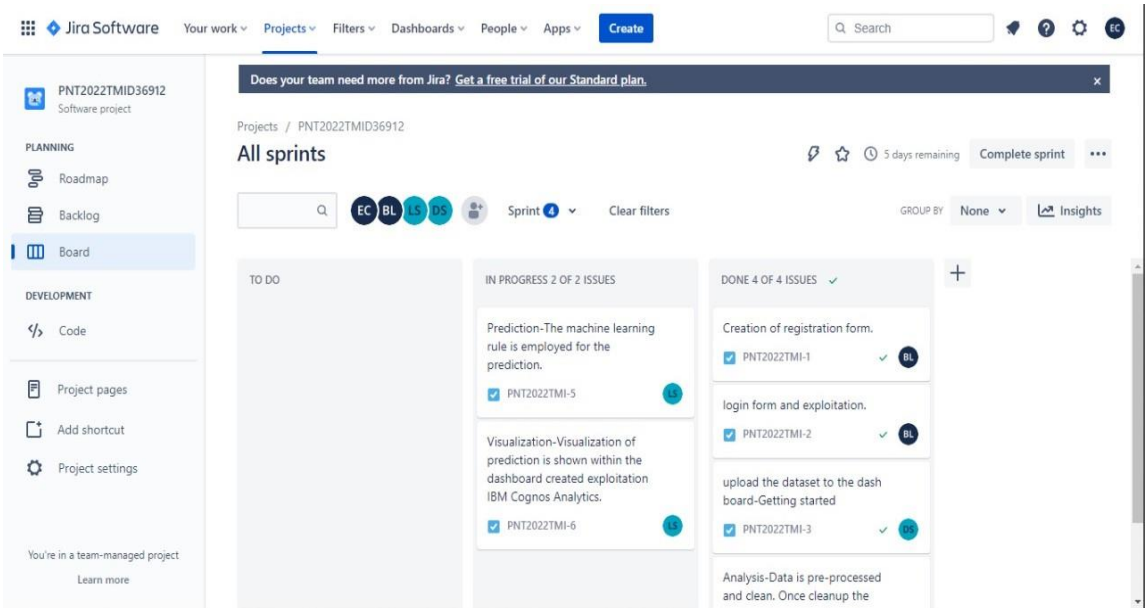
SPRINT-3 PROGRESS:

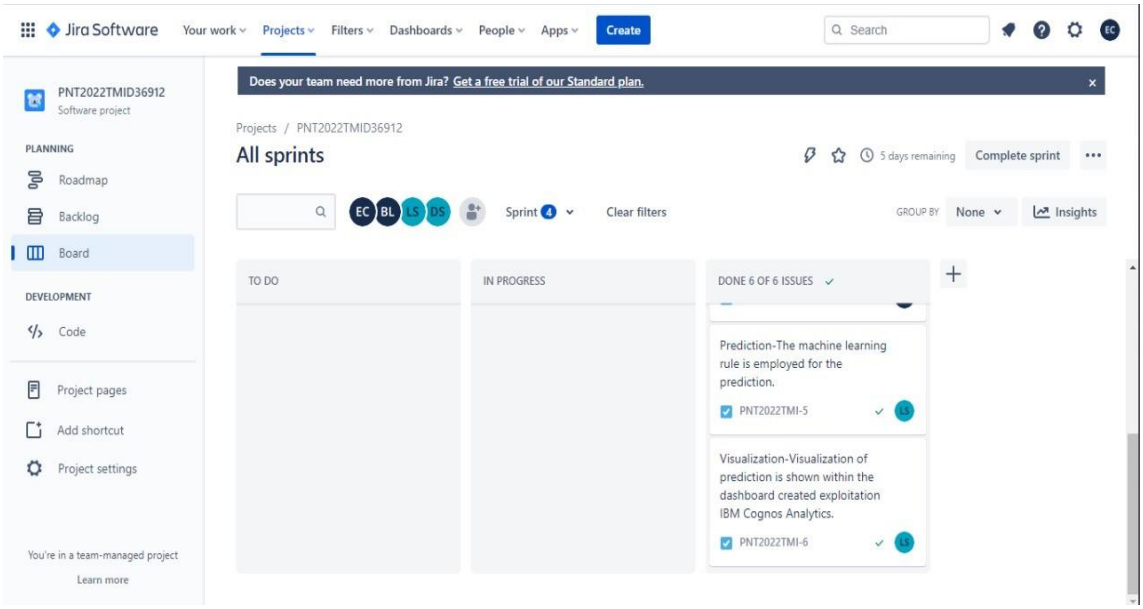




SPRINT-4 PROGRESS:




Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
PNT2022TMID36912
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / PNT2022TMID36912
All sprints
5 days remaining
Complete sprint
Sprint 4
Clear filters
GROUP BY
None
Insights
TO DO
IN PROGRESS 2 OF 2 ISSUES
Prediction-The machine learning rule is employed for the prediction.
PNT2022TMI-5
Visualization-Visualization of prediction is shown within the dashboard created exploitation IBM Cognos Analytics.
PNT2022TMI-6
DONE 4 OF 4 ISSUES
Creation of registration form.
PNT2022TMI-1
login form and exploitation.
PNT2022TMI-2
upload the dataset to the dash board-Getting started
PNT2022TMI-3
Analysis-Data is pre-processed and clean. Once cleanup the


Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
PNT2022TMID36912
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / PNT2022TMID36912
All sprints
5 days remaining
Complete sprint
Sprint 4
Clear filters
GROUP BY
None
Insights
TO DO
IN PROGRESS
DONE 6 OF 6 ISSUES
Prediction-The machine learning rule is employed for the prediction.
PNT2022TMI-5
Visualization-Visualization of prediction is shown within the dashboard created exploitation IBM Cognos Analytics.
PNT2022TMI-6

7. CODING & SOLUTIONING

7.1 Feature 1 :

SIGNUP.HTML:

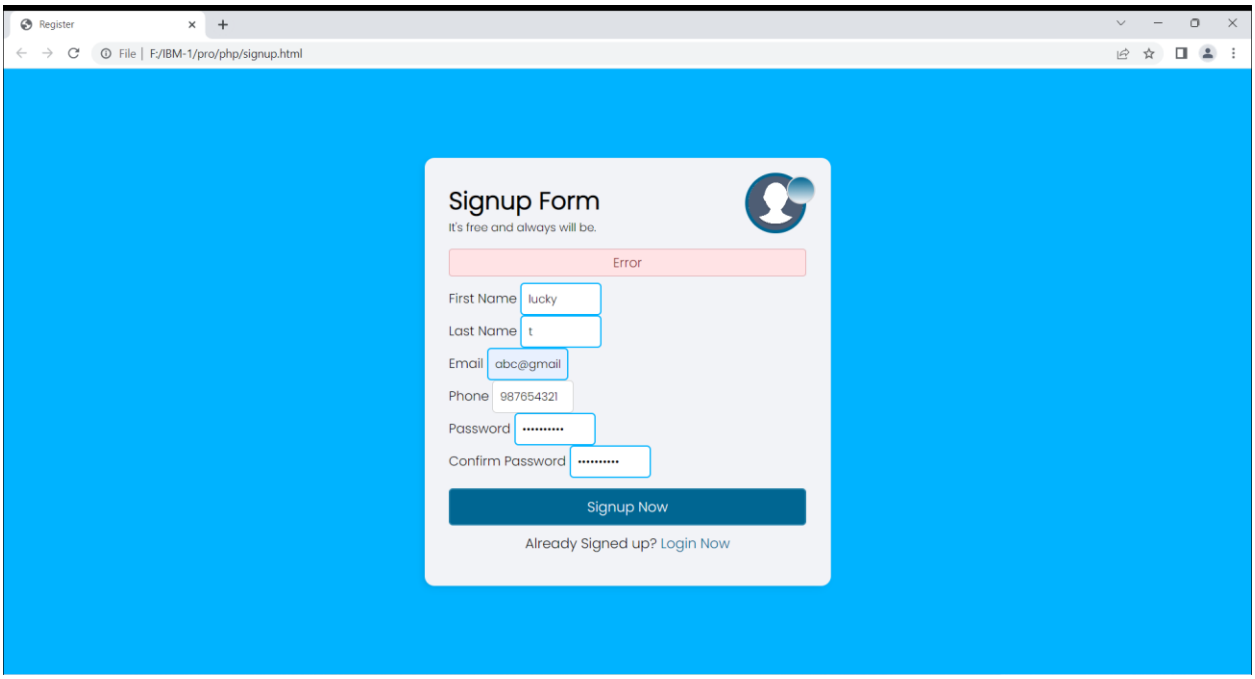
```
<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IEwedge">

<meta name="viewport" content="width=device-width, initial-scale=1.8">

<title>Register</title>

<link rel="stylesheet" href="F:\IBM-1\pro\php\form.css">


<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>

</head>

<body>

<div class="form">

<h2>Signup Form</h2>

<p>It's free and always will be.</p>

<form action="">

<div class="error-text">Error</div>

<div class="grid-detail">
```

```
<div class="input">
<label>First Name</label>
<input type="text" name="fname" placeholder="First Name" required pattern="[a-zA-
Z'-'\s]*">
</div>
<div class="input">
<label>Last Name</label>
<input type="text" name="lname" placeholder="Last Name" required pattern="[a-zA-
Z'-'\s]*">
</div>
</div>
<div class="input">
<label>Email</label>
<input type="email" name="email" placeholder="Enter your Email" required>
</div>
<div class="input">
<label>Phone</label>
<input type="phone" name="phone" placeholder="Phone Number" required
pattern="[0-9]{11}" oninvalid="this.setCustomValidity('Enter 11 Digits Number')"
oninput="this.setCustomValidity('')">
</div>
```

```
<div class="grid-detail">
<div class="input">
<label>Password</label>
<input type="password" name="pass" placeholder="Password" required>
</div>
<div class="input">
<label>Confirm Password</label>
<input type="password" name="cpass" placeholder="Confirm Password" required>
</div>
</div>
<div class="profile-img">
<div class="file-upload">
<input type="file" id="image-preview">
</div>
</div>
<div class="submit">
<input type="submit" value="Signup Now" class="button">
<i class="fas fa-user-edit"></i>
</div>
</form>
<div class="link">Already Signed up? <a href="">Login Now</a></div>
```

```
</div>

</body>

</html>
```

OUTPUT :



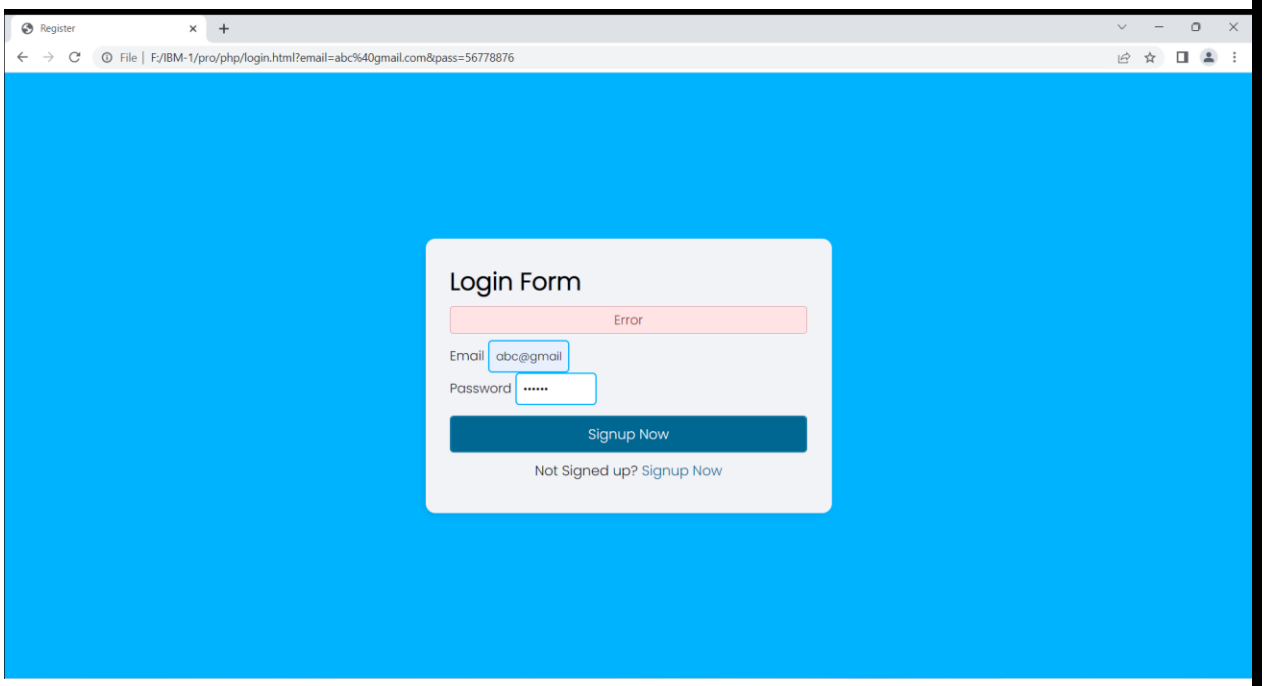
LOGIN.HTML:

```
<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">
```

```
<meta name="viewport" content="width=device-width, initial-
scale=1.0">
<title>Register</title>
<link rel="stylesheet" href="F:\IBM-1\pro\php\form.css">
</head>
<body>
<div class="form">
<h2>Login Form</h2>
<form action="">
<div class="error-text">Error</div>
<div class="input">
<label>Email</label>
<input type="email" name="email" placeholder="Enter your Email"
required>
</div>
<div class="input">
<label>Password</label>
<input type="password" name="pass" placeholder="Password"
required>
</div>
<div class="submit">
```

```
<input type="submit" value="Signup Now" class="button">

</div>

</form>

<div class="link">Not Signed up? <a href="">Signup Now</a></div>

</div>

</body>

</html>
```

OUTPUT :



FORM.CSS

```
@import

url("https://fonts.googleapis.com/css2?family=Poppins:wght@300&display=swap");
```

```
* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
    font-family: "Poppins", "sans-serif";
}
a{
  text-decoration: none;
}
body {
    position: relative;
    background: #00b3ff;
    display: flex;
    justify-content: center;
    align-items: center;
    min-height: 100vh;
}

::selection {
    color: #ececec;
    background: #00b3ff;
}
```

```
.form {
    position: relative;
    background: #f2f3f7;
    width: 100%;
    max-width: 500px;
    border-radius: 12px;
    box-shadow: -3px 3px 10px -5px rgba(0,0,0,0.2);
    padding: 30px 30px;
}

.form h2 {
    font-size: 30px;
    font-weight: 700;
}

.form p {
    font-size: 14px;
    padding-bottom: 8px;
}

.form form{
    margin: 8px 0;
}
```

```
.form form .profile-img{
    position: absolute;
    top: 0;
    right: 0;
    width: 75px;
    height: 75px;
    margin: 18px 30px;
    background: #f2f3f7;
    background-position: center;
    background-repeat: no-repeat;
    background-size: cover;
    background-image:
url("https://upload.wikimedia.org/wikipedia/commons/thumb/7/7e/Circle-icons-
profile.svg/768px-Circle-icons-profile.svg.png");
    border-radius: 50%;
    border: 4px solid #006692;
}
.file-upload {
    position: absolute;
    top: 0;
    right: -15px;
    height: 35px;
```

```
    width: 35px;
    display: flex;
    border-radius: 50%;
    border: 2px solid #d2d3d7;
    overflow: hidden;
    background: linear-gradient(to bottom, #006692, #f2f3f7 50%);
    background-size: 100% 200%;
    transition: all 1s;
    font-size: 14px;
    cursor: pointer;
}
form input[type="file"]{
    position: relative;
    height: 40px;
    width: 40px;
    opacity: 0;
    cursor: pointer;
    padding: 20px;
}

form i{
    position: absolute;
    top: 50%;
```

```
    left: 50%;
    transform: translate(-50%, -50%);
    font-size: 14px;
    color: #333;
    z-index: 100;
    pointer-events: none;
    cursor: pointer;
}
.file-upload:hover {
    background-position: 0 100%;
}
.file-upload:hover i {
    color: #2938aa;
}

.form form .error-text {
    color: #851923;
    padding: 4px 6px;
    text-align: center;
    border-radius: 4px;
    background: #ffe3e5;
    border: 1px solid #dfa5ab;
    margin-bottom: 8px;
```

```
}
.form .grid-details{
    display: flex;
}
.form .grid-details .input.first-child{
    margin-right: 8px;
    width: 100%;
}
.form .grid-details .input.last-child{
    margin-left: 8px;
    width: 100%;
}
.form form .input input{
    height: 40px;
    width: 100px;
    font-size: 14px;
    padding: 0 8px;
    border: 1px solid #ccc;
    border-radius: 5px;
    outline: none;
}
.form form .input input:focus,
```

```
.form form .input input:valid{
    border: 2px solid #00b3ff;
    transition: 0.1s ease;
    outline: none;
}
.form form input.button{
    height: 45px;
    border: none;
    color: #f2f3f7;
    width: 100%;
    background: #006692;
    font-size: 17px;
    border-radius: 5px;
    cursor: pointer;
    margin-top: 13px;
    border: 2px solid #2983aa;
}
.form .link{
    text-align: center;
    margin: 10px 0;
    font-size: 17px;
}
```

```
.form .link a{
color: #006692;
text-decoration: none;
transition: 0.5s all;
}
.form .link a:hover{
    text-decoration: underline;
}
```

7.2 Feature 2

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
import tensorflow as tf
```

```
data = pd.read_csv('../input/av-healthcare-analytics-ii/healthcare/train_data.csv')
```

```
data
```

OUTPUT :

	case_id	Hospital_code	Hospital_type_code	City_Code_Hospital	Hospital_region_code	Extra Rooms in Hospital	Department
0	1	8	c	3	Z	3	radiotherapy
1	2	2	c	5	Z	2	radiotherapy
2	3	10	e	1	X	2	anesthesia
3	4	26	b	2	Y	2	radiotherapy
4	5	26	b	2	Y	2	radiotherapy
...	...	...	...	...	...	...	...
318433	318434	6	a	6	X	3	radiotherapy
318434	318435	24	a	1	X	2	anesthesia
318435	318436	7	a	4	X	3	gynecology
318436	318437	11	b	2	Y	3	anesthesia
318437	318438	19	a	7	Y	5	gynecology

318438 rows × 18 columns

Preprocessing

Missing Values

```
data.isnull().sum()
```

OUTPUT :

```
case_id                    0
Hospital_code              0
Hospital_type_code         0
City_Code_Hospital         0
Hospital_region_code       0
```

```
Available Extra Rooms in Hospital      0
Department                             0
Ward_Type                              0
Ward_Facility_Code                     0
Bed Grade                            113
patient id                             0
City_Code_Patient                   4532
Type of Admission                      0
Severity of Illness                    0
Visitors with Patient                  0
Age                                    0
Admission_Deposit                      0
Stay                                   0
dtype: int64
```

```
def impute_missing_values(data, columns):
    for column in columns:
        data[column] = data[column].fillna(data[column].mean())
```

```
impute_columns = ['Bed Grade', 'City_Code_Patient']
```

```
impute_missing_values(data, impute_columns)
```

```
data.isnull().sum()
```

OUTPUT :

```
case_id                         0
Hospital_code                   0
Hospital_type_code              0
City_Code_Hospital              0
Hospital_region_code            0
Available Extra Rooms in Hospital    0
Department                      0
Ward_Type                       0
Ward_Facility_Code              0
Bed Grade                       0
patient id                      0
City_Code_Patient               0
Type of Admission               0
Severity of Illness             0
Visitors with Patient           0
Age                             0
Admission_Deposit               0
```

Stay 0

dtype: int64

Encoding

```
data.dtypes
```

OUTPUT :

case_id int64

Hospital_code int64

Hospital_type_code object

City_Code_Hospital int64

Hospital_region_code object

Available Extra Rooms in Hospital int64

Department object

Ward_Type object

Ward_Facility_Code object

Bed Grade float64

patient id int64

City_Code_Patient float64

Type of Admission object

Severity of Illness object

Visitors with Patient int64

```
Age                         object
Admission_Deposit           float64
Stay                        object
dtype: object
```

```
def get_categorical_uniques(data):
    categorical_columns = [column for column in data.dtypes.index if
data.dtypes[column] == 'object']
    categorical_uniques = {column: data[column].unique() for column in
categorical_columns}

    return categorical_uniques
```

```
get_categorical_uniques(data)
```

OUTPUT :

```
 {'Hospital_type_code': array(['c', 'e', 'b', 'a', 'f', 'd', 'g'], dtype=object),
 'Hospital_region_code': array(['Z', 'X', 'Y'], dtype=object),
 'Department': array(['radiotherapy', 'anesthesia', 'gynecology', 'TB & Chest disease',
        'surgery'], dtype=object),
 'Ward_Type': array(['R', 'S', 'Q', 'P', 'T', 'U'], dtype=object),
```

'Ward_Facility_Code': array(['F', 'E', 'D', 'B', 'A', 'C'], dtype=object),

'Type of Admission': array(['Emergency', 'Trauma', 'Urgent'], dtype=object),

'Severity of Illness': array(['Extreme', 'Moderate', 'Minor'], dtype=object),

'Age': array(['51-60', '71-80', '31-40', '41-50', '81-90', '61-70', '21-30',

'11-20', '0-10', '91-100'], dtype=object),

'Stay': array(['0-10', '41-50', '31-40', '11-20', '51-60', '21-30', '71-80',

'More than 100 Days', '81-90', '61-70', '91-100'], dtype=object)}

```
pd.get_dummies(data['Department'])
```

OUTPUT :

'''.

	TB & Chest disease	anesthesia	gynecology	radiotherapy	surgery
0	0	0	0	1	0
1	0	0	0	1	0
2	0	1	0	0	0
3	0	0	0	1	0
4	0	0	0	1	0
...	...	...	...	...	...
318433	0	0	0	1	0
318434	0	1	0	0	0
318435	0	0	1	0	0
318436	0	1	0	0	0
318437	0	0	1	0	0

318438 rows × 5 columns

```
def onehot_encode(data, columns):
    for column in columns:
        dummies = pd.get_dummies(data[column])
        data = pd.concat([data, dummies], axis=1)
        data.drop(column, axis=1, in place=True)
    return data
```

```
onehot_columns = ['Hospital_type_code', 'Hospital_region_code', 'Department',
'Ward_Type', 'Ward_Facility_Code']
```

```
data = onehot_encode(data, onehot_columns)
```

```
data
```

OUTPUT :

	case_id	Hospital_code	City_Code_Hospital	Available Extra Rooms in Hospital	Bed Grade	patientid	City_Code_Patient	Type of Admission	Sever of Illn
0	1	8	3	3	2.0	31397	7.0	Emergency	Extre
1	2	2	5	2	2.0	31397	7.0	Trauma	Extre
2	3	10	1	2	2.0	31397	7.0	Trauma	Extre
3	4	26	2	2	2.0	31397	7.0	Trauma	Extre
4	5	26	2	2	2.0	31397	7.0	Trauma	Extre
...	...	...	...	...	...	...	...	...	...
318433	318434	6	6	3	4.0	86499	23.0	Emergency	Mode
318434	318435	24	1	2	4.0	325	8.0	Urgent	Mode
318435	318436	7	4	3	4.0	125235	10.0	Emergency	Minor
318436	318437	11	2	3	3.0	91081	8.0	Trauma	Minor
318437	318438	19	7	5	2.0	21641	8.0	Emergency	Minor

318438 rows × 40 columns

```
categorical_uniques = get_categorical_uniques(data)

get_categorical_uniques(data)
```

OUTPUT :

```
{'Type of Admission': array(['Emergency', 'Trauma', 'Urgent'], dtype=object),
 'Severity of Illness': array(['Extreme', 'Moderate', 'Minor'], dtype=object),
 'Age': array(['51-60', '71-80', '31-40', '41-50', '81-90', '61-70', '21-30',
        '11-20', '0-10', '91-100'], dtype=object),
 'Stay': array(['0-10', '41-50', '31-40', '11-20', '51-60', '21-30', '71-80',
        'More than 100 Days', '81-90', '61-70', '91-100'], dtype=object)}
```

```
for column in categorical_uniques:
    categorical_uniques[column] = sorted(categorical_uniques[column])
```

```
categorical_uniques
```

OUTPUT :

{'Type of Admission': ['Emergency', 'Trauma', 'Urgent'],

'Severity of Illness': ['Extreme', 'Minor', 'Moderate'],

'Age': ['0-10',

'11-20',

'21-30',

'31-40',

'41-50',

'51-60',

'61-70',

'71-80',

'81-90',

'91-100'],

'Stay': ['0-10',

'11-20',

'21-30',

'31-40',

'41-50',

'51-60',

```
 '61-70',
 '71-80',
 '81-90',
 '91-100',
 'More than 100 Days']}

unique_list = categorical_uniques['Type of Admission']
unique_list.insert(0, unique_list.pop(unique_list.index('Urgent')))
unique_list.insert(0, unique_list.pop(unique_list.index('Trauma')))

unique_list = categorical_uniques['Severity of Illness']
unique_list.insert(0, unique_list.pop(unique_list.index('Moderate')))
unique_list.insert(0, unique_list.pop(unique_list.index('Minor')))

categorical_uniques
```

OUTPUT :

```
{'Type of Admission': ['Trauma', 'Urgent', 'Emergency'],
 'Severity of Illness': ['Minor', 'Moderate', 'Extreme'],
 'Age': ['0-10',
```

'11-20',

'21-30',

'31-40',

'41-50',

'51-60',

'61-70',

'71-80',

'81-90',

'91-100'],

'Stay': ['0-10',

'11-20',

'21-30',

'31-40',

'41-50',

'51-60',

'61-70',

'71-80',

'81-90',

'91-100',

'More than 100 Days']}

In [21]:

```
stay_mappings = {value: index for index, value in
enumerate(categorical_uniques['Stay'])}
stay_mappings
```

OUTPUT :

{'0-10': 0,

'11-20': 1,

'21-30': 2,

'31-40': 3,

'41-50': 4,

'51-60': 5,

'61-70': 6,

'71-80': 7,

'81-90': 8,

'91-100': 9,

'More than 100 Days': 10}

In [22]:

```
def ordinal_encode(data, uniques):
    for column in uniques:
        data[column] = data[column].apply(lambda x: uniques[column].index(x))
```

```
data['Stay']
```

OUTPUT :

```
0         0-10
1        41-50
2        31-40
3        41-50
4        41-50
          ...
318433    11-20
318434    31-40
318435    11-20
318436    11-20
318437     0-10
Name: Stay, Length: 318438, dtype: object
```

```
ordinal_encode(data, categorical_uniques)
```

```
data['Stay']
```

OUTPUT :

```
0       0
1       4
2       3
3       4
4       4
       ..
318433  1
318434  3
318435  1
318436  1
318437  0
Name: Stay, Length: 318438, dtype: int64
```

```
(data.dtypes == 'object').sum()
```

OUTPUT :

```
0
```

Scaling

```
data
```

OUTPUT :

Out[26]:

	case_id	Hospital_code	City_Code_Hospital	Available Extra Rooms in Hospital	Bed Grade	patientid	City_Code_Patient	Type of Admission	Severi of Illness
0	1	8	3	3	2.0	31397	7.0	2	2
1	2	2	5	2	2.0	31397	7.0	0	2
2	3	10	1	2	2.0	31397	7.0	0	2
3	4	26	2	2	2.0	31397	7.0	0	2
4	5	26	2	2	2.0	31397	7.0	0	2
...	...	...	...	...	...	...	...	...	...
318433	318434	6	6	3	4.0	86499	23.0	2	1
318434	318435	24	1	2	4.0	325	8.0	1	1
318435	318436	7	4	3	4.0	125235	10.0	2	0
318436	318437	11	2	3	3.0	91081	8.0	0	0
318437	318438	19	7	5	2.0	21641	8.0	2	0

318438 rows × 40 columns

```
data.set_index('case_id', in place=True)
```

```
y = data['Stay']

X = data.drop('Stay', axis=1)
```

```
scaler = StandardScaler()


X = pd.DataFrame(scaler.fit_transform(X), index=X.index, columns=X.columns)
```

```
X
```

OUTPUT :

ut[30]:

	Hospital_code	City_Code_Hospital	Available Extra Rooms in Hospital	Bed Grade	patientid	City_Code_Patient	Type of Admission	Severit Illness
case_id								
1	-1.195176	-0.571055	-0.169177	-0.716855	-0.904442	-0.053458	1.212557	1.6466
2	-1.890124	0.073580	-1.025217	-0.716855	-0.904442	-0.053458	-0.974973	1.6466
3	-0.963527	-1.215691	-1.025217	-0.716855	-0.904442	-0.053458	-0.974973	1.6466
4	0.889668	-0.893373	-1.025217	-0.716855	-0.904442	-0.053458	-0.974973	1.6466
5	0.889668	-0.893373	-1.025217	-0.716855	-0.904442	-0.053458	-0.974973	1.6466
...	...	...	...	...	...	...	...	...
318434	-1.426825	0.395897	-0.169177	1.574123	0.546379	3.342582	1.212557	0.1380
318435	0.658018	-1.215691	-1.025217	1.574123	-1.722559	0.158795	0.118792	0.1380
318436	-1.311001	-0.248738	-0.169177	1.574123	1.566288	0.583300	1.212557	-1.370
318437	-0.847702	-0.893373	-0.169177	0.428634	0.667022	0.158795	-0.974973	-1.370
318438	0.078895	0.718215	1.542903	-0.716855	-1.161314	0.158795	1.212557	-1.370

318438 rows × 38 columns

Training

```
X_train, X_test, y_train, y_test = train_test_split(X, y, train_size=0.8)
```

```
log_model = LogisticRegression()

log_model.fit(X_train, y_train)
```

OUTPUT :

/opt/conda/lib/python3.7/site-packages/learn/linear_model/_logistic.py:764:

ConvergenceWarning: lbfgs failed to converge (status=1):

```
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
```

Out[32]:

```
LogisticRegression()
```

```
inputs = tf.keras.Input(shape=(38,))
x = tf.keras.layers.Dense(16, activation='relu')(inputs)
x = tf.keras.layers.Dense(16, activation='relu')(x)
outputs = tf.keras.layers.Dense(11, activation='softmax')(x)

nn_model = tf.keras.Model(inputs=inputs, outputs=outputs)
```

```
nn_model.compile(
    optimizer='adam',
```

```
    loss='sparse_categorical_crossentropy',
    metrics=['accuracy']
)
```

```
batch_size = 32
epochs = 10

history = nn_model.fit(
    X_train,
    y_train,
    validation_split=0.2,
    batch_size=batch_size,
    epochs=epochs,
)
```

OUTPUT :

```
Epoch 1/10
6369/6369 [==============================] - 11s 2ms/step - loss: 1.6426 -
accuracy: 0.3769 - val_loss: 1.5901 - val_accuracy: 0.3925
Epoch 2/10
```

6369/6369 [==============================] - 10s 2ms/step - loss: 1.5766 - accuracy: 0.3995 - val_loss: 1.5674 - val_accuracy: 0.4011

Epoch 3/10

6369/6369 [==============================] - 10s 2ms/step - loss: 1.5618 - accuracy: 0.4030 - val_loss: 1.5591 - val_accuracy: 0.4023

Epoch 4/10

6369/6369 [==============================] - 10s 2ms/step - loss: 1.5560 - accuracy: 0.4047 - val_loss: 1.5560 - val_accuracy: 0.4012

Epoch 5/10

6369/6369 [==============================] - 10s 2ms/step - loss: 1.5520 - accuracy: 0.4064 - val_loss: 1.5485 - val_accuracy: 0.4076

Epoch 6/10

6369/6369 [==============================] - 11s 2ms/step - loss: 1.5486 - accuracy: 0.4078 - val_loss: 1.5490 - val_accuracy: 0.4030

Epoch 7/10

6369/6369 [==============================] - 10s 2ms/step - loss: 1.5457 - accuracy: 0.4088 - val_loss: 1.5440 - val_accuracy: 0.4100

Epoch 8/10

6369/6369 [==============================] - 10s 2ms/step - loss: 1.5440 -

accuracy: 0.4094 - val_loss: 1.5421 - val_accuracy: 0.4090

Epoch 9/10

6369/6369 [==============================] - 10s 2ms/step - loss: 1.5422 -

accuracy: 0.4101 - val_loss: 1.5419 - val_accuracy: 0.4089

Epoch 10/10

6369/6369 [==============================] - 10s 2ms/step - loss: 1.5406 -

accuracy: 0.4105 - val_loss: 1.5408 - val_accuracy: 0.4100

In [36]:

```
print(f"Logistic Regression Acc: {log_model.score(X_test, y_test)}")
print(f"    Neural Network Acc: {nn_model.evaluate(X_test, y_test, verbose=0)[1]}")
```

OUTPUT :

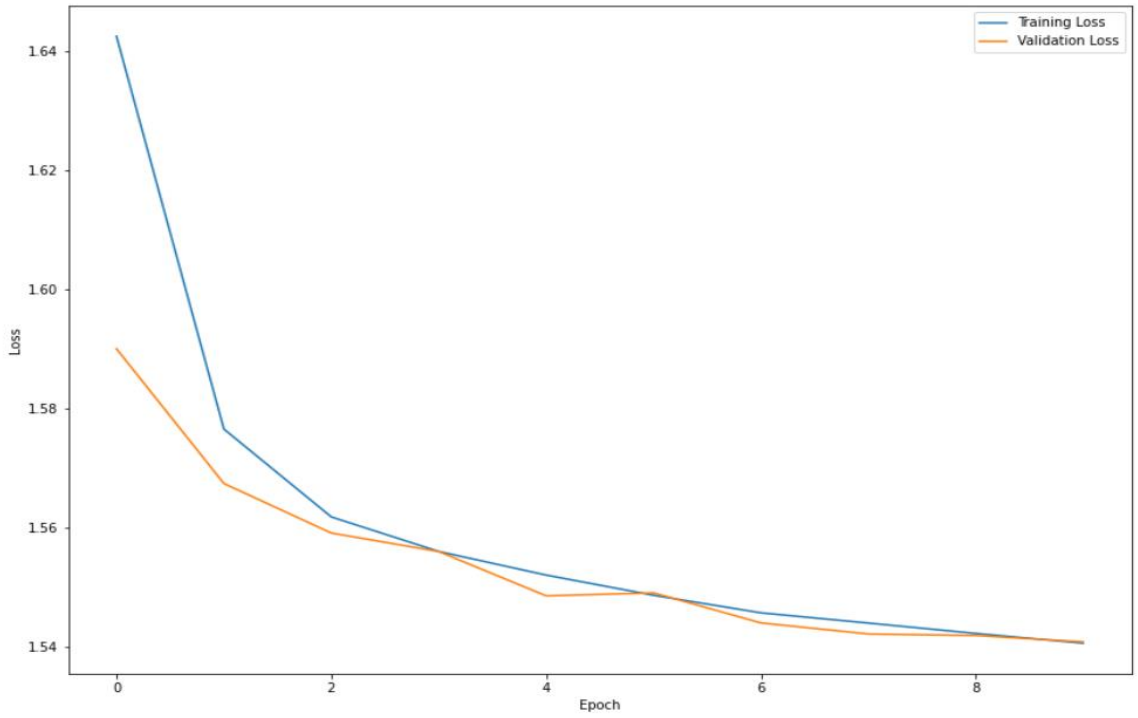
Logistic Regression Acc: 0.39104697902273583

    Neural Network Acc: 0.4083343744277954

```
plt.figure(figsize=(14, 10))

plt.plot(range(epochs), history.history['loss'], label="Training Loss")
```

```
plt.plot(range(epochs), history.history['val_loss'], label="Validation Loss")


plt.xlabel("Epoch")

plt.ylabel("Loss")

plt.legend(loc='upper right')

plt.show()
```

OUTPUT :




```
np.argmin(history.history['val_loss']) + 1
```

OUTPUT :

10

8. TESTING

8.1 Test Cases:

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	BUG ID	Executed By
LoginPage_TC_OO1	Functional	Home Page	Verify user is able to go to the signup page on clicking signup button	Register page URL	1.Enter URL 2.Click on signup now link 3.Verify Singup page is loaded or not	Signup page should be loaded	Working as expected	Pass	Success	Y	Null	Enitha
LoginPage_TC_OO2	UI	Home Page	Verify the UI elements in Login	Hompage URL	1.Enter URL and click go 2.Click on Login button/Signup now link 3.Verify login/Singup with below UI elements: a.email text box b.password text box c.Login button d.New customer? signup now link	Application should show below UI elements: a.email text box b.password text box c.Login button with green colour d.New customer? Signup now link	Working as expected	Pass	Success	Y	Null	Logeshwari
LoginPage_TC_OO3	Functional	Home page	Verify user is able to log into application with Valid credentials	Login page URL	1.Enter login page URL and proceed 2.Enter Valid email in Email text box 3.Enter valid password in password text box 4.Click on login button	User should navigate to user account homepage	Working as expected	Pass	Success	Y	Null	Divyadharshini
LoginPage_TC_OO4	Functional	Login page	Verify user is not able to log into application with InValid credentials	Login page URL	1.Enter login page URL and proceed 2.Enter Valid email in Email text box 3.Enter valid password in password text box 4.Click on login button	User should not be able to Login with valid inputs	Working as expected	Pass	Success	Y	Null	Bhagyalakshmi

8.2 User Acceptance Testing:

1. Purpose of Document:

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

2.Defect Analysis :

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

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	7	3	2	0	12
Duplicate	1	0	0	0	1
External	2	1	0	1	4
Fixed	10	2	3	20	35
Not Reproduced	0	1	0	0	1
Skipped	0	0	0	0	0
Won't Fix	0	0	0	0	0
Totals	20	7	5	21	53

9. RESULTS

9.1 Performance Metrics

S.No.	Parameter	Screenshot / Values
1.	Dashboard design	5
2.	Data Responsiveness	The final deliverables, outputs from IBM Cognos with Watson are converted to pdf format so that they would be available to be viewed from all kinds of devices and platforms.
3.	Amount Data to Rendered (DB2 Metrics)	26285 kb
4.	Utilization of Data Filters	Data filters like ward types, case ID, Hospital code, Severity, Department, etc.,
5.	Effective User Story	No of Scene Added - 6
6.	Descriptive Reports	5

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES:

• Helps an organization to make a better decision

• Increase the efficiency of the work

• The analytics keeps you updated of your customer behavioral changes.

• Personalization of hospital details.

• Improving quality of service and health care.

DISADVANTAGES:

• Lack of alignment within teams

• Lack of commitment and patience

• Low quality of data

• Privacy Concerns

• Complexity and Bias

11. CONCLUSION

Data analytics in health care is vital. It helps health care organizations to evaluate and develop Number of patients by ward, Age wise patients with department details, Various types of visualizations to analyze the hospital's datasets and hence predict outbreaks in illness, Data analytics can also lower costs for health care organizations and boost business intelligence.

This project shows that unstructured data can be used to predict LOS with acceptable predictive performance. The performance was similar to the performance of the model using structured data. Structured data, however, may have the drawback of being more time-consuming to extract. In many applications, unstructured text data contains valuable insights that are yet to be explored. As the methods to automatically extract knowledge evolve, they will undoubtedly give more accurate predictions. Modules to extract specific information like the primary complaint or presence of pain are currently being developed and could be combined or added to already existing software

12. FUTURE SCOPE

While every fact of the industry stands to be changed by data analytics in healthcare, data has significantly improved healthcare in three areas: conducting medical studies, understanding the cost of medical tests and health insurance, and making preventative recommendations to patients.

Hospital Healthcare data analytics helps in analyzing the patient details via hospital that the availability of doctors and number of beds to the patients and hence, it reduce the man power and time of the respective Hospital

13. APPENDIX

GitHub Link

https://github.com/IBM-EPBL/IBM-Project-54941-1663220111

Project Demo Link

https://youtu.be/h1SLt6Vruol