Process Incident and Defect Visualizer

A Thesis Submitted for the Degree of Master of Business Analytics

R. M. Isaac

University of Colombo School of Computing

2021

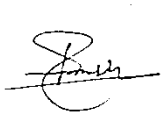
DECLARATION

I hereby declare that the thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis. This thesis has also not been submitted for any degree in any university previously.

Student Name: R.M. Isaac

Registration Number: 2018/BA/016

Index Number: 18880161



Rumesh Isaac

September 22, 2021

_____

Signature of the Student & Date

This is to certify that this thesis is based on the work of Mr. R.M.Isaac under my supervision. The thesis has been prepared according to the format stipulated and is of acceptable standard.

Certified by,

Supervisor Name:

_____

Signature of the Supervisor & Date

I would like to dedicate this thesis to all the students who struggled to complete thesis for their Master's degree in the midst of Covid-19 pandemic at University of Colombo School of Computing, Sri Lanka.

ACKNOWLEDGEMENTS

I would first like to thank my advisor and supervisor Dr. Dinuni K. Fernando for the wonderful guidance I had at UCSC. I learned the lot of art of communicating from the beginning of my research proposal to final dissertation with evaluators under your supervision. I have always appreciated the deep respect you have on your students. I probably made a lot of mistakes, but you never discouraged me. I am grateful to have supervised under you.

I really thank UCSC Management & lecturers for the given chance to start Master's studies and numerous helps and encouragements.

I thank my wonderful company CodeGen International (pvt) Ltd for granting me leaves and bearing me in this difficult time also.

I especially thank Mr. G.A.S.M Padmasiri (BA/2018/024) for encouraging me to continue the Master’s studies. I really admire your kindness and guidance.

I also thank my numerous bible study friends for making me grow spiritually in God and teaching me what is really important in life. I especially thank my Jesus Youth Fellowship and Delatura, Ja-Ela Church friends.

I thank my parents who have sacrificed so much for my education. Without them, I would not have been able to study at UCSC.

I thank my wife who have sacrificed so much time and energy for me and take care of me. I especially thank you for being such a kind and encouraging wife.

I love you all from the bottom of my heart.

ABSTRACT

Each and every one prefers to know and understand what ahead in our paths. For software industry this is essential. Being able to predict the customer satisfaction with maximizing efficiency and productivity would offer the ability to build reliable software. The tremendous amount of time and costs would be saved across the effectively manage human and material resources delivering the service that client expect of them.

The presence of temporal trends in incidents tends to show patterns in data with certain period of time. The incidents demonstrate certain issue classifications such as defect, explanation, data setup issue, data correction, connectivity issue, enhancement and invalid. These certain patterns in relation to these set periods, spend time, and priority, the output can be used to predict future trends.

This study discusses various ranges of problem of incident processes exist in the real world. Also discusses challenges when gathering such data. Then we process these incidents and visualize them using Google Data Studio “Data Studio is a free tool that turns your data into informative, easy to read, easy to share, and fully customizable dashboards and reports” (Welcome to Data Studio! - Data Studio Help, 2021). Finally we incorporate machine learning based approaches to predict customer satisfaction / dissatisfaction using processed incidents.

Table of Contents

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

The customer satisfaction is the goal of the business, and then constant feedback is the smartest way to improve over time to eliminate the customer dissatisfaction. The first attempt is to bring the areas and reasons of customer dissatisfactions.

The delivery gap specifies between customer requirements or expectations and delivered service quality specification. The incidents report from client and can arise at any time. The incident categorization process runs in very high-level manner according to the business priority. All the incidents cannot be defects. Different list of incident categories arise from client which do not have categorized in the beginning.

Reported incidents can be categorized based on the incident type as below.

- Defect
- Explanation
- Data Setup Issue
- Data Correction
- Connectivity Issue
- Enhancement
- Invalid

What are the concerns of reporting all incidents as defects?

If all the incidents report as defects;

- All the reported incidents would be prioritized.

Visualize high volume of incidents make chaos on emergent requests of responses from software delivery team.

- The wrong interpretation visualizes on top management.

Incident	High Priority	Medium Priority	Low Priority	No of Incident
Defect	4	3	6	13

Table 1.1: Incidents Reported as Defects

As an example, assume that the client has reported several incidents and it represents a negative impact on quality of software delivery team. Refer the 'Table 2.1: Incidents Reported as Defects'.
Initially all the reported incidents are considered as defects and according to the Service Level Agreement (SLA), only the customer reported issues will be set as high priorities.

According to the above 'Table 3.1: Incidents Reported as Defects'; Software delivery team is responsible for thirteen (13) incidents and all the thirteen reported incidents initially identified as defects.

Incident Category	High Priority	Medium Priority	Low Priority	No of Incident
Defect	3	1	2	6
Explanation	0	0	3	3
Enhancement	0	2	1	3
Data Setup Issue	1	0	0	1
Total	4	3	6	13

Table 4.2: Issue Categories

Demonstrate of incidents mitigate the negative impact on a certain extent.Observing above 'Table 5.2: Issue Categories'; Incidents visualization becomes next level view. Incidents do not belong to the one specific team or a person.

And also actual defect count had been reducedcomparatively.

- The poor interpretation buildup on quality standard levels.
- Absence of actual defects prioritization.

What are the concerns of incident categorization?

- Lack of accuracy on incident categorization.
- Incident categorization is a time consuming manual task.

Why incident categorization doesn't solve customer dissatisfaction problem?

- The incident categorization reveals only the type of the each incident (whether the incident is a Defect, Explanation, Data Setup Issue, Data Correction, Connectivity Issue, Enhancement or Invalid).
- The incident categorization is only a high-level visualization of the reported incidents.

"When customers share their story, they're not just sharing pain points. They're actually teaching you how to make your product, service, and business better. Your customer service organization should be designed to effectively communicate those issues."– Kristin Smaby, "Being Human is Good Business" (Khanka, 2005).

Why customer dissatisfaction levels are needed to be measured?

- Identification of customer dissatisfaction can make the positive impact on future dissatisfaction.
- Identification and comparison of the levels of customer dissatisfaction describes that the exact level of dissatisfaction.
- Modeling the algorithm to predict the customer dissatisfaction levels.

Knowing the customers are dissatisfied; the businesses are not capable to measure the customer dissatisfaction from data provided as the customer dissatisfaction is a quantitative factor. The project aims to identify customer satisfaction and dissatisfaction considering statistical and machine learning approaches.

Motivation

While the businesses might survive with angry, single-purchase customers, only the businesses with a focus on customer satisfaction will thrive in this competitive world. Customer satisfaction is the difference between surviving and thriving. And also customer satisfaction is the key in creating a long-term relationship with customers and the key of any successful business.

Nowadays, keeping a long term relationship functioning is becoming a hard work. Businesses are trying to gain new opportunities while the current businesses are operating. The hard part comes with there. When the business focuses on the new opportunities, planning marketing strategies the current customers might be neglected and due to that reason, the existing valuable customers would be losing or dissatisfied. Customer dissatisfaction is not a good sign for any of the businesses.

Customer dissatisfaction effects in many ways to the businesses. Due to the reason of customer dissatisfaction, the business starts losing the business opportunities and creates unhappy employees as well. Unhappy employees never take care of their customers and even happy valuable customers might be unhappy cause of unhappy employees. So the customer relationship management (CRM) is becoming a necessary and crucial factor. Maintaining customer satisfaction and dissatisfaction will be the major deliveries of the CRM.

If the customer satisfaction is the goal of the business, then constant feedback is the best way to improve over time to eliminate the customer dissatisfaction. To eliminate the customer dissatisfaction, the first task is to identify the areas and reasons of customer dissatisfactions. Once we identified the dissatisfaction areas and reasons of the customers, it will be given an opportunity to find and make correction the exact root cause of the dissatisfaction.

The software industry believes that the key factor of the business success is delivering the high quality products and services to their clients before their competitors. Any of the very small delay will be a critical impact of the many years spent for research investments. So the companies are following all the

best practices in SDLC and defect management system processes throughout the product delivery beforehand to their customers. The defect management system comprise from both aspects as internal and external. Internal aspect is basically focusing to give their product with the best quality. Internal product development teams are logging queries and concerns internally and giving solutions to response and resolve each queries and concerns. Purpose of this internal tracking is delivering quality products and services to their customers.

External aspect is how the client feedback of the given products and services in the live business flows. This is the most critical part of the defect management systems. Client is the one who is actually using the given products and services in a live business environment. While using the live business flow they identifying the major impact areas of the given products and services and they are recording it in a defect management system and assigned each and every issue to the product company. And also the business is now on live business phase and they have high requirement of immediate solution from the product development company to maintain their business scenarios without any flaws. Business is focusing and providing solutions according to the Business priority and system priority of each and every logged issue.

Businesses are practicing the same pattern when the issue arises from the client end; they are taking the issue and provide the fix. But with the timeline, as software Product Company needs to focus on which area is week and which resources are giving poor outcome.

What are the reasons of these outcomes and what kind of decisions should be taken from the process and technology perspectives? Because, we are living in a high competitive business world and customer expectation is always high. Both human resource and product development is essential for face the competition and thrive in the market.

This is the problem which have found in the most companies today. Though they know their customers are dissatisfied, the businesses are not capable to measure the customer dissatisfaction as the customer dissatisfaction is a qualitative factor.

This business analytics project will give the solution for identify the client satisfaction and dissatisfaction. We will measure the dissatisfaction levels of customer using statistical methodology.

Statement of the problem

Businesses expect financial stability to survive and to make customers happy. Satisfied customers are willingly spread the goodwill of the business always. It helps to make more business opportunities and bring new customers for the customer base. Only the businesses with a focus on customer satisfaction will thrive in the competitive business world. Customer satisfaction is the difference between surviving and thriving. Nowadays, keeping a long-term relationship is becoming a hard work. Businesses are trying to gain new opportunities while the current businesses are operating. If the customer satisfaction

is the goal of the business, then constant feedback is the best way to improve over time to eliminate the customer dissatisfaction. To achieve this, the first task is to identify the areas and reasons of customer dissatisfactions. When we identified the issue areas and reasons, it will be given an opportunity to find and do the correction for the exact root cause of the dissatisfaction.

Research Aims and Objectives

Aims

- Build the quality standard for incident interpretation.
- Reduce the false prioritization of incidents.
- Increase the accuracy of incident categorization and classification.
- Effectively investigation of root causes of the incidents.
- Visualize the ROI on time is a valuable task in incident categorization and classification.
- Show that the client satisfaction prediction is not an easy task.
- Presents that keep client satisfaction is a continuous effort.

Objectives

- Customer inquiries data set drill down into categorizations & classifications.
- Identify the levels of customer dissatisfaction.
- Predict the customer dissatisfaction.
- Introduce the visualization of dashboard to represent the levels of dissatisfaction.
- Identify the key areas to improve the technologies and processes in the business.
- Identify the major skill levels of human resources to decrease customer dissatisfaction.
- Generate the report of the customer inquiries with deep analysis of categorization and classification.
- Make easy of the decision taking using statistical report.
- Visualize defect management process with the time that team need to fix and giving statistical analysis on efficiency.

Scope

Visualize a dashboard by considering software incidents in different categories and classifications.

Below shows the scope of the dashboard; in “Figure 1.1: Incident Handling Process”.

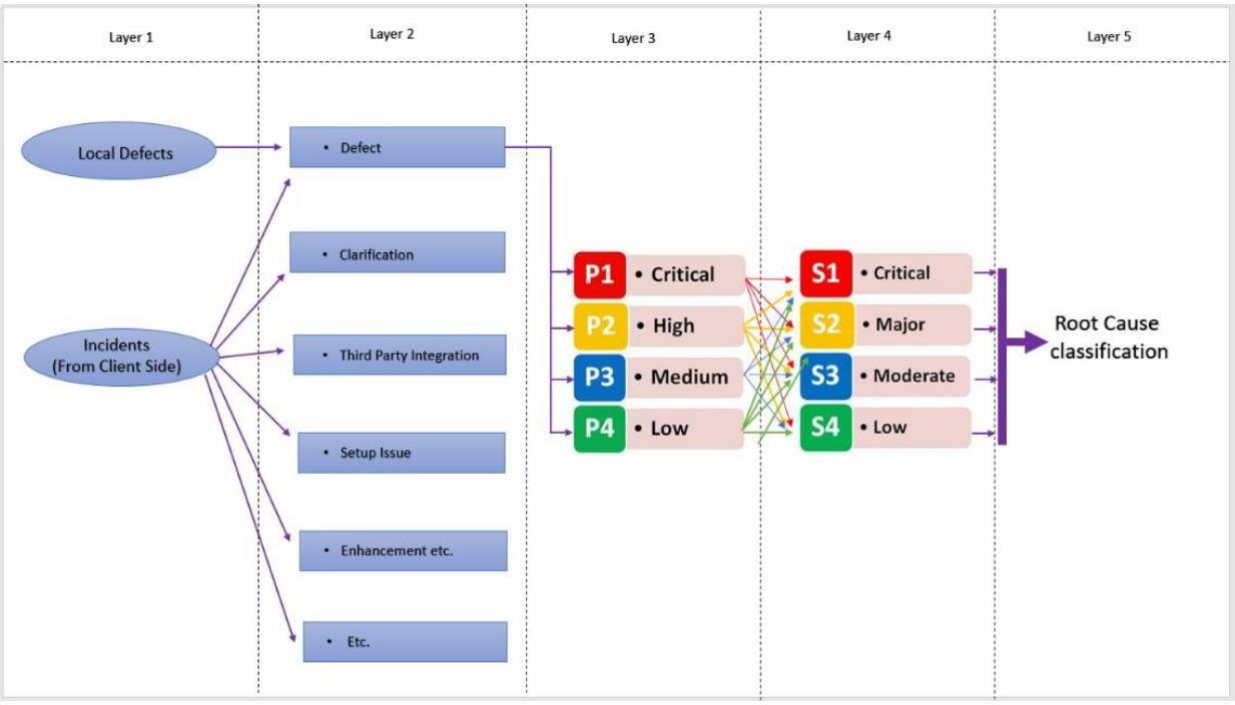


Figure 2.1: Incident Handling Process

Layer 1: It’s handling of incidents come from client side and local defects report from project level.

Layer 2: Its categorizing incidents/Local defects into incident category.

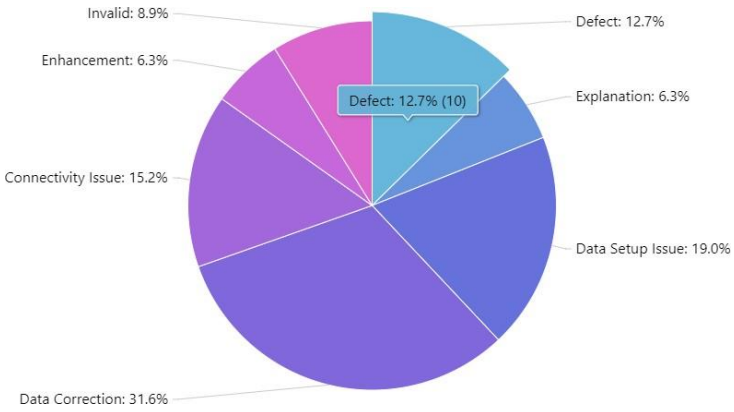


Figure 1.3: Incident Categories

Layer 3: In this layer, set the business priority for the defects by considering client requirement. Refer “Figure 1.4: Incident Categories”.

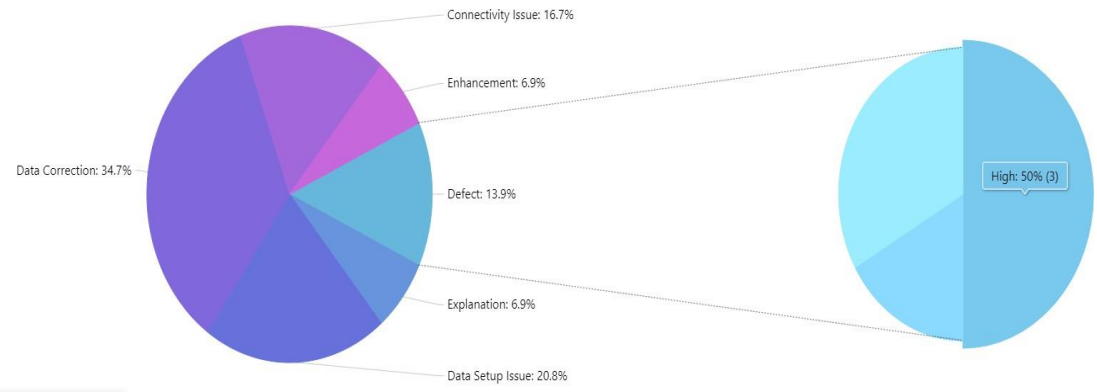


Figure 1.3: Set the business Priority

Layer 4: in this layer, team is involving with set the severity by considering issue impact on the system. Please refer "Figure 1.3: Set the business Priority".

Layer 5: Then defects will be classifying into root causes.

Defect Classification	High Priority	Medium Priority	Low Priority	No of Defects
Incorrect Requirement	1	0	1	2
Omitted Requirement	0	1	0	1
Misunderstood Requirement	1	0	1	2
Incorrect Design	0	1	0	1
Misunderstood Design	1	0	0	1
Coding Error	2	0	1	3
Total	5	2	3	10

Table 1-4: Defect Classification

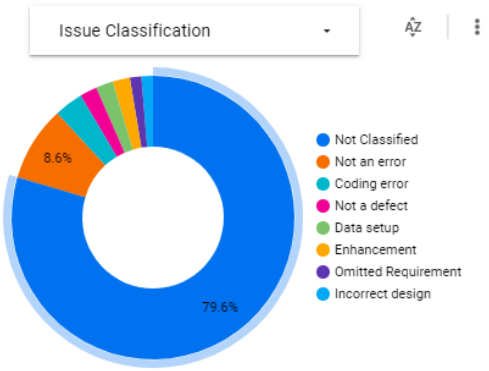


Figure 1.4: Issue Classification

Then proposed solution will be analyzing to visualize all the requirements, that project team needs to make the decision by considering customer Satisfaction/Dissatisfaction levels and improvements within the project team in next sub layers of the parent layers. Please refer the "1.4: Issue Classification".

Proposed solution will be covering following aspects.

Understanding the business issues and the data set

- Business objectives
- Information needed
- Type of analysis
- Scope of work
- Deliverables
- Initial data collection
- Data requirements
- Data availability
- Data exploration and characteristics

Prepare the data Set

- Gather data from multiple sources
- Cleanse
- Format
- Blend
- Sample

Perform exploratory analysis and modeling

- Develop methodology
- Determine important variables
- Build model
- Assess model

Validate the data set

- Evaluate results
- Review process
- Determine next steps
- Reevaluate results

Visualize and present the findings

- Communicate results

- Determine best method/graph to present insights based on analysis and audience
- Craft a compelling story
- Make recommendations

Limitations:

- This project will not predict the category or classification of customer queries. The categorization and classification will be setup manually.
- Customer Satisfaction/Dissatisfaction will not predict using emotional comment.

CHAPTER 2: LITERATURE REVIEW

2.1. Literature Review

This project can breakdown two major phases. First phase is research part and second phase is building a model and machine learning algorithm.

Understanding the business and preparing data set is belonged to the research part of this project. In this phase we are focusing on few points. Throughout this process we are collecting the knowledge of the business and business problems. Accordingly we are collecting data. So in this phase we are having thorough understanding of the business problem and characteristics of data. Please refer the "Figure 2.1: Research & Solution Flow".

Second part of this project is performing exploratory analysis and modeling with data validation and visualizing. It is helping to understand the problem we are already investigating for. Model presents a simple version of the decided solution.

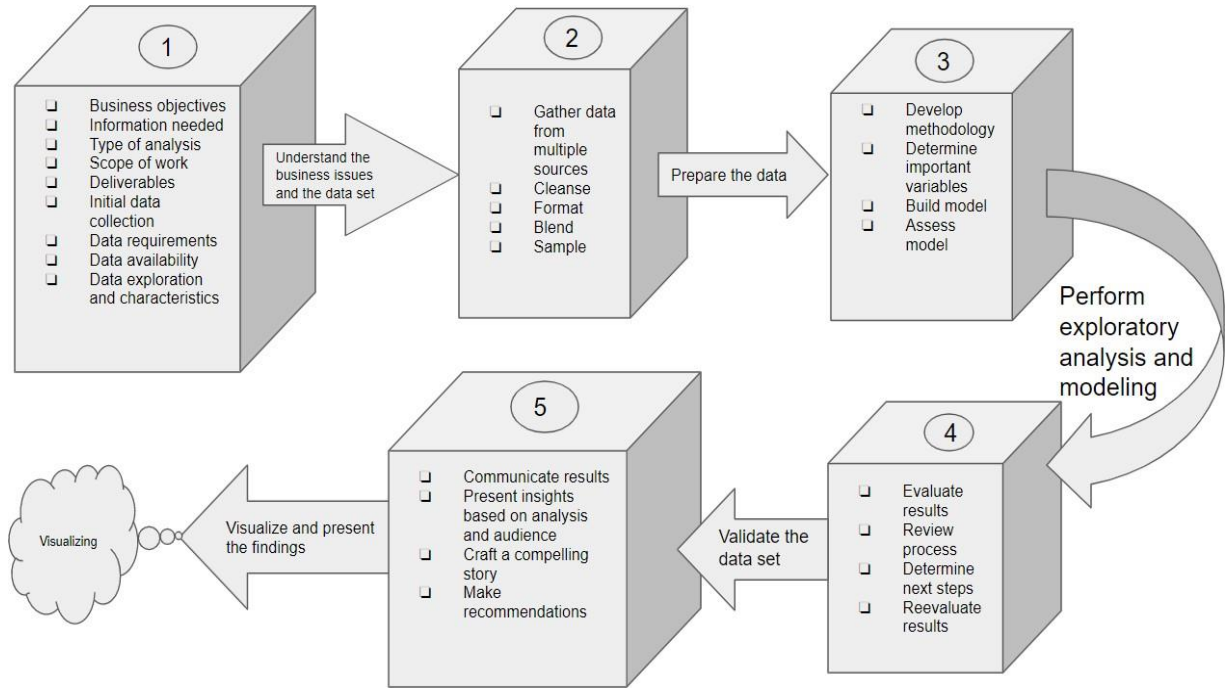


Figure 2.1: Research & Solution Flow

The first two sections (1st and 2nd) have been completed under the phase 1 research part. The following three sections (3rd, 4th and 5th) are under development of phase 2.

Understanding the business issues and the data set

We mainly focused on the maintenance phase of the Software Development Company. The client

issues are raising in the maintenance phase after delivered the product to the client. The important information we needed was the customer complaints. In software industry it's called incidents. "While executing atest, you might observe that the actual results vary from expected results. When the actual result is different from the expected result then it is called as incidents, bugs, defects, problems or issues" (Understanding Confusion Matrix - Towards Data Science, 2021). One of the main types of analysis in here is prediction of the customer satisfaction and dissatisfaction. To achieve this analysis first we had to label the incidents accordingly. Initial data collection is processing on Apache organizations'(open source project and open data set) incidents which were reported by the users. All the Resolved- Fixed | Verified-Fixed | Closed-Fixed incidents were taken for the root cause analysis. All the incidentsconsider for predictions under the customer satisfaction and dissatisfaction.

Prepare the data

In this phase conducted the process of data preparation focusing on following points.

Gather data from source

Data gathered basically from five software projects in the Apache organization. Apache organization using defects managed using Jira Project Management System. What is Jira; "Jira is a proprietary issue tracking product developed by Atlassian that allows bug tracking and agile project management" (Jira (software) - Wikipedia, 2021). With the availability of data, few projects selected as follows; Mesos, UIMA, Beam, Apache Cordova, and Atlas.

When collecting data, there were few observations found.

1. Spent time had not log.
2. Issue classification didn't
   These two factors are crucial for data set when analysis.

Cleanse and Transformation

Data had to clean for take insight of it. There were garbage data and had to remove, fix and correct the corrupted, incorrectly formatted, duplicated and incomplete data within a data set.

Mainly took following methods when cleansing data;

1. Removed data that did not belong to data set.
2. Transformed data from one format to another.

Blend

Data blending is processed combining data from multiple sources into a functioning data set. This process gained straightforward method used to extract value from multiple data sources.

Sample

When identifying data sample from the data pool, incident created date used to extract data sample. The sample selected between 2015-2021 years. Please refer the “Table 2.1:Data Pool” for further information.

Field	Type	Default Aggregation	Description
DIMENSIONS (12)			
Client Satisfaction	ABC Text	None	
Created	Date	None	
Estimated SLA in Days	123 Number	Sum	
Estimated SLA in Hours	Hour	None	
Issue Category and Class...	ABC Text	None	
Priority	ABC Text	None	
Project Name	ABC Text	None	
Resolution	ABC Text	None	
Status	ABC Text	None	
Total Days Spent for Issues	123 Number	Sum	
Total Hours Spent for Iss...	123 Number	Sum	
Total Time Spent	123 Number	Sum	
METRICS (1)			
Record Count	123 Number	Auto	

Table 2.2: Data Pool

Perform exploratory analysis and modeling

Here we are performing mainly analysis on modeling.

- Develop methodology
- Determine important variables
- Build model
- Assess model

1. Validate the data set

- Evaluate results
- Review process
- Determine next steps
- Reevaluate results

2. Visualize and present the findings

- Communicate results
- Determine best method/graph to present insights based on analysis and audience
- Craft a compelling story
- Make recommendation

CHAPTER 3: METHODOLOGY

A data analytical visualization is to overcome above mentioned problems and implementing a dashboard visualizing to improve the quality of the defect management process and make the high profit to the software company by considering customer dissatisfaction and team commitments.

To achieve this purpose, the target data set is identified and collected. Data set included both categorical and numerical data types.

When collecting primary data set for incidents collected manually as because of the many open data sets are not available in this area. This data set belongs to Apache organization and they have set it in open Jira. Apache organization used Jira as an issue tracking application and data set is available for registered users.

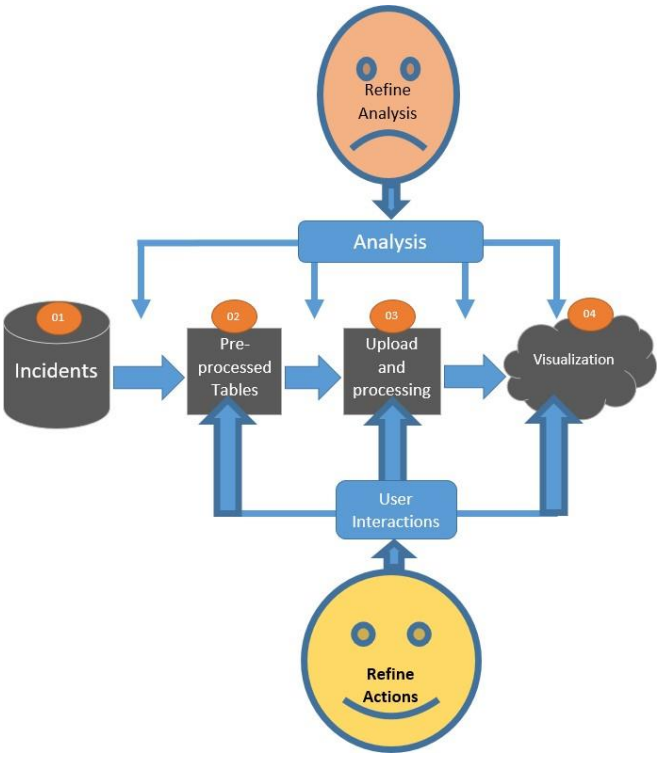


Figure 3.1: Solution Flow

3.1. Collecting Raw Data

Raw data collected as incidents. Incidents were collected from the open source project which is called Apache organization. Here we have 31944 incident counts of the data and it's varying. This study will useonly 20% of above data set and rest will be taken as training data. "Jira is a proprietary issue tracking product developed by Atlassian that allows bug tracking and agile project management. Atlassian provides Jira services to Apache projects. The tool's name is a short form of the name of the Japanese movie monster, Godzilla, and was an early developer nickname for the application. Anyone can review existing Jira tickets, or issues. You must register and log in if you want to create, comment or vote on,or watch issues. Only developers can edit, prioritize, schedule and resolve issues. ASF and many of its projects use Jira to keep track of work to be done. The largest group of tickets assigned to Infra is requests for Infra to perform a task of one sort or another. The next largest category is reports of possible bugs in the Infrastructure system" [12].

At once the incidents were very complex to analyses and insert into correct category. Many incidents needed considerable effort to categorize as a Defect, Explanation, Data Setup Issue, Data Correction, Connectivity Issue, Enhancement or Invalid. As an assumption and suggestion for this time consuming manual work load, we decided that once the incidents were taking to investigation, the assignee was appointed as a responsible person to make and convert the incidents into valuable information. This was a manual task and it was taking much time only for the investigations and it was simplified the update the analyzed information. We introduced new drop down as 'Incident Category' and finally we could label the incident data set and introduced to learning model.

3.2. Preparing Data Tables

After the incidents were categorized accordingly again had to classify as stated by the defects classifications as follows. Please refer the "Table 3-1: Issue category & classification".

Issue Category	Issue Classification
Defect	Incorrect Requirement
	Omitted Requirement
	Misunderstood requirement
	Incorrect design
	Misunderstood design
	Defect - Coding error
	Performance Issue
Explanation	Not a defect
	Unable to Re-create
	Incorrect workflow
	Lack of understanding
	Incomplete
Setup_Issue	Config error
	Data setup
Data_Correction	Not an error
Invalid	Not an error
Enhancement	Enhancement
	Report request

Table 3-2: Issue category & classification

This was a big mess as all the project's artifacts and details had to analyze properly to find out the correct defect classification. In this phase, projects artifacts required to keep up to date information and store properly in a common shared location.

3.3. Generate Visual Structures

With the pre-processed data tables, there's a need of processing structure to prepare the data to visualize. Here we are generating an upload mechanism for pre-processed data tables and developing data processing tool with Google Spread Sheets.

The main purpose of this phase is preparing the information structure for a smooth visualization.

3.4. Visualizations

All the incidents show with the charts, graphs and many more visualization methods using Google Data Studio. It is focused for visualizing Project wise issues, priority, status, Issue category and classification, and also the customer satisfaction/dissatisfaction predictiondashboard and all the client reported queries in dashboard.

3.5. Refine Analysis

Refine analysis is a major action and proceeding in every stage above mentioned. Always there will be improvements for analysis methods. When seeing the data and information, the more visualizing factors always emerge. At the moment it has been identified relevant scope and will precede top onthe analysis stage. This will be analyzed for all four phases.

3.6. Refine Actions

Emerge of analysis methods in the above; there will be a room always for a refine for actions on what we already have taken. This is effecting last three phases and in-progress now.

3.7. Incident Visualization

Why incident visualization is important?

- Identify and explore the major areas of poor technology and weak processes in the business.
- Present the major skill levels of human resources to decrease customer dissatisfaction.
- Categorization and classification of the data set according to the defects.
- Generate statistical report of issue categorization and classification.
- Root cause analysis on classified incidents.
- Predict the customer dissatisfaction.

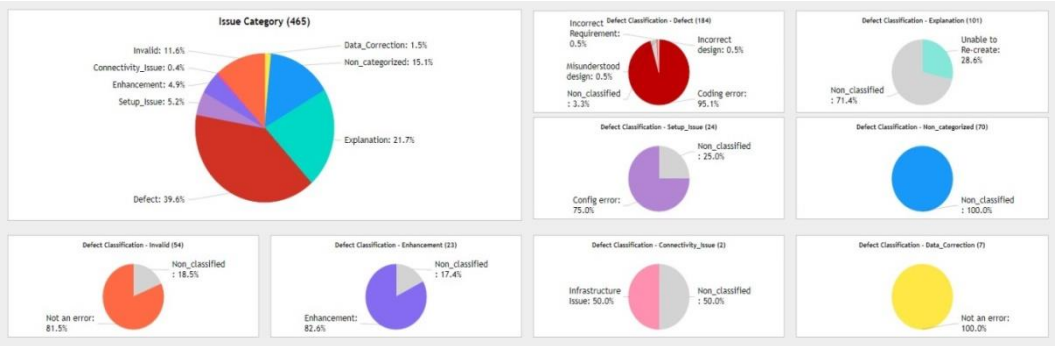


Figure 3.2: Categorization and Classification

Figure 3.3: Time Spent on Fixes - Categorization and Classification

What is the Root Cause Analysis?

"RCA (Root Cause Analysis) is a mechanism of analyzing the Defects, to identify its cause. We

brainstorm, read and dig the defect to identify whether the defect was due to "testing miss", "development miss" or a "requirement or designs miss".When RCA is done accurately, it helps to prevent defects in the later releases or phases. If we find, that a defect was due to design miss, we can review the design documents and can take appropriate measures. Similarly, if we find that a defect was due to testing miss, we can review our test cases or metrics, and update it accordingly" (88 Science and Tech Blogs & Publications That Hire Freelance Writers, 2021). Please refer the "Figure 3.3: Time Spent on Fixes - Categorization and Classification".

What are the advantages of root cause analysis on defects?

- Prevent the reoccurrence of the same problem in the future.
- Eventually, reduce the number of defects reported over time.
- Reduces developmental costs and save time.
- Improve the software development process and hence aiding quick delivery to market.
- Improve customer satisfaction.
- Boost productivity.
- Find hidden problems in the system.
- Provide support and guidance in continuous improvement.

3.8. Client Satisfaction & Dissatisfaction Prediction

The basic target of predict the client satisfaction, it has identified as a qualitative factor. Basically quality cannot be identified using numerical data types directly. So categorical variable created and used it for prediction.

In the data set that "Satisfaction" column with categorical variable represented the satisfaction of client with the given data. As mentioned above, it is difficult to identify whether the client satisfied or not using only the difference between "∑ Time Spent" and "Estimated Time Hours" values. Please refer the "Figure 5.4:Raw Data".

Here we can derive the idea using following formula;

Client Satisfaction = (Estimated Time Hours - ∑ Time Spent)

If 'Answer' <= Estimated Time Hours

Client Satisfied

Otherwise;

Client Not Satisfied

	Priority	Status	Created	Resolution	Updated	Time Spent	Σ Time Spent	Issue Category and Classification	Status Category Changed	[CHART] Date of First Response	Σ Hours Spent for Issues	Σ Days Spent for Issues	Estimated Time Hours	Estimated Time Days	Estimated Hours	Satisfaction	Project Name
0	Medium	Open	9/10/2021 14:36	NaN	9/10/2021 14:36	NaN	NaN	NaN	9/10/2021 14:36	NaN	0.0	0.0	6.0	0.75	6.0	Satsfied	UIMA
1	Medium	Open	9/10/2021 14:17	NaN	9/10/2021 14:17	NaN	NaN	NaN	9/10/2021 14:17	NaN	0.0	0.0	6.0	0.75	6.0	Satsfied	UIMA
2	Medium	Open	9/10/2021 13:57	NaN	9/10/2021 14:02	NaN	NaN	NaN	9/10/2021 13:57	NaN	0.0	0.0	6.0	0.75	6.0	Satsfied	UIMA
3	Medium	Open	9/10/2021 13:51	NaN	9/10/2021 14:03	NaN	NaN	NaN	9/10/2021 13:51	NaN	0.0	0.0	6.0	0.75	6.0	Satsfied	UIMA
4	Medium	Open	9/10/2021 13:49	NaN	9/10/2021 14:04	NaN	NaN	NaN	9/10/2021 13:49	NaN	0.0	0.0	6.0	0.75	6.0	Satsfied	UIMA
5	Medium	Open	9/10/2021 13:26	NaN	9/10/2021 13:26	NaN	NaN	NaN	9/10/2021 13:26	NaN	0.0	0.0	6.0	0.75	6.0	Satsfied	Beam
6	Medium	Open	9/10/2021 13:24	NaN	9/10/2021 13:26	NaN	NaN	NaN	9/10/2021 13:24	NaN	0.0	0.0	6.0	0.75	6.0	Satsfied	Mesos
7	Critical	Open	9/10/2021 10:58	NaN	9/10/2021 10:58	NaN	NaN	NaN	9/10/2021 10:58	NaN	0.0	0.0	4.0	0.50	4.0	Satsfied	Beam
8	Low	Open	9/10/2021 10:53	NaN	9/10/2021 11:41	NaN	NaN	NaN	9/10/2021 10:53	NaN	0.0	0.0	8.0	1.00	8.0	Satsfied	Mesos
9	Medium	Open	9/10/2021 10:18	NaN	9/10/2021 10:18	NaN	NaN	NaN	9/10/2021 10:18	NaN	0.0	0.0	6.0	0.75	6.0	Satsfied	UIMA

Figure 6.4:Raw Data

Data Cleansing and Formatting

There were Non-required columns set in the data set. After thorough investigation, decided that data were not necessary for the analysis. Removed the unnecessary data and prepared the data set for further analysis. Figure 3.5: Data Cleansing has shown it below.

```
df.drop(['Updated','Time Spent','Σ Hours Spent for Issues', 'Σ Days Spent for Issues','Status Category Changed','Estimated Time Hours', 'Estimated Time Days'],axis=1,inplace=True)
```

Figure 3.5: Data Cleansing

Refined data set shows below; in Figure 3.6: Redefined Data Set

	Priority	Status	Created	Resolution	Σ Time Spent	Issue Category and Classification	[CHART] Date of First Response	Estimated Hours	Satisfaction	Project Name
0	Medium	Open	9/10/2021 14:36	NaN	NaN	NaN	NaN	6.0	Satsfied	UIMA
1	Medium	Open	9/10/2021 14:17	NaN	NaN	NaN	NaN	6.0	Satsfied	UIMA
2	Medium	Open	9/10/2021 13:57	NaN	NaN	NaN	NaN	6.0	Satsfied	UIMA
3	Medium	Open	9/10/2021 13:51	NaN	NaN	NaN	NaN	6.0	Satsfied	UIMA
4	Medium	Open	9/10/2021 13:49	NaN	NaN	NaN	NaN	6.0	Satsfied	UIMA

Figure 3.6: Redefined Data Set

And also identified few columns to re-name otherwise conflict may rise from the visualization dashboard data and prediction data as same data set used. Maintained consistency of data set through renamed and pre-processed. Please refer “Figure 3.7: Pre-Processed Data Set”.

	Priority	Status	Created Date	Resolution	Total Time Spent	Issue Category and Classification	Date of First Response	Estimated SLA in Hours	Client Satisfaction	Project Name
0	Medium	Open	9/10/2021 14:36	NaN	NaN	NaN	NaN	6.0	Satsfied	UIMA
1	Medium	Open	9/10/2021 14:17	NaN	NaN	NaN	NaN	6.0	Satsfied	UIMA
2	Medium	Open	9/10/2021 13:57	NaN	NaN	NaN	NaN	6.0	Satsfied	UIMA
3	Medium	Open	9/10/2021 13:51	NaN	NaN	NaN	NaN	6.0	Satsfied	UIMA
4	Medium	Open	9/10/2021 13:49	NaN	NaN	NaN	NaN	6.0	Satsfied	UIMA

Figure 3.7: Pre-Processed Data Set

There were many numbers of null values in the data fields and needed to handle and refine null values in the data set for gain the normal data distribution.

The null value distribution showed the clear idea of the null value distribution which have spread all over the data set. (Figure 3.8: Null Value Distribution)

```
Priority                            0
Status                              0
Created Date                        0
Resolution                       2013
Total Time Spent                10915
Issue Category and Classification    28220
Date of First Response           6252
Estimated SLA in Hours             44
Client Satisfaction                44
Project Name                       13
dtype: int64
```

Figure 3.8: Null Value Distribution

Removed the data points which were having null values in the 'Project Name', 'Client Satisfaction', and 'Estimated SLA in Hours' and 'Date of First Response' columns.(Figure 3.9:After Remove Null Values)

```
Priority                            0
Status                              0
Created Date                        0
Resolution                        579
Total Time Spent                 8689
Issue Category and Classification    22273
Date of First Response              0
Estimated SLA in Hours              0
Client Satisfaction                 0
Project Name                        0
dtype: int64
```

Figure 3.9:After Remove Null Values

After renamed and pre-processed the data set, it made sure the unique values in columns which were going to use for analysis. This helped to ensure that the duplicate values were not recorded in the data set further. (Figure 3.10: Unique Values)

```
# unique values in Resolution
df['Resolution'].unique()
```

```
array(['Fixed', nan, 'Await Client Feedback', 'Invalid', 'Responded',
       'Duplicate', 'Await 3rd Party Feedback', 'Later', 'Done',
       'Cannot Reproduce', 'Works for Me', "Won't Fix", "Won't Do",
       'Incomplete'], dtype=object)
```

```
# unique values in Priority
df['Priority'].unique()
```

```
array(['Medium', 'Critical', 'Blocker', 'Low'], dtype=object)
```

```
# unique values in Status
df['Status'].unique()
```

```
array(['Resolved', 'Verified', 'In Progress', 'Open', 'On Hold', 'Closed',
       'Reopened'], dtype=object)
```

```
# unique values in Issue Category and Classification
df['Issue Category and Classification'].unique()
```

```
array([nan, 'Defect - Incorrect design', 'Invalid - Not an error',
       'Defect - Coding error', 'Defect - Omitted Requirement',
       'Explanation - Not a defect', 'Setup_Issue - Data setup', 'Defect',
       'Setup_Issue', 'Enhancement - Enhancement', 'Invalid',
       'Explanation', 'Setup_Issue - Config error',
       'Data_Correction - Not an error', 'Enhancement',
       'Explanation - Unable to Re-create',
       'Explanation - Lack of understanding',
       'Connectivity_Issue - Third party Issue',
       'Enhancement - Report request', 'Defect - Performance Issue',
       'Data_Correction', 'Explanation - Incorrect workflow',
       'Connectivity_Issue', 'Explanation - Incomplete',
       'Defect - Misunderstood requirement',
       'Connectivity_Issue - Infrastructure Issue',
       'Defect - Misunderstood design', 'Defect - Incorrect Requirement'],
      dtype=object)
```

Figure 3.10: Unique Values

The ‘Total Time Spent’ columns had null values. It means task didn’t complete and the data set had to breakdown into two data sets. One data set was which had time spent and other one was which hadn’t log time spent.(Figure 3.11: Time Spent - Null vs Not Null)

```
df_time_spent_null.head()
```

	Priority	Status	Created Date	Resolution	Issue Category and Classification	Date of First Response	Estimated SLA in Hours	Client Satisfaction	Project Name
0	Medium	Resolved	2021-09-10 08:25:00	Fixed	N/A	2021-09-10 09:13:00	6.0	Satsfied	Apache Cordova
1	Critical	In Progress	2021-09-10 05:14:00	N/A	N/A	2021-09-10 06:37:00	4.0	Satsfied	Mesos
2	Medium	Open	2021-09-10 05:06:00	N/A	N/A	2021-09-10 05:13:00	6.0	Satsfied	Beam
3	Critical	On Hold	2021-09-10 00:27:00	Await Client Feedback	N/A	2021-09-10 14:18:00	4.0	Satsfied	Mesos
4	Medium	Closed	2021-09-09 21:01:00	Fixed	N/A	2021-09-10 02:40:00	6.0	Satsfied	UIMA

```
df_time_spent_not_null.head()
```

	Priority	Status	Created Date	Resolution	Total Time Spent	Issue Category and Classification	Date of First Response	Estimated SLA in Hours	Client Satisfaction	Project Name
0	Critical	Verified	2021-09-10 07:07:00	Fixed	7200.0	Defect - Incorrect design	2021-09-10 11:23:00	4.0	Satsfied	Mesos
1	Critical	Open	2021-09-10 03:20:00	N/A	14400.0	N/A	2021-09-10 04:04:00	4.0	Not Satisfied	Mesos
2	Critical	Resolved	2021-09-09 17:39:00	Invalid	10800.0	Invalid - Not an error	2021-09-10 10:16:00	4.0	Satsfied	Mesos
3	Medium	In Progress	2021-09-09 12:19:00	Await Client Feedback	1800.0	N/A	2021-09-09 12:33:00	6.0	Satsfied	Beam
4	Blocker	On Hold	2021-09-09 12:12:00	Await Client Feedback	21600.0	N/A	2021-09-10 11:17:00	2.0	Not Satisfied	Mesos

Figure 3.11: Time Spent - Null vs Not Null

Created feature to find the time taken for first response to the client. This time shows in hours. Now had created two data sets; one for Time spent for null and another for Time spent for not null. Data frames shuffled to avoid the ordered data points. It was ensured that the null value distribution had no distribution. That means null value impact for analysis was less.

Need to make sure that the client satisfaction actually depends on what? Then analyzed with the data in "Client Satisfied" column whether the client satisfied or not according to the projects in the organization. Projects were consisted with different clients. (Figure 3.12: Client Satisfaction of Projects)

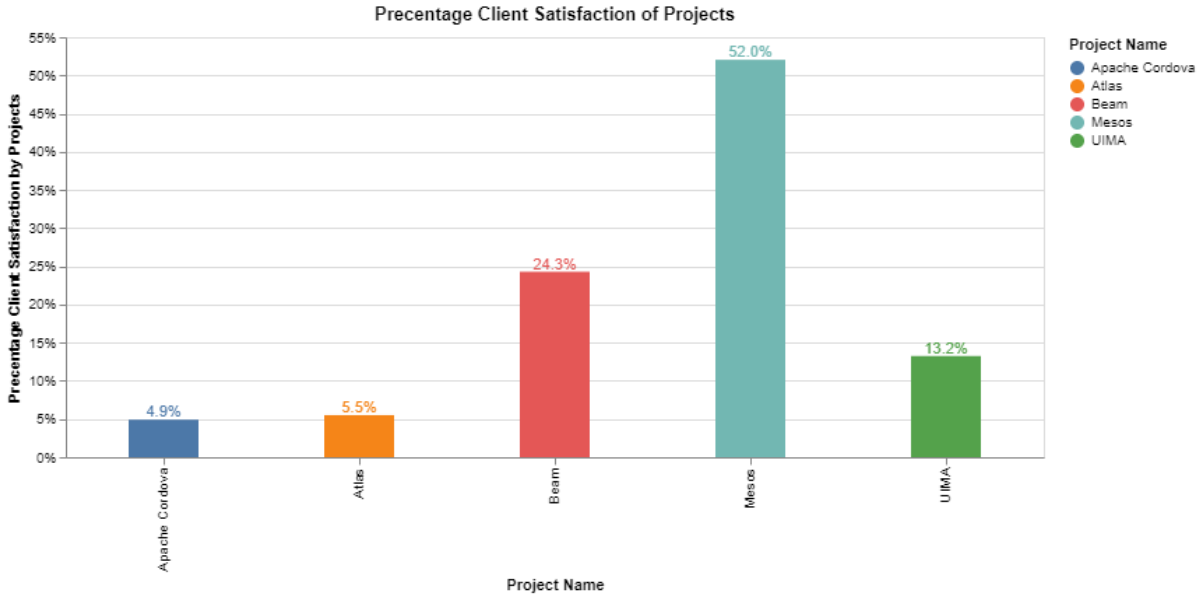


Figure 3.12: Client Satisfaction of Projects

As well as it could analyze the client dissatisfaction on projects. In the below it displayed the visualized details of the client dissatisfaction. (Figure 3.13: Client Dissatisfaction of Projects)

Figure 3.13: Client Dissatisfaction of Projects

With this analyzed data, it was recommended that the both satisfied and not-satisfied data frames were ready to train the model further.

Outlier handling

Outlier is a major task. We have taken 'Total Time Spent', 'Estimated SLA in Hours' and 'Time Taken for First Response' columns and had removed outliers in those columns.

Total Time Spent

There were thirteen (13) points outliers according to the box plot and removed all. (Figure 3.14: Outlier in Total Time Spent)

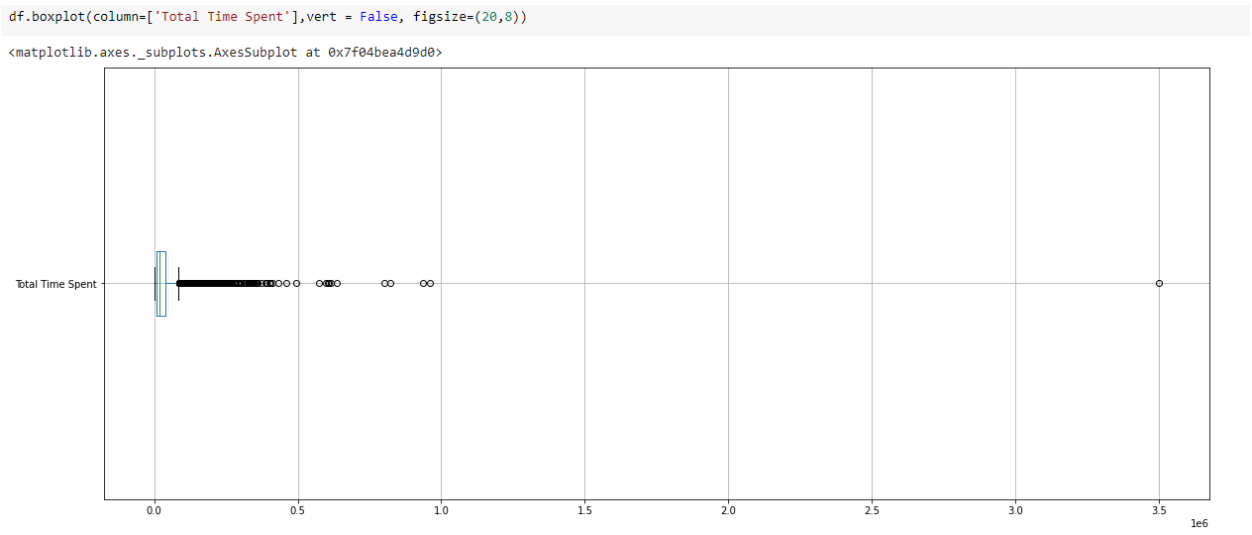


Figure 3.14: Outlier in Total Time Spent

Figure 3.15: Outliers Before & After shows after removing outliers.

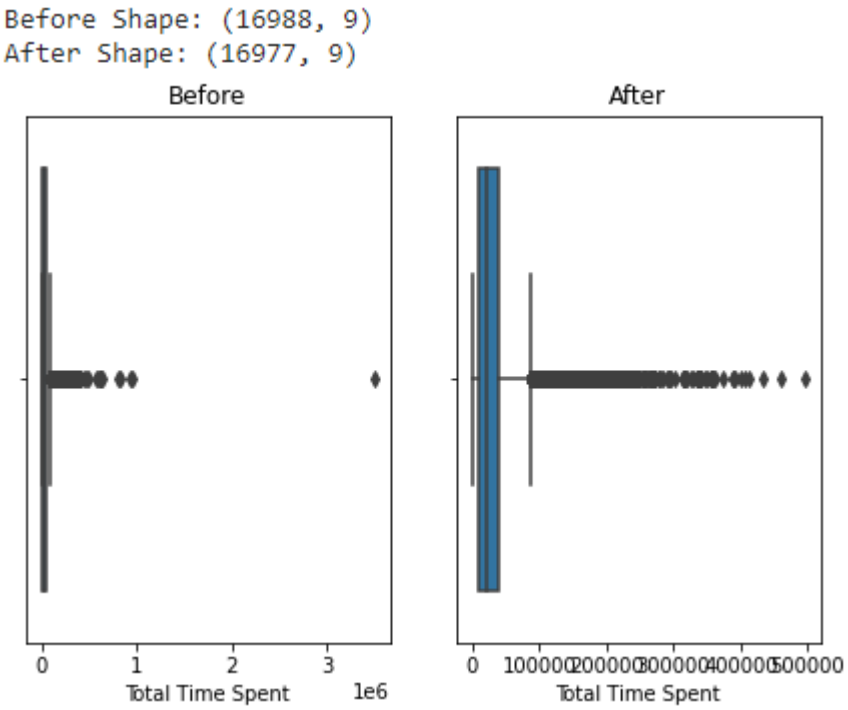


Figure 3.15: Outliers Before & After

Estimated SLA in Hours

It seems no outliers according to the box plot. Estimated SLA in Hours was static values and it was pre-defined. Other than the human mistake there cannot be presented outliers. (Figure 3.16: Outlier in Estimated SLA in Hours)

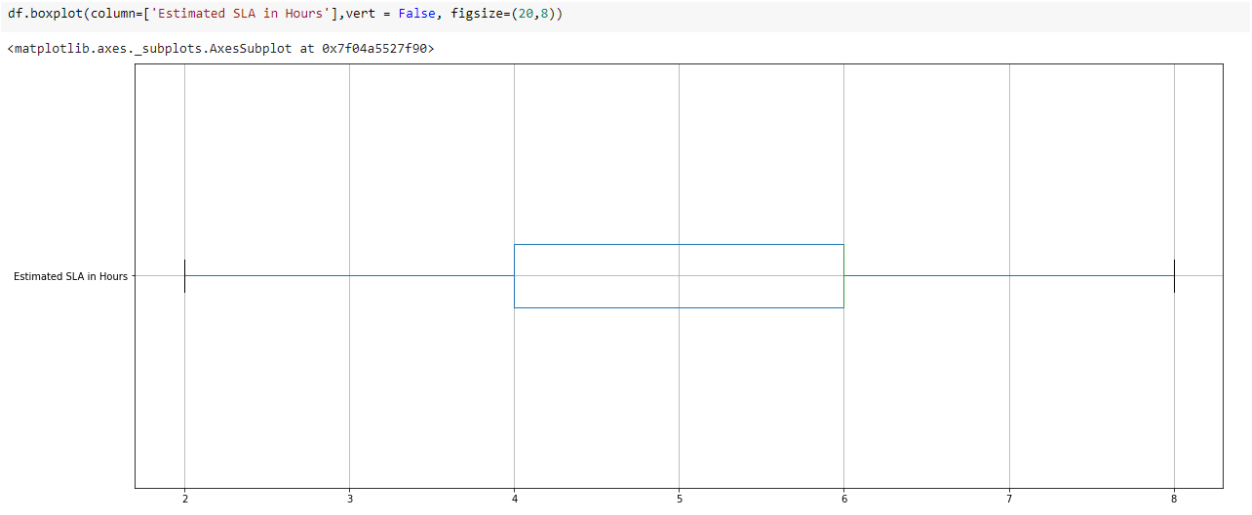


Figure 3.16: Outlier in Estimated SLA in Hours

Time Taken for First Response

There were thirty nine (39) points outliers according to the box plot and removed all. (Figure

3.17: Outliers in Time Taken for First Response)

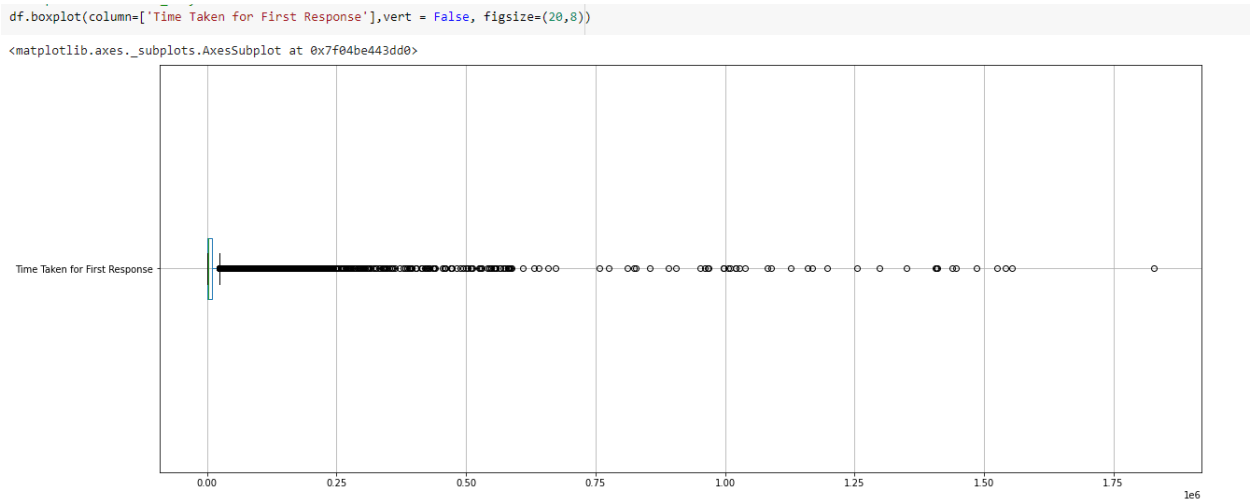


Figure 3.17: Outliers in Time Taken for First Response

Figure 3.18: Outlier before and after in Time Taken for First Response shows clearly.

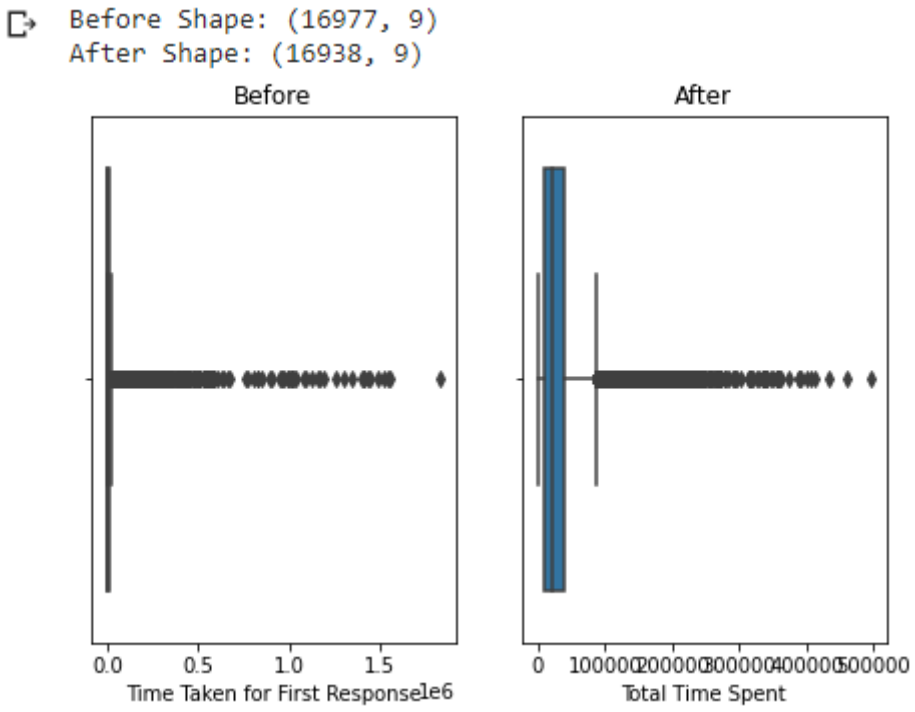


Figure 3.18: Outlier before and after in Time Taken for First Response

Outlier handling was completed successfully.

Feature Transformation

Feature transformation was needed to the process of modifying data but kept the information. These modifications made Machine Learning algorithm understanding easier, which delivered better result with reduced repetition, improved performance, and maintained data integrity.

Need to plot the histograms and Q-Q plots to see the normal distribution.

Estimated SLA in Hours

The ‘Estimated SLA in Hours’ column data distribution was discrete feature and it only contained four values. (Figure 3.19: Discrete Distribution)

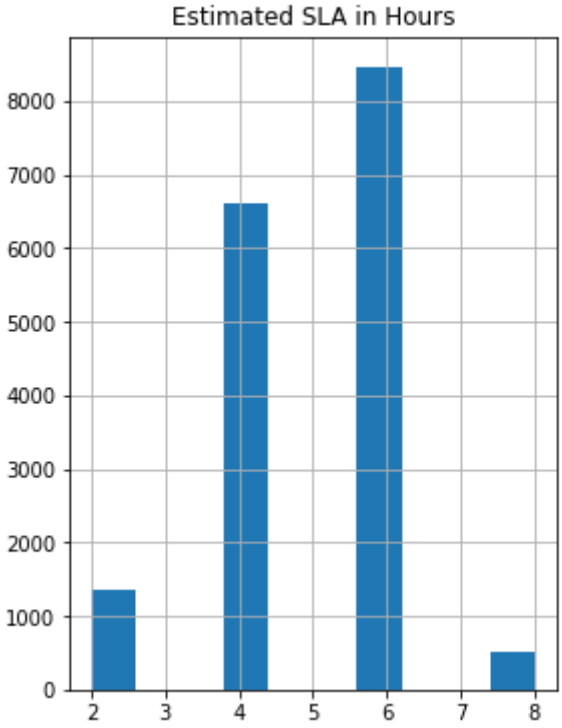


Figure 3.19: Discrete Distribution

Total Time Spent

The ‘Total Time Spent’ showed clear distribution of data spreading and it was right skewness data spread. There was a huge skewness on to the right side. Here applied SQRT transformation to reduce the skewness. This continued several times inserting new values and finalized values identified for analysis.(Figure 3.20: Right Skewness)

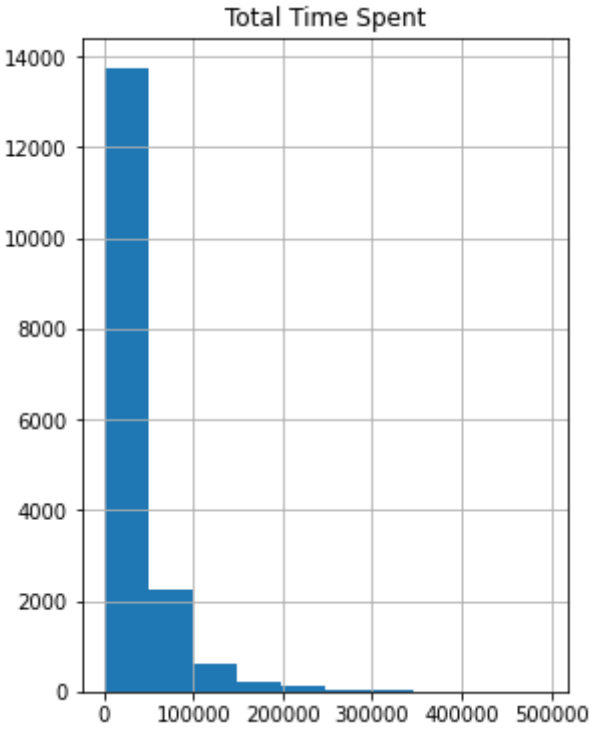


Figure 3.20: Right Skewness

After SQRT transformation the skewness reduced comparatively. The data was distributed well. (Figure 3.21: Data Distribution)

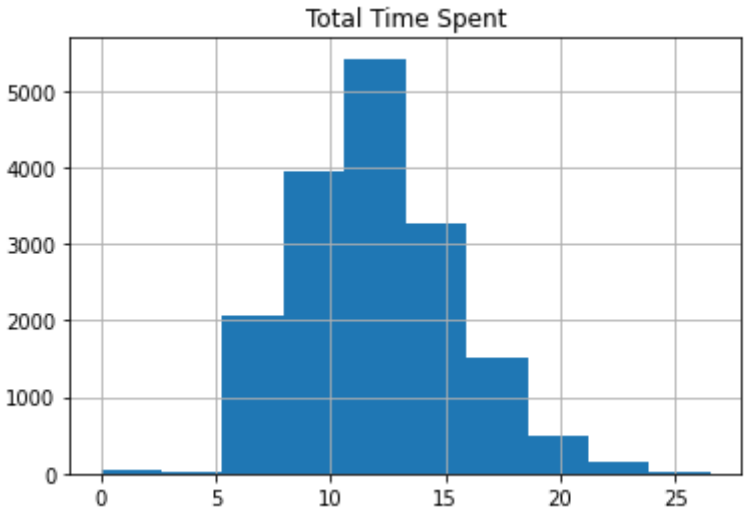


Figure 3.21: Data Distribution

Time Taken for First Response

Needed to check the skewness first and it was right skewness. (Figure 3.22: Left Skewness)

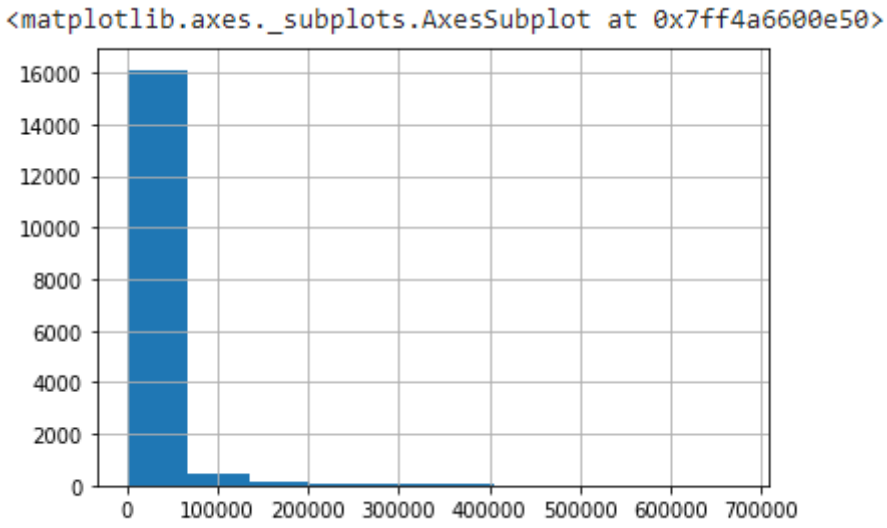


Figure 3.22: Left Skewness

Check again and applied SQRT transformation and its well distributed after that. (Figure 3.23: Distribution)

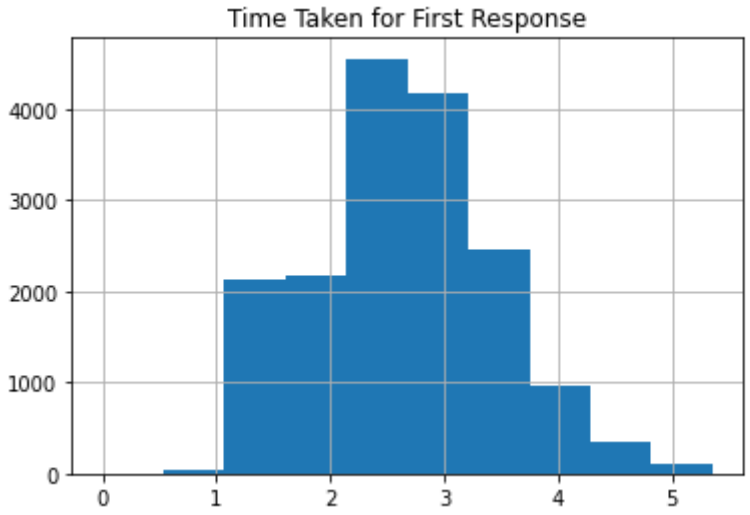


Figure 3.23: Distribution

Data frames were ready to analyze and well distributed and also feature transformation completed successfully. (Figure 3.24: After SQRT)

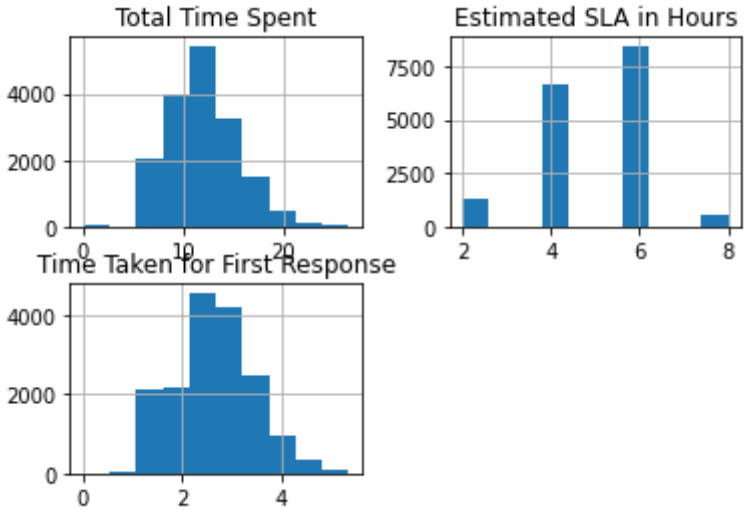


Figure 3.24: After SQRT

Label Encoding

Here, used label encoding since, the target was categorical and the domain is classification and it did not create order of label.

Label Encoding converted the labels into a numeric form so as to convert them into the machine-readable form. Machine learning algorithms able to decide in a correct way how those labels must be operated. It was an important pre-processing step for the structured dataset in supervised learning. (Figure 3.25: Before Encoding)

```
df.head()
```

	Priority	Status	Resolution	Total Time Spent	Issue Category and Classification	Estimated SLA in Hours	Client Satisfaction	Project Name	Time Taken for First Response
0	Critical	Closed	Done	15.491933	N/A	4.0	Not Satisfied	Mesos	3.605756
1	Critical	Closed	Fixed	15.611571	N/A	4.0	Not Satisfied	Atlas	1.444921
2	Critical	Resolved	Responded	12.599408	N/A	4.0	Not Satisfied	Mesos	2.792989
3	Medium	Closed	Done	12.599408	N/A	6.0	Not Satisfied	Beam	1.147203
4	Critical	Closed	Fixed	11.175865	N/A	4.0	Not Satisfied	Mesos	2.531567

Figure 3.25: Before Encoding

```
priority_codes = dict()
index = 0
for topic in list(df['Priority'].unique()):
    priority_codes[topic] = index
    index += 1

priority_codes
```

```
{'Blocker': 2, 'Critical': 0, 'Low': 3, 'Medium': 1}
```

Figure 3.26: Priority Encoding

```
status_codes = dict()
index = 0
for topic in list(df['Status'].unique()):
    status_codes[topic] = index
    index += 1

status_codes
```

```
{'Closed': 0,
 'In Progress': 6,
 'On Hold': 3,
 'Open': 5,
 'Reopened': 4,
 'Resolved': 1,
 'Verified': 2}
```

Figure 3.27: Status Encoding

```
resolution_codes = dict()
index = 0
for topic in list(df['Resolution'].unique()):
    resolution_codes[topic] = index
    index += 1

resolution_codes
```

```
{'Await 3rd Party Feedback': 12,
 'Await Client Feedback': 3,
 'Cannot Reproduce': 7,
 'Done': 0,
 'Duplicate': 6,
 'Fixed': 1,
 'Incomplete': 13,
 'Invalid': 4,
 'Later': 9,
 'N/A': 5,
 'Responded': 2,
 "Won't Do": 11,
 "Won't Fix": 10,
 'Works for Me': 8}
```

Figure 3.28: Resolution Encoding

```
issue_codes = dict()
index = 0
for topic in list(df['Issue Category and Classification'].unique()):
    issue_codes[topic] = index
    index += 1

issue_codes
```

```
{'Connectivity_Issue': 24,
 'Connectivity_Issue - Infrastructure Issue': 12,
 'Connectivity_Issue - Third party Issue': 14,
 'Data_Correction': 13,
 'Data_Correction - Not an error': 6,
 'Defect': 9,
 'Defect - Coding error': 4,
 'Defect - Incorrect Requirement': 27,
 'Defect - Incorrect design': 7,
 'Defect - Misunderstood design': 17,
 'Defect - Misunderstood requirement': 5,
 'Defect - Omitted Requirement': 16,
 'Defect - Performance Issue': 23,
 'Enhancement': 19,
 'Enhancement - Enhancement': 8,
 'Enhancement - Report request': 25,
 'Explanation': 15,
 'Explanation - Incomplete': 22,
 'Explanation - Incorrect workflow': 21,
 'Explanation - Lack of understanding': 26,
 'Explanation - Not a defect': 1,
 'Explanation - Unable to Re-create': 18,
 'Invalid': 20,
 'Invalid - Not an error': 11,
 'N/A': 0,
 'Setup_Issue': 3,
 'Setup_Issue - Config error': 2,
 'Setup_Issue - Data setup': 10}
```

Figure 3.29: Issue Encoding

```
client_satis_codes = dict()
index = 0
for topic in list(df['Client Satisfaction'].unique()):
    client_satis_codes[topic] = index
    index += 1

client_satis_codes
```

```
{'Not Satisfied': 0, 'Satsfied': 1}
```

Figure 3.30: Client Satisfaction Encoding

```
project_codes = dict()
index = 0
for topic in list(df['Project Name'].unique()):
    project_codes[topic] = index
    index += 1

project_codes
```

```
{'Apache Cordova': 4, 'Atlas': 1, 'Beam': 2, 'Mesos': 0, 'UIMA': 3}
```

Figure 3.31: Project Encoding

	Priority	Status	Resolution	Total Time Spent	Issue Category and Classification	Estimated SLA in Hours	Client Satisfaction	Project Name	Time Taken for First Response
0	0	0	0	15.491933	0	4.0	0	0	3.605756
1	0	0	1	15.611571	0	4.0	0	1	1.444921
2	0	1	2	12.599408	0	4.0	0	0	2.792989
3	1	0	0	12.599408	0	6.0	0	2	1.147203
4	0	0	1	11.175865	0	4.0	0	0	2.531567

Figure 3.32: After Encoding

Avoid Data Leak (Breach):

The data set was split into two sets in order to prevent data leak from train data to test data samples. Test data selected 20% of the data set and rest of 80% training data. (Figure 3.33: Define X and Target Y)

```
df.columns
```

```
Index(['Priority', 'Status', 'Resolution', 'Total Time Spent',
       'Issue Category and Classification', 'Estimated SLA in Hours',
       'Client Satisfaction', 'Project Name', 'Time Taken for First Response'],
      dtype='object')
```

```
# define features as X and target as y

X = df.drop('Client Satisfaction',axis=1)
y = pd.DataFrame(df['Client Satisfaction'],columns = ['Client Satisfaction'])
```

```
from sklearn.model_selection import train_test_split

# spliting dataset
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.2,random_state = 101)

# reset indexes
X_train = X_train.reset_index(drop=True)
X_test = X_test.reset_index(drop=True)
y_train = y_train.reset_index(drop=True)
y_test = y_test.reset_index(drop=True)

X_train.head(5)
```

Figure 3.33: Define X and Target Y

	Priority	Status	Resolution	Total Time Spent	Issue Category and Classification	Estimated SLA in Hours	Project Name	Time Taken for First Response
0	0	0	2	10.594797	0	4.0	4	1.553942
1	1	0	2	17.629674	16	6.0	1	1.349504
2	1	0	1	9.211559	0	6.0	1	3.623386
3	3	0	2	8.909127	0	8.0	1	1.801629
4	0	0	0	10.954451	10	4.0	1	3.904119

```
y_train.head()
```

	Client Satisfaction
0	1
1	0
2	1
3	1
4	0

Figure 3.34: X and Y Tables

```
print(f'X_train shape = {X_train.shape}, y_train shape = {y_train.shape}, X_test shape = {X_test.shape}, y_test shape = {y_test.shape}')
```

```
X_train shape = (13550, 8), y_train shape = (13550, 1), X_test shape = (3388, 8), y_test shape = (3388, 1)
```

Figure 3.35: Print Training X and Y

Feature Scaling (Standard Scaling)

Feature scaling method used to normalize the features of data. It was performed during the data preprocessing step. (Figure 3.36: Feature Normalizing)

```
from sklearn.preprocessing import StandardScaler

# Removing Categorical Features before the feature scaling
columns = X_train.columns
columns_new = np.delete(columns,[0,1,2,4,5,6])
removed_columns = np.delete(columns,[3,7])

# Applying Standardization
# Init StandardScaler
scaler = StandardScaler()

#Transformation of training dataset features
X_train_except = pd.DataFrame(X_train, columns = columns_new)
scaler.fit(X_train_except)
X_train = pd.DataFrame(scaler.transform(X_train_except), columns = columns_new).join(X_train[removed_columns])

#Transformation of testing dataset features
X_test_except = pd.DataFrame(X_test, columns = columns_new)
X_test = pd.DataFrame(scaler.transform(X_test_except), columns = columns_new).join(X_test[removed_columns])

X_train.head()
```

	Total Time Spent	Time Taken for First Response	Priority	Status	Resolution	Issue Category and Classification	Estimated SLA in Hours	Project Name
0	-0.429681	-1.344051	0	0	2	0	4.0	4
1	1.659496	-1.598325	1	0	2	16	6.0	1
2	-0.840467	1.229859	1	0	1	0	6.0	1
3	-0.930281	-1.035986	3	0	2	0	8.0	1
4	-0.322873	1.579025	0	0	0	10	4.0	1

Figure 3.36: Feature Normalizing

The ‘Total Time Spent’ and ‘Time Taken for First Response' scaled in feature scaling. The histogram showed it clearly. (Figure 3.37: Feature Scaling)

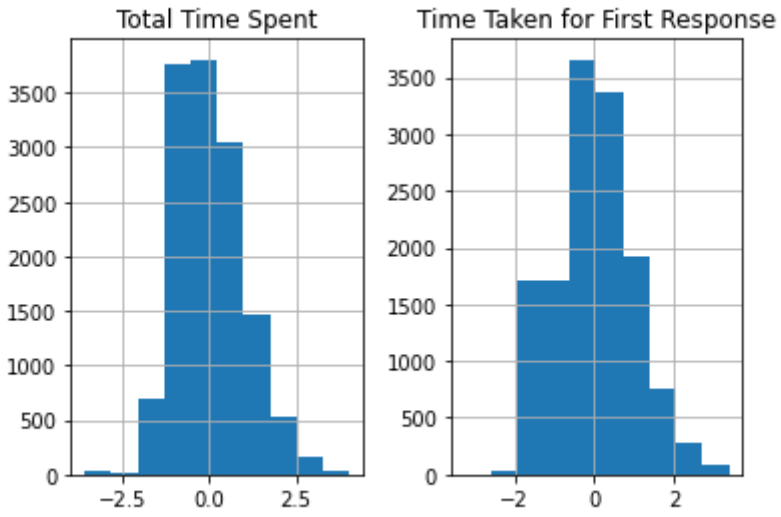


Figure 3.37: Feature Scaling

Feature Engineering

Feature engineering used in the process of transforming raw data into features that higher standard represents the incidents to the predictive models, resulting in improved model accuracy on unseen data. (Figure 3.38: Heat Map)

```
import seaborn as sns
print(X_train.corr());
plt.figure(figsize = (10,10))
sns.heatmap(X_train.corr());
```

```
                                Total Time Spent  ...  Project Name
Total Time Spent                        1.000000  ...      0.096128
Time Taken for First Response           0.116877  ...      0.015224
Priority                               -0.036474  ...      0.055973
Status                                 -0.010967  ...      0.049043
Resolution                             -0.090417  ...      0.101422
Issue Category and Classification      -0.007058  ...     -0.087313
Estimated SLA in Hours                 -0.064398  ...      0.061347
Project Name                            0.096128  ...      1.000000

[8 rows x 8 columns]
```

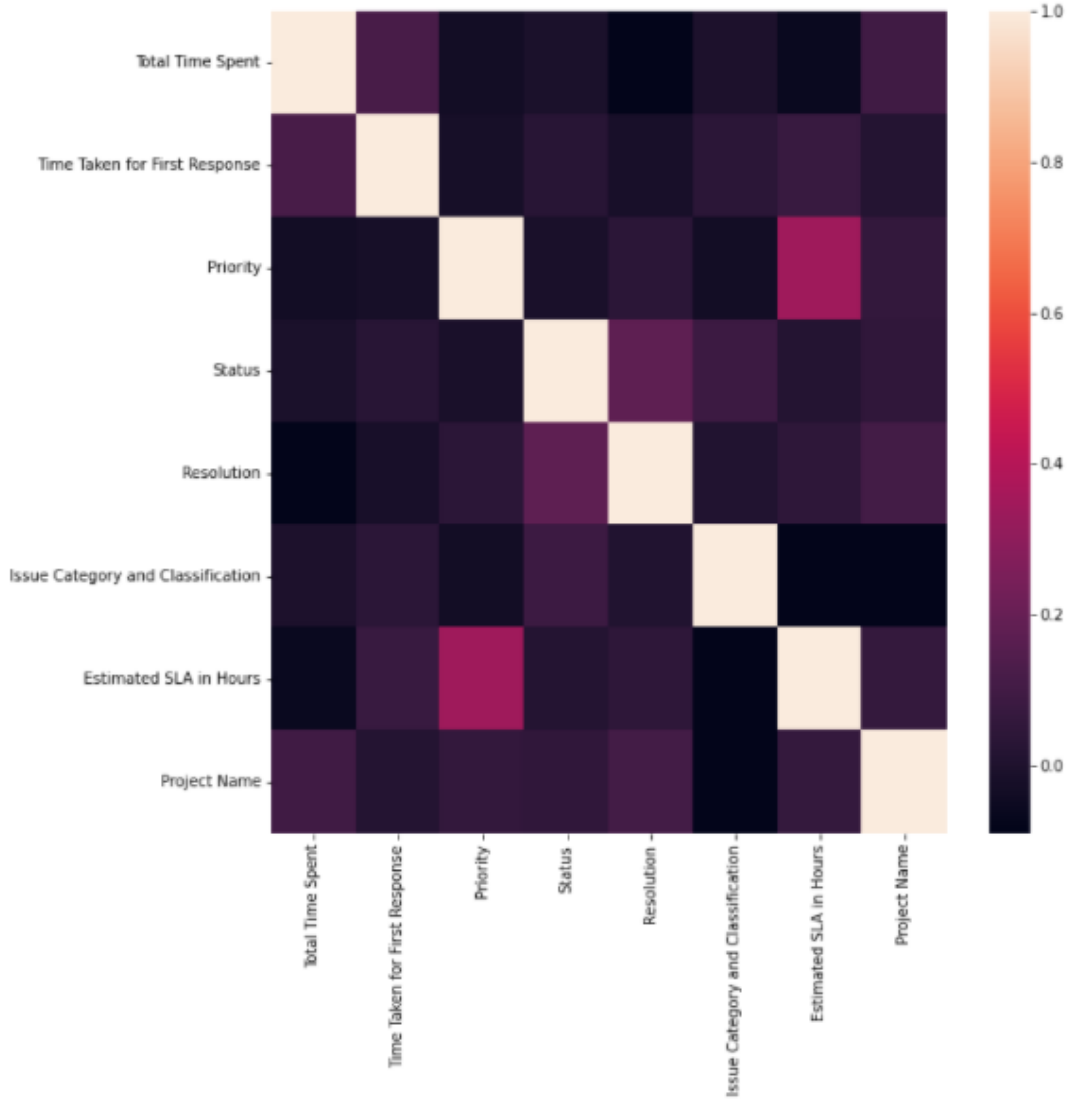


Figure 3.38: Heat Map

All features seemed as independent features, since, there were no significant correlation to moderate or higher correlation. Therefor the significant and Independent features selected as features in training data set for target data.

1. Total Time Spent

2. Priority

3. Estimated SLA in Hours

	Total Time Spent	Priority	Estimated SLA in Hours
0	-0.429681	0	4.0
1	1.659496	1	6.0
2	-0.840467	1	6.0
3	-0.930281	3	8.0
4	-0.322873	0	4.0

Figure 3.39: Selected Significant Data Columns and Data

CHAPTER 4: EVALUATION AND RESULTS

4.1. Model Training

Support Vector Machine

The reason for selecting the support vector machines (SVMs) was, setup of supervised learning methods used for classification and outlier detection. The advantages of support vector machines were: Effective in high dimensional spaces.

According to the above analysis, following three data columns had selected as follows. (Figure 4-7: Selected Significant Columns)

```
X_train.head()
```

	Total Time Spent	Priority	Estimated SLA in Hours
0	-0.326000	0	4.0
1	-1.031235	0	4.0
2	-0.685530	1	6.0
3	0.607214	0	4.0
4	0.287615	2	2.0

Figure 4-8: Selected Significant Columns

Define the X and Y axis in SVM. (Figure 4-9: Client Satisfaction Show)

```
temp = pd.concat([X_train,y_train],axis=1)
```

```
temp.head()
```

	Total Time Spent	Priority	Estimated SLA in Hours	Client Satisfaction
0	-0.326000	0	4.0	0
1	-1.031235	0	4.0	1
2	-0.685530	1	6.0	1
3	0.607214	0	4.0	0
4	0.287615	2	2.0	0

Figure 4-10: Client Satisfaction Show

Client satisfaction reviewing in “Figure 4-11: Satisfied or Not”.

```
client_satis_codes_rev = {0 : 'Not Satisfied', 1 : 'Satsfied'}
```

Figure 4-12: Satisfied or Not

Linear Separable

The 3D scatter plot used to check that the data set is linear separable for put into SVM. (Figure 4-13: Linear Separable)

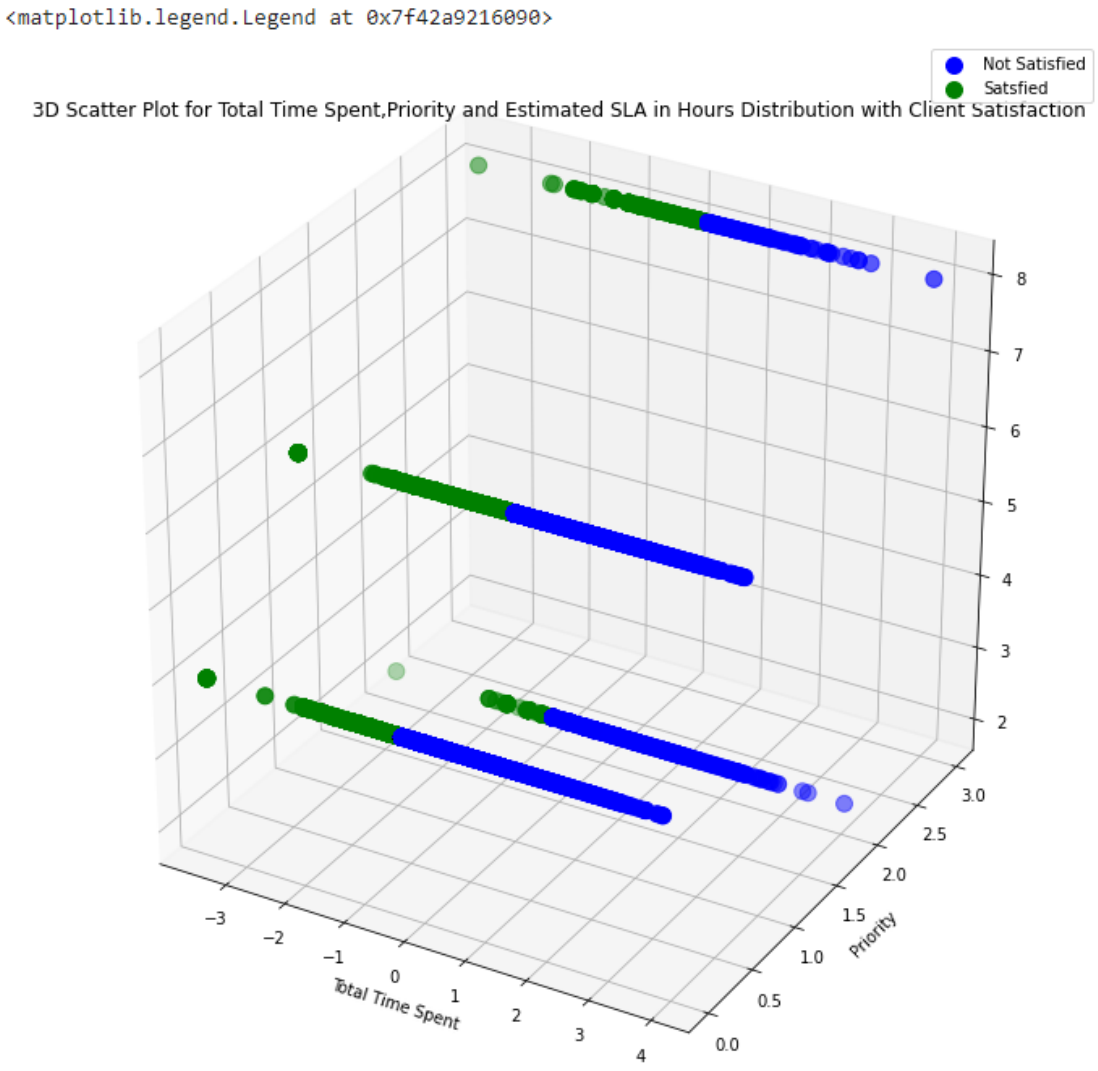


Figure 4-14: Linear Separable

The client satisfaction is linear separable according to the 3D scatter plot. So, we can apply linear **SVM**

Defined SVM Classifier

```
from sklearn import svm
_C = 0.1 # tune SVM regularization parameter 0.1
_gamma = 1.0 # tune gama to 1.0
svc = svm.SVC(kernel='linear', C=_C, gamma=_gamma).fit(X_train, y_train)
```

Figure 4-15: SVM Classifier

4.2. Model evaluation

Model evaluation aims to estimate the generalization accuracy of a model on future (unseen/out-of-sample) data.

Training Accuracy

```
print("The training accuracy is: ")
print(accuracy_score(y_train, svc.predict(X_train)))
```

```
The training accuracy is:
0.9886346863468635
```

Figure 4-16: Training Accuracy

Testing Accuracy

```
print("The test accuracy is: ")
print(accuracy_score(y_test, svc.predict(X_test)))
```

```
The test accuracy is:
0.9902597402597403
```

Figure 4-17: Testing Accuracy

Classification Report

```
print("Classification report for training")
print(classification_report(y_train,svc.predict(X_train)))
```

```
Classification report for training
              precision    recall  f1-score   support

           0       0.98      1.00      0.99      7713
           1       1.00      0.98      0.99      5837

    accuracy                           0.99     13550
   macro avg       0.99      0.99      0.99     13550
weighted avg       0.99      0.99      0.99     13550
```

Figure 4-18: Classification Report for Training

```
print("Classification report for testing")
print(classification_report(y_test,svc.predict(X_test)))
```

```
Classification report for testing
              precision    recall  f1-score   support

           0       0.99      1.00      0.99      1919
           1       0.99      0.98      0.99      1469

    accuracy                           0.99      3388
   macro avg       0.99      0.99      0.99      3388
weighted avg       0.99      0.99      0.99      3388
```

Figure 4-19: Classification Report for Testing

Confusion Matrix

"When we get the data, after data cleaning, pre-processing, and wrangling, the first step we do is to feed it to an outstanding model and of course, get output in probabilities" (Understanding Confusion Matrix - Towards Data Science, 2021).

True positives: Predict an observation belongs to a class and it actually does belong to that class.

True negatives: Predict an observation does not belong to a class and it actually does not belong to that class.

False positives: Predict an observation belongs to a class when in reality it does not.

False negatives: Predict an observation does not belong to a class when in fact it does.

These four outcomes were plotted on "Figure 4-20: Confusion Matrix for Training" and "Figure 4-21: Confusion Matrix for Testing" confusion matrix.

```
print('Confusion Matrix for Training')
cnf_matrix = confusion_matrix(y_train, svc.predict(X_train))
plt.figure(figsize = (10,10))
fig, ax = plt.subplots(1)
ax = sns.heatmap(cnf_matrix, ax=ax, annot=True,cmap="YlGnBu")
plt.title('Confusion matrix')
plt.ylabel('True category')
plt.xlabel('Predicted category')
plt.show()
```

```
Confusion Matrix for Training
<Figure size 720x720 with 0 Axes>
```

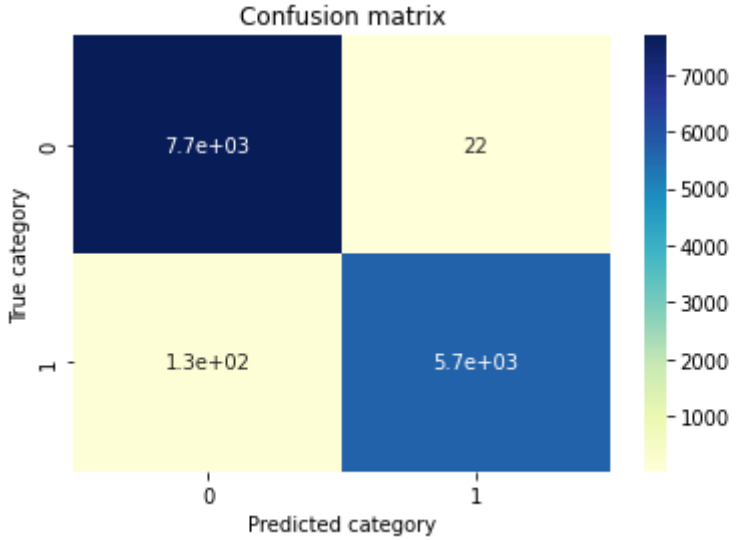


Figure 4-22: Confusion Matrix for Training

```
print('Confusion Matrix for Testing')
cnf_matrix = confusion_matrix(y_test, svc.predict(X_test))
plt.figure(figsize = (10,10))
fig, ax = plt.subplots(1)
ax = sns.heatmap(cnf_matrix, ax=ax, annot=True,cmap="YlGnBu")
plt.title('Confusion matrix')
plt.ylabel('True category')
plt.xlabel('Predicted category')
plt.show()
```

```
Confusion Matrix for Testing
<Figure size 720x720 with 0 Axes>
```

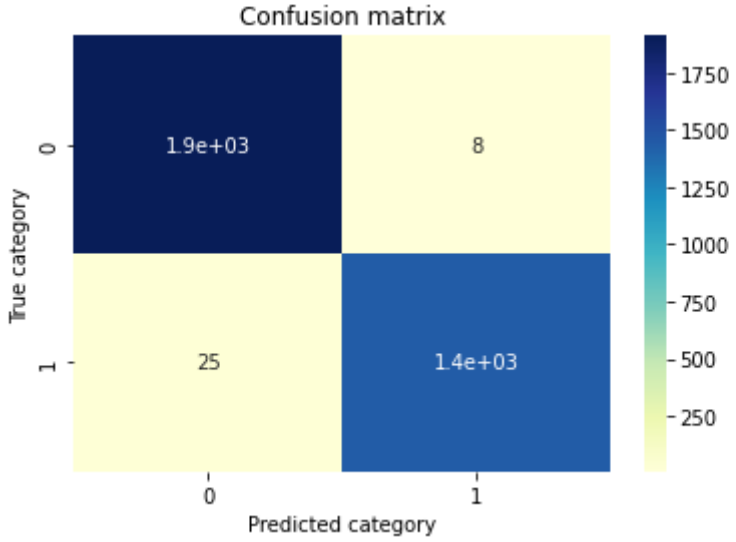


Figure 4-23: Confusion Matrix for Testing

4.3. Model summary

```
model = {
     'Model': 'Support Vector Machine',
     'Training Set Accuracy': accuracy_score(y_train, svc.predict(X_train)),
     'Test Set Accuracy': accuracy_score(y_test, svc.predict(X_test))
}

svm_model_accuracy = pd.DataFrame(model, index=[0])
```

Figure 4-24: Model Summary

CHAPTER 5: CONCLUSION AND FUTURE WORK

Client satisfaction is a mirror of how a customer feels about the company. It's the comparison between client expectations and the type of experience they actually receive from product and service.

Defining client satisfaction is not easy task. Defining client dissatisfaction is also not an easy task. On the surface, anyone can decide they know what it means to have a satisfied customer. However, analyzing little more into depth, it definitely would be difficult to say, what makes them satisfied or dissatisfied.

According to this research we can say clients were satisfied and clients were very happy on services what company and its projects were delivering at that moment. And also we can predict the client satisfaction and dissatisfaction using few columns. But with this data, absolutely we cannot recommend this is the exact way of analyzing customer satisfaction or dissatisfaction.

There are various perspectives of the business strategy, and client could be satisfied with one part and not another.

E.g.: Providing issue fix within SLA (Service Level Agreement) specified time period. We can say client satisfied. But company may need to go beyond that incident (Cannot satisfy because of the client satisfaction) and it is necessary to find the exact root cause of the fix. And also business should not allow to repeating the same issue twice. Figuring out those different areas will help to build a better all-around customer experience, satisfy existing customers, and establish more loyal customers that will praise the brand via word of mouth.

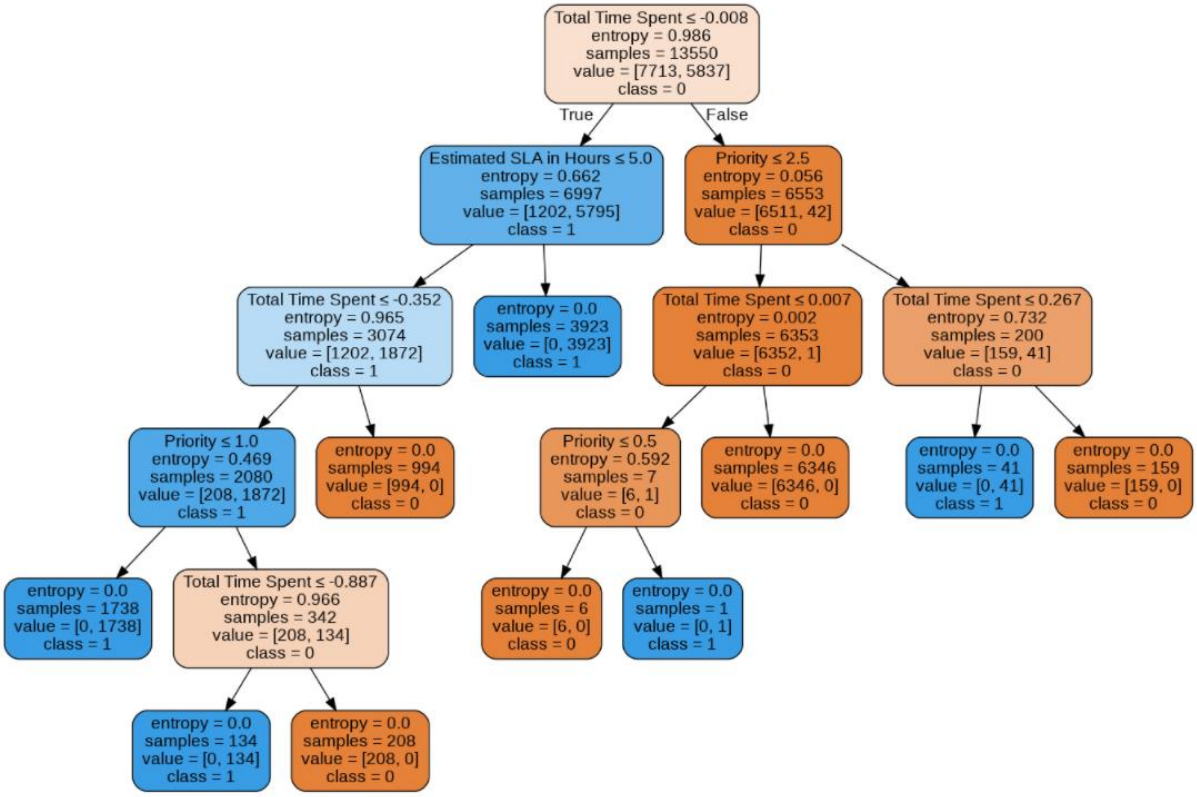
Once we understand the areas, the business wants to get feedback on, business may need to create a way to measure that sentiment. There are different types of surveys that the business could consider on client satisfaction; also dissatisfaction. Expect to extend the areas in future with the more evidence and valid data.

When analyze more about the customers, business voluntarily learn more about the business as well. It is time to take the time to get to know the clients deeper and deeper. When business does, it will be able to better serve clients and create a clearer path to success.

APPENDICES

Visualizing Decision Tree

```
[ ]  from sklearn.externals.six import StringIO
     from IPython.display import Image
     from sklearn.tree import export_graphviz
     import pydotplus
     dot_data = StringIO()
     export_graphviz(clf, out_file=dot_data,
                     filled=True, rounded=True,
                     special_characters=True, feature_names = features,class_names=['0','1'])
     graph = pydotplus.graph_from_dot_data(dot_data.getvalue())
     graph.write_png('decision_tree.png')
     Image(graph.create_png())
```




```
[ ]  model = {
          'Model': 'Decision Tree Classifier',
          'Training Set Accuracy': accuracy_score(y_train, clf.predict(X_train)),
          'Test Set Accuracy': accuracy_score(y_test, clf.predict(X_test))
     }

     clf_model_accuracy = pd.DataFrame(model, index=[0])
```

REFERENCES

[1] Khanka, S.S., 2005. How Being Good Is Good For Business? *Asia Pacific Business Review*, 1(2), pp.83-90.

[2] Anon., 2021. *88 Science and Tech Blogs & Publications That Hire Freelance Writers*. [online] Available at: <https://beafreelanceblogger.com/science-tech-blogs-freelance-writers/> [Accessed 22 September 2021].

[3] (Artificial intelligence for issue analytics: a machine learning powered .., 2021)

Anon., 2021. *Artificial intelligence for issue analytics: a machine learning powered .* [online] Available at: <https://blog.developer.atlassian.com/artificial-intelligence-for-issue-analytics-a-machine-learning-powered-jira-cloud-app/> [Accessed 22 September 2021].

[4] (Defect Management Dashboard | Agile Dashboards | Bold BI, 2021)

Anon., 2021. *Defect Management Dashboard | Agile Dashboards | Bold BI*. [online] Available at: <https://www.boldbi.com/dashboard-examples/agile/defect-management-dashboard> [Accessed 22 September 2021].

[5] (A deep learning model for estimating story points, 2019)

Anon., 2019. *A deep learning model for estimating story points*. [e-book] Available at: <https://arxiv.org/pdf/1609.00489> [Accessed 22 September 2021].

[6] (Learning for Life, 2014)

Anon., 2014. Learning for Life. *Great Quotes for Great Educators*, pp.59-82.

[7] (DEFECT PREVENTION BASED ON 5 DIMENSIONS OF DEFECT .., 2012)

Anon., 2012. *DEFECT PREVENTION BASED ON 5 DIMENSIONS OF DEFECT .* [e-book] Available at: <http://www.airccse.org/journal/ijsea/papers/3412ijsea07.pdf> [Accessed 22 September 2021].

[8] (What is Jira Software used for? | Atlassian, 2021)

Anon., 2021. *What is Jira Software used for? | Atlassian*. [online] Available at: <https://www.atlassian.com/software/jira/guides/use-cases/what-is-jira-used-for> [Accessed 22 September 2021].

[9] (What is Defect or bugs or faults in software testing?, 2021)

Anon., 2021. *What is Defect or bugs or faults in software testing?*. [online] Available at: <http://tryqa.com/what-is-defect-or-bugs-or-faults-in-software-testing/> [Accessed 22 September 2021].

[10] Anon., 2021. *Jira (software) - Wikipedia*. [online] Available at: <https://en.wikipedia.org/wiki/Jira_(software)> [Accessed 22 September 2021].

[11] Anon., 2021. *Understanding Confusion Matrix - Towards Data Science*. [online] Available at: <https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62> [Accessed 22 September 2021].