R.V. COLLEGE OF ENGINEERING
BENGALURU – 560059
(Autonomous Institution Affiliated to VTU, Belagavi)

"Automatic Ticket Generation for Complaints Raised on Social Media"

PROJECT REPORT
Submitted by

KARTHIK N A **1RV13CS069**

Under the Guidance of

Prof. Kowcika A
Assistant Professor,
Department of CSE, R.V.C.E.,
Bengaluru - 560059

Madhur Ahuja
Tech Lead, OneDirect
Coraza Technologies Pvt. Ltd.,
Bengaluru - 560029

In partial fulfillment for the award of degree of

Bachelor of Engineering
In
COMPUTER SCIENCE AND ENGINEERING
2016-2017

R.V. COLLEGE OF ENGINEERING, BENGALURU - 560059
(Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the major project work titled **"Automatic Ticket Generation for Complaints Raised on Social Media"** has been carried out by **Karthik N A (1RV13CS069)** who is bonafide student of R.V. College of Engineering, Bengaluru in partial fulfillment for the award of degree of **Bachelor of Engineering** in **Computer Science and Engineering** of the **Visvesvaraya Technological University, Belagavi** during the year **2016-2017**. It is certified that all corrections/suggestions indicated for the internal assessment have been incorporated in the report deposited in the departmental library. The major project report has been approved as it satisfies the academic requirements in respect of project work prescribed by the institution for the said degree.

Prof. Kowcika A Assistant Professor, Department of CSE, R.V.C.E., Bengaluru –59	**Dr. G. Shobha** Head of Department, Department of CSE, R.V.C.E., Bengaluru–59	**Dr. K N Subramanya** Principal, R.V.C.E., Bengaluru–59

External Viva

Name of the Examiners	**Signature with Date**
1. _____	_____
2. _____	_____

Date: 10-05-2017

TO WHOMSOEVER IT MAY CONCERN

This is to certify that **Mr. Karthik N A** pursuing his 8th semester in **Computer Science and Engineering** at **R.V. College of Engineering, Bengaluru** has been interning with us at **Coraza Technologies Pvt. Ltd., Bengaluru** and successfully completed the project titled **"Automatic Ticket Generation for Complaints Raised on Social Media"** from **09-01-2017** to **10-05-2017**.

Authorised Signatory

Annett Macgowan	**Madhur Ahuja**
Associate Director, Human Resources,	**Tech Lead, OneDirect,**
Coraza Technologies Pvt. Ltd.,	**Coraza Technologies Pvt. Ltd.,**
5th Floor, Tower D, IBC Knowledge Park,	**5th Floor, Tower D, IBC Knowledge Park,**
Bannerghatta Road, Bengaluru-560029	**Bannerghatta Road, Bengaluru-560029**

R.V. COLLEGE OF ENGINEERING, BENGALURU - 560059
(Autonomous Institution Affiliated to VTU)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

I, **Karthik N A (1RV13CS069)** the student of Eighth Semester B.E., Computer Science and Engineering, R.V. College of Engineering, Bengaluru hereby declare that the major project titled **"Automatic Ticket Generation for Complaints Raised on Social Media"** has been carried out by us and submitted in partial fulfillment for the award of degree of **Bachelor of Engineering** in **Computer Science and Engineering** of **Visvesvaraya Technological University, Belagavi** during the academic year **2016 - 2017**. I declare that matter embodied in this report has not been submitted to any other university or institution for the award of any other degree or diploma.

Place: Bengaluru **Signature**
Date:

Karthik N A

ACKNOWLEDGEMENT

Any achievement, be it scholastic or otherwise does not depend solely on the individual efforts but on the guidance, encouragement and cooperation of intellectuals, elders and friends. A number of personalities, in their own capacities have helped me in carrying out this project work. I would like to take this opportunity to thank them all.

First and foremost I would like to thank **Dr. K N Subramanya, Principal,** R.V.C.E, Bengaluru, for his moral support towards completing my project work.

I deeply express my sincere gratitude to my guide **Prof. Kowcika A, Assistant Professor,** Department of CSE, R.V.C.E, Bengaluru, for her able guidance, regular source of encouragement and assistance throughout this project

I would like to thank **Dr. G Shobha, Head of Department**, Computer Science & Engineering, R.V.C.E, Bengaluru, for her valuable suggestions and expert advice.

I also extend my cordial thank to **Coraza Technologies Pvt. Ltd.,** Bengaluru for providing me an opportunity to carry out the internship in its organization. I also would like to thank my tech lead and all team members for their support and guidance.

I thank my Parents, and all the Faculty members of Department of Computer Science & Engineering for their constant support and encouragement.

Last, but not the least, I would like to thank my peers and friends who provided me with valuable suggestions to improve my project.

Abstract

The project is based on Social Customer Relationship Management(SCRM) and is one of the most recent applications of social media platforms like Twitter and Facebook. Companies understand its value and are spending large sums of money to have a positive social presence. The first response time for most companies in social media is under 100 seconds, that is staggeringly fast and the customer complaints are addressed almost instantly. SCRM is highly valued in the market currently and its requirement is only growing every passing day. According to Gartner, the market for SCRM had crossed a billion dollars across the world as early as 2012. Also, 30% of all major corporate and startups have reported having adopted some form of SCRM as per a survey conducted by Gartner in 2015.

Data from the social media touch points is fed to the workflow engine and at the same time stored in the database. The first step in the workflow engine is the Natural Language Processing(NLP) engine that is responsible for the sentiment analysis and parses the incoming data. If the given data indicates a negative sentiment, an appropriate ticket is automatically raised capturing essential details corresponding to the issue and is done by the Ticketing Engine. The ticket is automatically assigned to one of the available customer care representatives. The Rules and Priority Engine is responsible for determining if the complaint is common, repeated and many such criteria. The priority of the ticket raised is also determined by social presence as well as position in the company they work for. The last is the Analytics and Reporting engine that provides insight into to real-time trends, ticket analytics, response metrics and agent performance shown in the form of graphs and pie charts.

The proposed system is able to capture a tweet and raise a ticket under a minute. This provides the companies with a mechanism to almost immediately respond to customer complaints. Also, the project is expected to provide a great user experience to interact with their customers and at the same time document all their interactions in a proper manner. The sentiment engine used has a peak classification accuracy of 85%. From the experimental results it was found that the number of users using social media has been growing steadily. Most of the complaints raised on social media are of low priority and are resolved within an hour from the time of complaint registration.

Table of Contents

List of Figures

List of Tables

Glossary

API	:	Application Program Interface
CCR	:	Customer Care Representative
CRM	:	Customer Relationship Management
CSS	:	Cascading Style Sheets
DDR	:	Double Data Rate
DFD	:	Data Flow Diagram
DM	:	Direct Message
HTML	:	Hypertext Markup Language
DOM	:	Document Object Model
GUI	:	Graphical User Interface
IDE	:	Integrated Development Environment
JDK	:	Java Development Kit
JSON	:	JavaScript Object Notation
NLP	:	Natural Language Processing
NLTK	:	Natural Language Toolkit
SCRM	:	Social Customer Relationship Management
SMA	:	Social Media Analytics
SRS	:	Software Requirements Specification
SVM	:	Support Vector Machine
UI	:	User Interface
UX	:	User Experience
WIFE	:	Work Integrated Flow Engine

Chapter 1

INTRODUCTION

SCRM combines and integrates the positives of both traditional CRM and social media. SCRM is the next big thing that will modernise the interaction of companies and their clientele. With technology advancements, social media is not just a tool to interact with your near and dear but a powerful tool to openly voice out your opinions. It provides a possibility to reach out a large audience from the mere touch of your smart phones.

Social media has countless opportunities, it would be foolish for companies to not take advantage of it. Social platforms bring companies closer to their clients and also gives the companies a chance to boost revenue, cost reduction and improve efficiencies. . Businesses are building virtual communities out of social media platforms and thrive on popularity to grow their business.

This research study is aimed at understanding the difficulties faced by the CRM teams in companies specifically with issues and complaints raised on social media platforms as well as provide efficient solutions to tackle the problems.

1.1 State of the Art Developments

State of art refers to most recent and advanced developments in that field of research or domain in a particular time period. It highlights the methodologies and practices employed to achieve the task in that time period.

Zeynab Soltani and Nima Jafari Navimipour [1] performed a study to understand the potentialities of SCRM .They looked at 5 categories knowledge data mining, Computerised CRM , data cleaning and socialising in the web . In each category, a couple of studies were performed to roughly identify the key factors of CRM. Data mining and data acquisition techniques to extract data from social media platforms have been discussed.

José Martins et al [2] performed a systematic literature survey on social networks sites mainly covering on methods of social interaction using social media sites. The authors

concentrated on the type of interaction companies should have on social media with their clients and adoption of latest technology for SCRM. The authors proposed technology adoption models and their impact on both individual and firm level.

Fatima Zohra et al [3] proposed a social intelligence framework that could extract and consolidate reviews and opinions raised on social media to aid companies to understand customer opinions toward a particular product or service. Sentiment analysis was done on tweets filtered from Twitter using Twitter4J. MapReduce architecture of the system implemented using Hadoop has also been discussed.

Marcel Rosenberger [4] proposed a systematic approach to separate the various components involved in SCRM. The paper distinguishes the concerns of SCRM using architectural perspectives and aimed at building a better understanding of SCRM. A five stage development and research methodology have been proposed to tackle the problem of SCRM.

Nyoman Karna, Iping Supriana and Ulfa Maulidevi [5] proposed a method to do SCRM using web mining on social media, blogs and consumer complaint platforms. The paper proposed the method to analyze the relationships of people posting on these platforms and the people who respond to it other than the companies in the discussion. The collected knowledge are stored in form of semantic maps later used for targeted marketing.

Abid Hussain and Ravi Vatrapu [6] proposed a technique that has been is engineered to perform analysis in a such a way that it could identify the company the people are discussing about and their views about the company in their chats on social media.

Surya Nepal, Athman Bouguettaya and Cecile Paris [7] discussed the impact of social platforms and cloud computing on the way individuals communicate with each other and companies over social media from their comfort of their home. A method was proposed to handle the large information flows generations as a result of these interactions online. The case study covered in the paper explained how in the case of an emergency, social media played a key role to spread information between survivors and rest of the world.

Marcel Rosenberger et al [8] proposed an interaction model for companies with their customers on social media platform. A workflow has been indicated from start to end on how to respond and communicate with customers over social media to systematically resolve the issue in hand. Middleware solutions have been proposed that allows the CSR teams interact with the customers indirectly and at the same time store and maintain the short interactions with the customers for future reference.

Priyanga Gunarathne, Huaxia Rui and Avi Seidmann [9] performed analysis on customer interaction of airlines in the USA over social media between June 2013 and August 2014. It has been found that airlines compared to other industries pay more attention to their image in social media. Also, customers who have a large number of followers and tweets are generally attended with higher priority compared to the rest of lower first response time and faster resolution time.

Ifeoma Adaji and Julita Vassileva [10] predicted the churn of expert respondents in Stack Overflow. Experts were identified based on the InDegree of the respondents and the value of the incentives earned by these experts from the questions they have answered in the past. Four different data mining techniques were evaluated to predict the user churn. The results were expressed in the form of ratios and graphs for better understanding.

Nargiza Bekmamedova and Graeme Shanks [12] explained the value of social media analytics. Bankco was taken up as a case study that had previously used SMA as a part of their critical marketing campaign and came out with flying colors. Bankco cleverly leveraged specific social media channels to target their clients and it was able stand out among its competitors. Bankco was provided with important insights into customer satisfaction [11], requirements and knowledge of their products.

Hyesun Jeon and Hyung Jun Ahn [13] studied factors that affected the reaction of consumers to the posts that companies make on their brand pages. Brand pages of two local mobile gaming companies were analyzed. Each post on the page was evaluated for a number of likes, informativeness, reshares, structure and type of reward. The paper concluded that videos had the most impact on the audience. Also, contests held from the brand pages overwhelming response from the public.

Babak Abedin and Hamed Jafarzadeh [14] proposed the CSM for effective CRM on Facebook for companies. The model was designed keeping in mind seven critical factors necessary for a brand to be successful with SCRM. The model adopted dynamic SNS plans, integrated marketing with SCRM, encourage user participation, reward the customers and shared the the truth with the customers.

David Alfred Ostrowski [15] evaluated Latent Dirichlet Allocation used for the generative probabilistic model. The model was used for a collection of discrete data. The technique was able to identify noteworthy discussion topics from a dataset of filtered Twitter messages. It also had the power to identify sub-topics and classify them accordingly when dealing with large datasets. New trends on social media could be identified by these techniques.

Marcel Rosenberger, Christiane Lehrer and Reinhard Jung [17] discussed the impact on methods of linking user activities in social media into government applications. The authors suggest these integrations must not rely on 3rd party services but have an independent API stack to perform the various operations. The authors proposed an interesting conceptual model to categorize the user activities in 40 different categories.

A. Marzak et al [18] first discussed the existing architectures for CRM and next propose a new architecture that is a mix of all the positives in the different architectures along with the integration of the latest technologies.The proposed architecture has the capability to exploit new data sets and adapt its model to suit the changes in the dataset.

The authors [19] of this incredible paper proposed a methodology to examine the behaviour of customers towards a particular fashion company on Instagram. Complex clustering rules were applied that was followed by application of association rules to get the results. These results could act as a starting point for further marketing strategies.

Jitendra Ajmera et al [21] were working in IBM during the time when the paper was published. The authors present their experiences and learnings from building the system that mines conversations on social media platforms to identify and prioritize those posts and messages that are relevant to enterprises. The system presented in this work aims to empower an agent or a CCR [20] in an enterprise to monitor, track and respond to customer

communication at the same time encouraging community participation. The authors proposed the SCION Content Analytics model.

Stephen Wan et al [22] propose a model called VIZIE that is used for social media data aggregation and mining for internet scale CRM. VIZIE is a social media monitoring system designed to help media monitors to distill the perception of potential and actual customers of the value of existing products and services, identify and correct customer misinformation, discover how to better meeting customer expectations, and (if needed) communicate directly with the customers by using the customers' preferred social media forum. A detailed overview of the design used in specified in the paper.

Kevin J. Trainor et al [23] examined how social media technology usage and customer-centric management systems contribute to a firm-level capability of SCRM. The study focuses on the conceptualization and measurement of social CRM capability.

Jyotirmoyee Bhattacharjya and Adrian B. Ellison [24] examined the CRM activities of business to business companies and service providers like Shopify to identify opportunities for business resilience and improving brand perception. The author draws a comparison between B2B and B2C on the usage of social media like Twitter for business improvement and growth. The authors concluded that the activities of B2B companies on social media is a one-way communication with little or no response from the client companies. It will take a significant amount of time for B2B companies to catch up with B2C companies on social media platforms.

Ulrike Baumöl, Linda Hollebeek and Reinhard Jung [25] highlighted the dynamics of customer interaction on social media platforms. The literature review covers various aspects of CRM. The evolution from traditional CRM [16] to modern SCRM is very well highlighted. The paper also highlights the shift from company oriented CRM practices to customer-centric practices.

Siaw Ling Lo, David Cornforth and Raymond Chiong [26] proposed a model to identify the high-value social audience for targeted marketing. The authors proposed High-Value Social Audience (HVSA) index to identify these people. HVSA index enables a company or organization to devise their marketing and engagement plan according to

available resources, so that a high-value social audience can potentially be transformed to customers, and hence improve the return on investment. A combination of LDA and SVM is used to determine HVSA. Text based mining has been used to understand the interests of HVSA and people with similar interests are to HVSA are targeted with products and services that may be used by HVSA.

Yoav Achiam, Inbal Yahav and David G. Schwartz [27] simulated Companies' ego networks on Twitter, meaning the companies' numbers and type of followers. Evident from the collected data, the authors were able to show that followers' distribution is neither scale-free nor random, thus common network simulations cannot be used to mimic observed data. To tackle this problem, the authors have proposed novel rate equations model to capture the complex dynamics of these ego networks. The experimental results are in good agreement with the actual available data.

S. S. Askool and K. Nakata [28] reported on a scoping study that explored the current situation of CRM adoption in the banking industry in Saudi Arabia. It aimed to identify the factors that influenced the use of SCRM. An enhancement to the existing Technology Acceptance Model(TAM) was proposed in the paper along with a survey on its effects in Saudi Arabia.

Sanaa Askool and Keiichi Nakata [29] identified the factors that influence customers in the adoption of SCRM, which is useful for CRM practitioners to improve existing and future relationships with their customers. A model for exploring and predicting the acceptance of SCRM was been presented in this paper. Based on the scoping study about the current situation of CRM adoption in Saudi banks, traditional CRM, social networking and Web 2.0 literature, a range of factors that influence IS acceptance attitude and behavior were identified and integrated with TAM.

Chunjing Xiao et al [30] proposed a fuzzy triangular number based method to measure the influence of a Youtuber and method is free from any form of graph representation.Technique for Order Preference by Similarity to an Ideal Solution (TOPSIS) method is used to synthesize multiple triangular fuzzy numbers and rank users. The experiments conducted by authors based on YouTube data showed that, when only

considering view counts, our results are in good agreement with other popular measures such as h-index. There was a significant deviation from predicted results for parameters other than view counts.

Edward C. Malthouse et al [31] examined the need for adaptation of social media in CRM. The pitfalls and opportunities that companies have through social CRM were discussed.organization's lack of control over message diffusion, big and unstructured data sets, privacy, data security, the shortage of qualified manpower, etc were some of the key pitfalls.

Julia Heidemann et al [32] explained the impact of social media on the world. The article explores the possible means by which it could be exploited by both companies and individuals for their benefits. The article covers the user motives to participate is social media, functionality and context of social media and the risks in using social media.

Seyedreza Mousavi and Haluk Demirkan [33] measured the impact of social media CRM tools used to interact with social platforms like Facebook and Twitter. 65 IT professionals working on these products and the similarities and differences between their architectures were discussed. Customer engagement was a key feature in all the models. The authors at the end proposed a framework that covered the positives from all the architectures.

Imre Petkovic [34] discussed the implications of moving the CRM into the cloud. The author describes the characteristics necessary to move the CRM to cloud and explained how cloud computing can be exploited for the purpose of CRM. There is a benchmark proposed as a part of the paper to measure the performance of SCRM in the cloud.

Eun Go and Kyung Han You [35] examined the usage of social media of 317 organizations and how they leveraged it for the purpose of CRM. the authors were able to categorize them into 6 categories in accordance with the social media applications predominantly used by each organization. The groups were created based on the number of social platforms the companies were active on and also the type of social media platforms (blogs, vlogs, etc.) they are using. Also, the authors found that organizations preferred the use of two-way communication social media platforms to have active user participation.

Dushyant Kumar Sharma [36] highlights the foundational components of SCRM and explained the importance of incorporating it as a root level function to improve customer relationships. The author lists out the nine possible types of customer interaction through social media with companies with its pros and cons.

David Alfred Ostrowski [37] explained the procedure that must be used for feature set selection for Twitter messages. The author started by explaining the complexity of the available features and procedure to correctly select the features necessary for the NLP operation. Both NLP and information theory practices were examined for feature set selection and their results were compared.

Hanno Zwikstra [38] proposed a system to reuse social data gathered from one platform and use it as well as map it to the same user in another platform. This type of mapping could be used to enhance the SCRM and improve the user experience. The data from LinkedInFinder and fed into the Microsoft Dynamics CRM system.The POC has been successful in detecting second degree connections among the users of the same company. The survey proved that such a system would be useful if it is designed properly and meets the user requirement.

Adam Czyszczoń and Aleksander Zgrzywa [39] attempt the address the problem of customer segmentation in SCRM using Kohonen networks.The proposed model takes into account the traditional loyalty-profitability link model as well as linking it with new social media components. The proposed model was implemented and results obtained were very close to the results obtained by K- means method. The implementation provides an analysis tool with data visualization capabilities that could be used by marketing managers for customer segmentation.

Dario Liberona, Manuel Ruiz and Darcy Fuenzalida's [40] research are about knowledge management information technology tools and social monitoring tools. Chilean companies were interviewed through a survey to understand the method in which the various organizations utilise and store the data. The authors tried to access the value of customer data to each organization. It was concluded that the Chilean companies are not yet exploiting SCRM and majority of customer interaction is through traditional CRM.

David Alfred Ostrowski [41] have developed a method of filtering to characterize trends of consumer behavior in relationship to specific products using the Twitter messaging system. The process identifies semantics at 3 consecutive levels to determine the demand signal. At stage one, the determination of ground truth keywords followed by word-level and category-level empirical keywords. Next semantic classification of into categories like negation, humor and emotion are considered. Additional filtering is done in the last stage to understand consumer behavior in a better way. The author used this method to identify customers who might be interested in the purchase of a vehicle based on their behavior on social media platforms.

Minbo Li [42] built a social collaboration platform (www.yonyou.com) has been developed in collaboration with Ufida Software Company which provides enterprise profile, microblogging, grouping features, project and product management, document collaboration and instant messaging service. The portal provided an enterprise-grade social network for entrepreneurs, companies, angel investors and venture capitalists interact with each other in a secure and safe platform. It provides a great ecosystem for growth and sharing new ideas. The platform encourages two-way communications between companies.

Meyliana, Budiardjo and Eko K [43] proposed a model by merging Architecture for Integrated Information System (ARIS) and Zachman framework for the construction of an enterprise grade SCRM. The paper started with the description and comparison of the two models. The useful deliverables are picked from both the architectures to build a hybrid that is a baseline for the creation of an enterprise grade architecture for SCRM.

Neil Woodcock et al [44] discussed the ethical issues wrt linking social data with CRM data. The encroachment of privacy and security that must be implemented to safeguard the vast amounts of information. The possible misuse of information has also been discussed. The standard 8 stage SCRM model right from ‘Listen and Learn’ all the way up to ‘Measure and Evaluate’ has been discussed in detail. Identifying high-value customers is one of the biggest advantages of SCRM.

Niels Buus Lassen et al[45] questioned the recent development in the field of SCRM. The author remarks that SCRM is just old wine in new bottle. The author lays stress on the

fact that there is nothing innovative from shifting traditional CRM to SCRM, only the method of interaction with the customers has altered. The author urges the companies to rethink their social media strategy and concentrate on innovation rather than acquiring a fan following on social media platforms.

Sergio Orenga-Roglá and Ricardo Chalmeta [46] proposed a novel methodology that helps companies to implement Social CRM considering various aspects like social consumer strategy and SCRM performance measurement to name a few. The methodology was adopted by a multinational company and had positive results. SCRM-IRIS methodology provides a roadmap for organizations to systematically obtain value from the available data.

Lawrence Ang [47] in the paper suggested that SCRM is a wrong term and it must be replaced with community relationship management (that is, CoRM) since companies are indirectly interacting with a large audience with every single blog, post, comment or tweet. The method employed companies to connect, converse, create and collaborate reflects the image of the company itself. The author urges the companies not to over-exploit social media only to improve profits but rather concentrate on holistic improvement in the relationship with their clientele.

Olaf Acker et al [48] wrote the paper with the following goals to develop the credibility needed in the social web and to use this web to boost sales. The shift to SCRM requires significant effort and capital. All the possible scenarios that are explored by considering case studies on Dell, Best Buy, etc. The author suggests that companies must employ MASTER(monitor, assess and analyze, strategize and structure, test, embed, review) methodology while moving from traditional CRM to SCRM. The author describes the changes that need to be made with the strategy after completing the initial transition and reached a rather matured stage with a fair and stable plan in place wrt to SCRM.

Nicolas Wikuhn et al [49] examined contemporary research and juxtaposes it to current business needs within a holistic SCRM performance dimension framework. The paper provided new and validated definitions of infrastructure and process components related to SCRM and developed propositions regarding customer-centric resources and capabilities. It

further revealed research gaps within the literature regarding SCRM performance measurement and provides suggestions for further research.

Tobias Lehmkuhl and Reinhard Jung [50] provided a state the art developments in the field of SCRM. The paper revealed the concurrent opinions, diverging perceptions and future directions for research along the dimensions SCRM definitions. The authors concluded that SCRM is a novel method that requires immense transformational efforts among the various organizational parts. CRM life cycles, processes and performance metrics for the CCR must be altered to align with SCRM.

1.2 Challenges

The challenges for the Social CRM are quite complicated and multifaceted in nature. The major challenges are as follows:

- The fundamental challenge is being able to identify customer complaints on the internet. It is like finding a needle in a haystack. It is no easy task to capture data that is specific to our needs. This challenge mostly concentrates on how to get relevant data and where to find it?

- Once the data is obtained, the next logical step is obviously to process this data, but the question that arises is how you are going to process the data and what is expected out of processing?

- The assessment of the impact of a complaint posted on a social media platform. The metrics largely vary from Number of followers of a user, social activeness, Number of shares or retweets, etc. There is no definite way to assess the impact of a complaint or dissatisfaction, but one thing is for certain that any negative comments on social media for companies is bad PR.

- Resolution of a complaint generally requires sharing some confidential information of the user to cross verify the problem. Social media platforms are no place to share such information since it is open to all. There are some serious security concerns to be taken care of.

- There are no clear-cut government policies to handle social CRM. Companies act on their own. There is a high chance that private and confidential data may be mishandled and misused. Some governments prevent companies in engaging through social CRM.

1.3 Motivation

Traditional CRM largely is customer data bucket. Any information related to the customer is collected and dumped in this bucket with the intention of using for some or the other advantage at a later stage. In traditional CRM, the business determines the mode of communication but Social CRM is an all new ball game where the customer decides the communication channel and not the business.

Customers' preferred communication channels are evolving and are not few in number. Customers spend a significant amount of time online and they control of their experience with the brands.

Everything social is public. Each and everyone's' voice gets heard. Every opinion counts. In a world that is hooked to the internet, the brand value of a company is largely determined their social presence and the ability to act fast to changing trends on the internet. At the end, it all boils down to keeping the customer happy and how fast a company is able to respond to a complaint in any social platform. Social media platforms like Twitter, Facebook, LinkedIn, etc are open platforms where the public openly share their views about anything and everything. The same applies to the brands as well. If a customer has a positive thing to say on the internet, it is a plus point for the company and on the downside, a negative comment destroys the image of a company. With the advent of social CRM, the metrics of success has changed greatly. Word-of-mouth referrals is a thing of past and a simple Tweet or a post on Facebook does wonders. Imagine if someone else retweets the same, it could send a chain of events to unfold. Not only does it give free publicity but also increases the popularity of the company. Companies are moving to social platforms to captivate this growing audience. Social media empowers the service over the traditional transaction-oriented customer service. To stay ahead of the game, companies meticulously scour the internet for any customer complaints or dissatisfactions. This is usually done by tools that do the

searching for them. The search is based a predefined set of rules that is able to capture only necessary data. Once the relevant data is captured and a ticket is raised, the data is passed onto CCR to handle the problem with a personalised touch.

1.4 Problem Statement

Social CRM is not merely being responsive or creating the appearance of immediately responding, but rather it is important to know who exactly the customer in the discussion is is. Companies have amassed a large number of followers like Twitter and Facebook but the company has no clue who its actual customers are and what is the reason they are being followed. Determining the customer base is a data-driven process with little or no automation involved and this is also is output oriented and there is no development of customer relationship.

The challenges are obvious – privacy concerns both from the brand and customer perspectives is of top priority. Social networks would understandably be reluctant to permit brands to suck out such information in order to reconcile it in CRM – though the companies are willing to pay a hefty price for such a luxury (and prove to be a valuable new revenue stream for social networks).

Add in the complexity of a financial services company, and perhaps regulatory hurdles and we've got a whole new set of problems in the way waiting to be solved.

1.5 Objectives

Objectives are simple statements that describe the specific deliverables that the project is intended to deliver. The key objectives of the project are listed below:

- Listen, analyze, collect feedback and respond to customer grievances across multiple digital touch points.
- Automatically raise tickets for issues raised on social media.
- Faster resolution of customer grievances.

- Minimize the interactions the manual interaction of the CSR personnel on multiple digital touch points.

It is expected that a tool will be created for Automatic Ticket generation for complaints raised on social media platforms that satisfy the all the objectives listed above.

1.6 Scope

One of the key aspects of project planning is the project scope that involves the determination and documenting the list of goals, deliverables, deadlines and estimated cost. In other words, it is what needs to be achieved and the work that must be done to deliver a project.

A tool will be created that is able to capture relevant data from social media touch points and able to generate tickets from them after processing it. The available services will be utilized to the maximum extent so that the project is primarily focussed on processing the captured data and raising tickets.

1.7 Methodology

Research methodology is a way to systematically solve the research problem which may be understood as a science of studying how research is done scientifically and methodically. It is vital and extremely important for the researcher to know not only the research methods/techniques but also the detailed methodology. Researchers not only need to be aware of the existing techniques of calculation but also have the ability to choose the technique that is most suitable for the current problem. Researchers must be able to justify their decision. All this means that it is necessary for the researcher to design the methodology that best suits the problem as the same may differ from problem to problem.

Figure 1.1 shows the methodology of the project. Data from the social media touch points is fed to the workflow engine and at the same time stored in the database. The first step in the workflow engine is the NLP engine that is responsible for the sentiment analysis and parses the incoming data. If a given data indicates a negative sentiment, an appropriate ticket is automatically raised capturing essential details corresponding to the issue and is done by the Ticketing Engine. The ticket is automatically assigned to one of the available customer

care representatives. The Rules and Priority Engine is responsible for determining if the complaint is common, repeated and many such criteria. The priority of the ticket raised is also determined by social presence as well as position in the company they work for. The last is the Analytics and Reporting engine that provides insight into to real-time trends, ticket analytics, response metrics and agent performance shown in the form of graphs and pie charts.

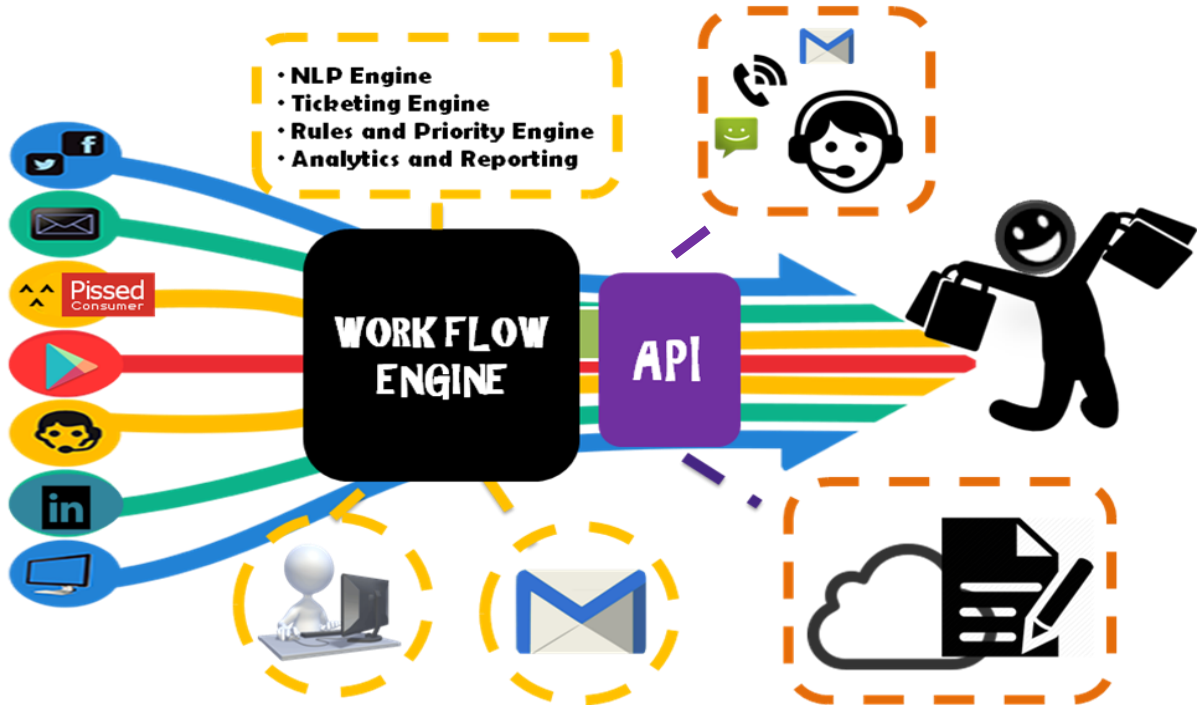

Figure 1.1: Methodology

1.7 Organisation of The Report

The main body of the report is preceded by a table of contents including a list of figures, tables, and glossary, followed by the units used in the report, followed by appendices, which contains screenshots.

The report also contains details of development and design criterion used during the implementation of the project and how these were implemented. The body is divided into the following parts:

Chapter 2 describes the requirement specifications of the project. This chapter explores the functional requirements, the hardware, software requirements and the constraints imposed on them to get a better understanding of the product that needs to be developed and its environment.

Chapter 3 describes the high-level design of the system. This chapter describes the design considerations, architectural strategies (like programming language and UI), system architecture, and data flow diagrams.

Chapter 4 explains the detailed design of the system. It explains dependencies of the components, the structure of the components, implementation blocks, etc.

Chapter 5 describes the implementation in detail. It is a detailed description of the naming conventions, the organization of project resources, programming challenges, etc.

Chapter 6 explains software testing. Testing comprises of unit testing, where the components are tested individually; integration testing, where the components are tested by integrating each with the other, and system testing, where the entire system as a whole is tested.

Chapter 7 describes experimental analysis and results. Tests described in chapter 5 are run on sample data sets and results are recorded. This chapter is a log of such results, and inferences were drawn.

Chapter 8 concludes this project work by outlining the limitations of the project. Some future enhancements for the project are also indicated.

1.8 Summary

This chapter was intended to provide a peek into the project in hand and it did just that. A clear explanation of the task in hand was given down through the well described and detailed problem statement, objectives and methodology. The state of the art showed the latest developments in the field. It gave us clear context on what are the existing possibilities and the ones that are most suitable and match the project requirements.

Chapter 2

SOFTWARE REQUIREMENTS SPECIFICATION OF AUTOMATIC TICKET GENERATION FOR COMPLAINTS RAISED ON SOCIAL MEDIA

Software Requirements Specification (SRS) is a document or set of documents that describes the features and the intended behaviour of a software application in detail. It is an amalgamation of multiple modules that aims to define the intended functionality required by the customer to satisfy their requirements from the software. The specification highlights the core business processes that must be supported by the software, the assumptions that have been made and the key performance parameters that need to be met by the system. The specified requirements must be testable, quantifiable and must be defined in a manner that it leaves no room for ambiguity.

2.1 Overall Description

This section deals with general constraints and requirements of the system. The requirements, constraints, dependencies and performance metrics are well documented in the following sections. The project has been explained from the perspective of the system and also with respect to the characteristics of the user.

The project takes multiple parameters into account while processing the incoming data. Care is specifically taken to handle every type of system failure and fail-safes have been implemented to prevent the breakdown of the entire system in case of failure of a single module. The use of microservice architecture makes it possible to plug and play any microservice without hampering the performance of the other modules.

The system requirements in the project have been developed keeping in mind of long term usage and growth. Non-functional requirements like reliability, availability, and scalability have been seriously taken and a significant amount of time was spent during the design and development process to satisfy these requirements.

This project was developed keeping in mind that humans are not perfect and bound to do mistakes. Error correction algorithms have been incorporated into the sentiment engine so as to accurately predict the emotion of the incoming data. The sentiment classification model has been tweaked and trained accordingly to incorporate and identify the slang used in local Indian languages.

2.1.1 System Perspectives

System perspective is the process of understanding how those things which may be regarded as systems influence one another within a complete entity, or larger system. It is an approach to problem-solving that attempts to balance holistic thinking and reductionistic thinking

The system has a very modular design. The microservice architecture paradigm was chosen as it best matches the requirements of the project. All the components work asynchronously and support parallel processing. The system can be broken down into three may stage on the activity they perform. First, the preprocessing stage. It primarily comprises of the GNIP service, Redis and RabbitMQ service. In the preprocessing stage, it is ensured that the entire data has been received before the actual processing and unwanted data is filtered out. The incoming data is the form of JSON and converted to Java objects for the convenience of further processing. Second, the processing stage is the Twitter data Service comprises the workflow engine that comprises of the sentiment engine, rules and priority engine, ticketing engine and lastly analytics and reporting engine. Each engine functions individually and not interdependent on each other and ensures faster processing. Third, the post processing stage and commonly called the wife service is responsible for providing an interface that supports a mechanism to continuously interact with the user by the CCR to resolve the engine. The wife service is reliant on the previous stage for details of the ticket created and other user information necessary before dialog with the customer could be started.

2.1.2 Constraints

Project constraints are hurdles that restrict or dictates the actions of the project team. The three main types of constraints are time, resources, and scope, and every project is

hindered by it. A project's scope involves specific goals, deliverables and tasks that define the boundaries or limitations of the project. The schedule specifies the timeline according to which those components are intended to be delivered, including the final deadline for completion. Cost ie often referred as resources involve the financial limitation or upper limit of resources input to the project and also the overall limit for the total amount that can be spent.

Capturing tweets that corresponding to a particular brand or company is one of the biggest challenges. The tweets are captured using GNIP service. GNIP requires that we specify the rules to capture and stream that match our requirement. The rules that are specified are for the major case and it is not possible to cover all the cases. Every now and then few relevant tweets are missed out.

Rate limits have been imposed by Twitter to prevent a denial of service attack. Rate limiting of the API is primarily on a per-user basis — or more accurately described, per user access token. Consider a method makes 15 requests per minute and returns an error code for the subsequent requests, then each access token allows 15 requests per minute per user.

The reporting engine is not real time and lags by almost 10 minutes. This lag is present as sufficient time must be provided for the computer to calculate the various metrics that require significant computation power.

Tweets deleted are not automatically updated in the databases, their status is updated only when a user tries to communicate with the tweet. It is not possible to check each tweet captured is present or deleted as it is inefficient and wastes a lot of data.

Many times tweets match one of the rules specified and are captured by the system. The sentiment engine is unable to categorise its sentiment and is stored as a neutral sentiment tweet for the CCR to decide if a ticket has to be raised or not. There is another possibility that sometimes tweets that are not intended to be captured are also shown since they accidently match the rules.

The Internet is unreliable and there is a high possibility that few packets or data streamed by GNIP may be due to poor network connection.

2.1.3 Assumptions and Dependencies

An assumption is a belief of what one assumes to be true in the future. Assumptions are based on one's knowledge, experience or the information available on hand. Assumptions are anticipated events that are expected to happen during the project's lifecycle. Assumptions may not be true always and may jeopardise the project and also add to the existing risks of the project.

A tweet is captured only if it matches the specified rules in the GNIP console. Tweets that relevant but do not match the specified rules won't be captured. If multiple tweets are being missed, it is an indication that changes have to be incorporated into the rules to capture the missed tweets.

The assumption is that a Twitter user tweets about a company or brand only if they want to tag them in one of their tweets to showcase their gratitude or raise a complaint on a social media platform.

The system is not perfect and human intervention is an absolute necessity. The specified sentiment and other parameters are just guided for the CCR. It is the CCR responsibility to verify the received information.

Dependencies are the relationships of the preceding tasks to the succeeding tasks. Tasks can have multiple preceding tasks as well as multiple succeeding tasks. It is the duty of the programmer to identify these dependencies and schedule the tasks accordingly.

Though the entire system runs in an asynchronous mode and supports parallelism, the order of execution is the preprocessing stage, the Twitter data service and lastly the wife service.

The system relies on GNIP to provide the tweets that match the rules. If there is a delay or some issue with the GNIP service, the entire system comes to a standstill. Without any data to process, the system remains idle.

2.2 Specific Requirements

In the previous section, the general requirements that would be necessary for any similar system has been described. In this section, we concentrate on requirements that are necessary for this project alone.

2.2.1 Functional Requirements

The functional requirements of the system are:

- The system must be able to capture and process data from Twitter independent of human intervention.
- All the databases must be in sync and capture and update its records in real time.
- The reporting and analytics are expected to be accurate up to few decimal points.
- The expectation is that the CCR has to never log onto Twitter or any other social media site for any operation. The user interface must be capable of replicating every single feature on the social media platforms within the UI.

2.2.2 Non-Functional Requirements

These are the requirements that specify criteria that can be used to judge the operation of the system rather than specific behaviour and are not directly concerned with the specific functions delivered by the system. The important non-functional requirements with respect to this project are :

- Reliability: The system should be very accurate in its operation corresponding to the data.
- Ease of Use: Since the system has very less interaction with the user, the system is very easy to use. The user interface is very simple and aesthetic.
- Availability: The application shall be up and running during the entire session of user operation. The system is expected to have only scheduled downtimes and does not intend to have downtimes for any kind of system failure.

- Supportability: The system is to be developed in a generic manner. The system takes data in a JavaScript Object Notation format and also provides output in the same JavaScript Object Notation format. This data is stored in the database and the UI designed is supported by and Java enabled web browser.

- Scalability: The system is built using microservices. At given time if the system does not have enough instances of a particular resource, only the resource instances are increased. The system is both horizontally as well as vertically scalable. The database, UI and other major components are stored on various servers across the cloud and support distributed computing.

2.2.3 Performance Requirements

The performance characteristics of the system are outlined in this section. The resources that are utilized by the product are explained in detail. The system is expected to raise a new ticket with a maximum latency of about 10 seconds from the time it is tweeted. This includes all the overheads that may come along with the API call, functions in the backend, the retrieval of results and their display time. The network overheads are ignored for the purpose of calculation of performance. The reporting engine is expected to lag a maximum of 10 minutes of time. The computations are large and complex. Also, the computed data has to be visualized and requires time to do the same. There should not be a single minute of unscheduled downtime for any reason.

2.2.4 Interfaces

This section deals with the way the application interacts with the system and the user. It tells us in detail the kind of input that can be given and the type of output generated.

The user interacts with the system by means of the website. The website is user-friendly interactive and has a simple GUI. Both keyboard and mouse are used to give inputs. The user interfaces are built using HTML, CSS and JavaScript. The website is responsive and hence automatically adjusts the components based on the screen size. The website runs smoothly in any Java-enabled web browser

2.2.5 Hardware Requirements

The hardware requirement is restricted to an administrator device (Desktop/Laptop) with the following configuration:

- 2.7 GHz Intel i5 Processor or higher
- 8 GB DDR3 RAM or higher
- 1.5 GB Graphics Card or higher
- 128 GB Flash Memory or higher

2.2.6 Software Requirements

The software requirements of the project are listed below :

- Platform: JDK 7
- Language: Java 1.7
- Compulsory Packages: GNIP, Redis, RabbitMQ, Spring Hibernate, Log4j, Twitter4j
- Servlet Container: Apache Tomcat
- Database: MySQL
- Version Control Tool: GitHub
- Project Management Tool: JIRA
- REST Client: Postman
- Visual Interface : HTML,CSS, JavaScript
- IDE: IntelliJ IDEA

2.3 Summary

The specific requirements and constraints of the system have been dealt in detail in this chapter. These include the assumptions, dependencies and functional, non-functional and performance requirements. The hardware and software requirements have been specified as well. All these have to be managed while using the system.

Chapter 3

HIGH LEVEL DESIGN OF AUTOMATIC TICKET GENERATION FOR COMPLAINTS RAISED ON SOCIAL MEDIA

High Level Software Design is the first design step after analyzing all requirements for software. The goal is to define a software structure which is able to fulfil the requirements. Also the non-functional requirements, such as scalability, portability and maintainability have to be considered in this step.

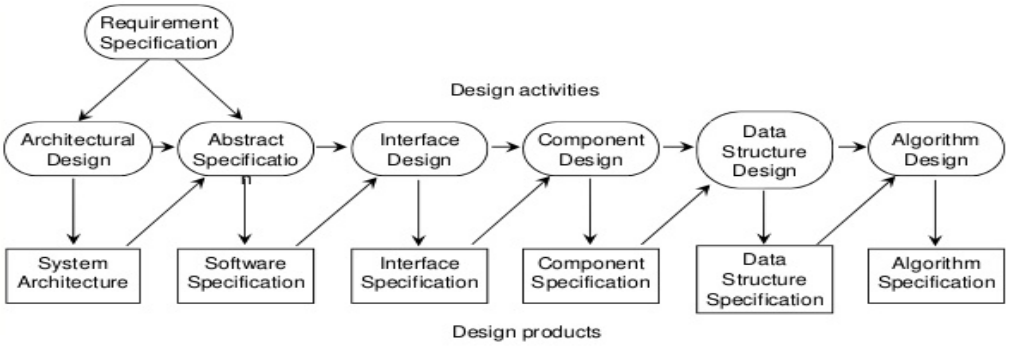

Figure 3.1: Software development process

The software design process in this project is depicted in the Figure 3.1. In Architectural design, all the subsystems are decided for the system and their relationships are identified and also documented. Abstract specification provides a subsystem constraints and operating conditions. Interface design specifies the interfaces among the subsystems. Next the component design is done where the specification for individual components is done and their interfaces are discussed. Data structure design gives an insight into the actual implementation of the project. The data structures and internal implementation details are specified in it. Algorithmic design specifies the algorithms used in the implementation. Usually the algorithms are expressed as pseudo code.

3.1 Design Considerations

There are several design consideration issues that need to be addressed before designing a solution for the system to be implemented. The following sections describe constraints that have heavily impact the software, the methods and approaches used for the development and the architectural strategies. A description of the system design is also provided as part of this section.

3.1.1 General Considerations

General constraints which need to be considered to use the system are listed below:

- Proper authentication must be provided to access the system.
- The end users are not aware of the underlying logic and it is necessary that they need to be able to use the system without any issues.
- Appropriate errors need to be displayed to indicate the problems.

3.1.2 Development Methods

A software development methodology or system development methodology in software engineering is a framework that is used to structure, plan, and control the process of developing an information system.

In the development of the project, Agile development methodology has been used. In particular the Scrum framework which is used for developing and sustaining complex products has been implemented. Scrum is a framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value.

Scrum, defined as “a flexible, holistic product development strategy where a development team works as a unit to reach a common goal”. It challenges the assumptions of the “traditional, sequential approach” to product development, and enables teams to self-organize by encouraging physical co-location or close online collaboration of all team

members. There is a daily face-to-face communication among all team members and disciplines in the project.

The Scrum framework consists of Scrum Teams and their associated roles, events, artefacts', and rules. Each component within the framework serves a specific purpose and is essential to Scrum's success and usage. The rules of Scrum bind together the events, roles, and artefacts , governing the relationships and interaction between them.

A key principle of scrum is its understanding that during production processes, the requirements may be changed due to certain factors(often called requirements volatility), and that unpredicted challenges cannot be easily addressed in a traditional predictive or planned manner. As such, scrum adopts an empirical approach. It accepts that the problem cannot be fully understood or defined, and hence focuses on maximizing the team's ability to deliver quickly, to respond to emerging requirements and to adapt to evolving technologies and changes in market conditions.

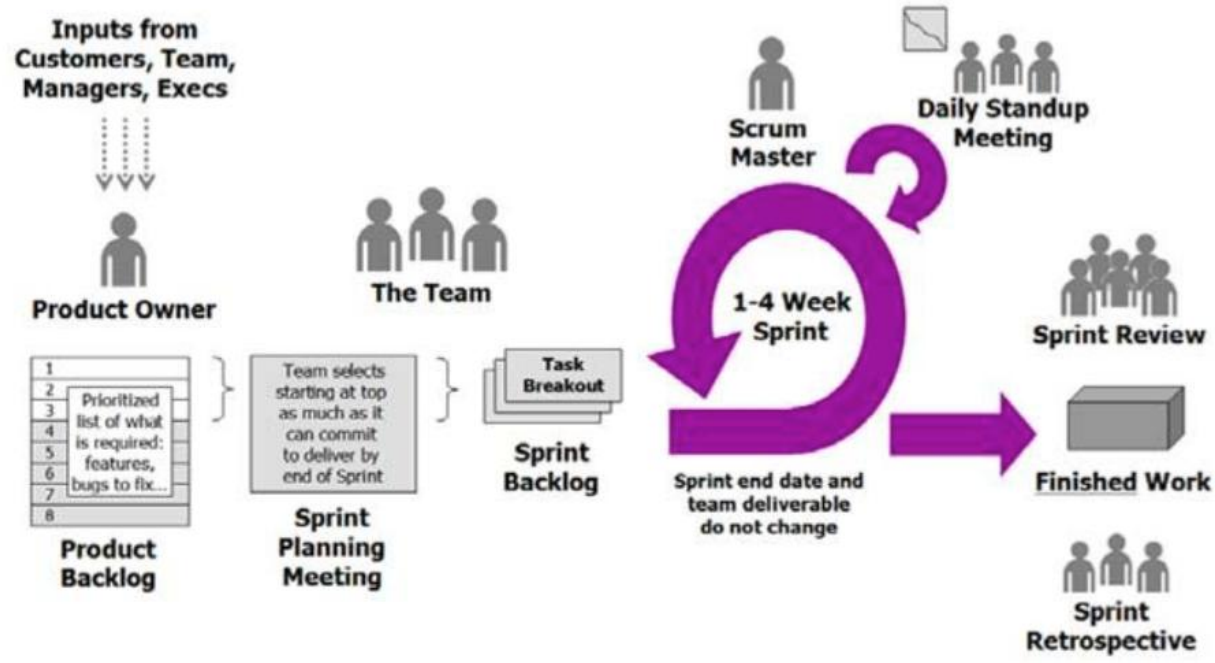

Figure 3.2: Scrum framework

The main components of Scrum Framework as shown in Figure 3.2 are:

- The three roles: Scrum Master, Scrum Product Owner and the Scrum Team

- A prioritized Backlog containing the end user requirements
- Sprints
- Scrum Events: sprint planning, daily scrum meeting, sprint review and sprint retrospective meeting.

The heart of scrum is a sprint, a time-box of two weeks or one month during which a potentially releasable product increment is created. A new Sprint starts immediately after the conclusion of the previous Sprint. Sprints consist of the Sprint planning, daily scrums, the development work, the Sprint review, and the Sprint retrospective.

- In sprint planning, the work to be performed in the Sprint is planned collaboratively by the Scrum Team.
- The daily scrum meeting is a 15-minute time-boxed event for the scrum team to synchronize the activities and create a plan for that day.
- A sprint review is held at the end of the sprint to inspect the Increment and make changes to the product backlog, if needed.
- The sprint retrospective occurs after the sprint review and prior to the next sprint planning. in this meeting, the scrum team is to inspect itself and create a plan for improvements to be enacted during the subsequent sprint

3.2 Architectural Strategies

This section describes the design decisions and strategies that affect the overall organization of the system and its higher-level structures. These strategies when implemented in the project provide insight into the key abstractions and mechanisms used in the system architecture.

3.2.1 Programming Language

Java programming language is used to design the system. Java supports object-oriented programming, a wide range of data types and application programming interface for handling the data. The user interface is developed using HTML, CSS and JavaScript.

Java is the optimal programming language for the project because of the following reasons -

- Java is platform independent which is one of its design goals. This results in great level cross-platform independence and portability.
- Java has simple methods for reading images from files of different formats. This is an important feature as **reading input images is the basic requirement of project.**

3.2.2 User Interface Paradigm

User interface paradigm specifies the design strategies involved that decide upon the method in which the computers communicate with the end user. Both the console and an interactive GUI are examples of user interface. In this project, however, the end users interact with the system primarily through the interactive website accessible from any java enabled web browser. But the developers are able to control and debug using the console and the command line interface.

3.2.3 Error Detection and Recovery

Error detection and recovery is an important aspect of the implemented project. Exceptions may occur, may occur both in the front end and the back end. Validations have been included in the front end to ensure that the user does not input invalid information which may again lead to exceptions in the backend. Try-catch blocks and logging has been implemented to capture the exceptions and debug the errors by cross checking the logs.

3.2.4 Data Storage Management

The application stores the information of all the users and their interaction with the system. This creates a lot of data for over a million users and hence MySQL is used as the database.

MySQL is one of the most commonly used RDMS in existence. There is wide-ranging support for it and the queries are simple and easy to use. A number of stable interactive interfaces such as Sequel Pro and Php MyAdmin makes it easier to access the database and not restricting ourselves the primitive command line tools.

Some of the features of the data storage management in use for this system:

- No data is ever deleted from the system and only the is_available status is changed to 1 indicated it is a soft delete.
- All the data is stored in both the master and slave database. Indexing has been implemented to support faster data retrieval.
- All the data retrievals are performed only on the slave database.
- Multiple layers of authentication have been implemented to prevent unauthorized access.
- Both virtual and physical servers have been used for data storage.
- Data corresponding to reporting is automatically archived after one year and not deleted.

In memory data structures like Redis is used for caching purpose while the data from GNIP is still in process. Also properties like set operations that is present in Redis makes it feasible to perform some amount of processing in the database side. Redis stores the data in the form of key-value pairs making it easier for data retrieval for further uses.

3.2.5 Communication Mechanism

Data is exchanged between the components through JSON. All the programming languages convert the data into JSON before transmission and also convert to its native form from the incoming JSON. GET, POST and PUT requests are made to the APIs for data storing, retrieval and processing. Internet itself is used as the common pipeline for communication.

3.3 System Architecture

The software system architecture of a computing system is a depiction of the system that aids in the understanding of how the system will behave. Software system architecture serves as the blueprint for both the system and the project developing it, defining the work assignments that must be carried out by design and implementation teams. The architecture is

the primary carrier of system qualities such as performance, modifiability, and security, none of which can be achieved without a unifying architectural vision. Architecture is an artefact for early analysis to make sure that a design approach will yield an acceptable system. By building effective architecture, you can identify design risks and mitigate them early in the development process.

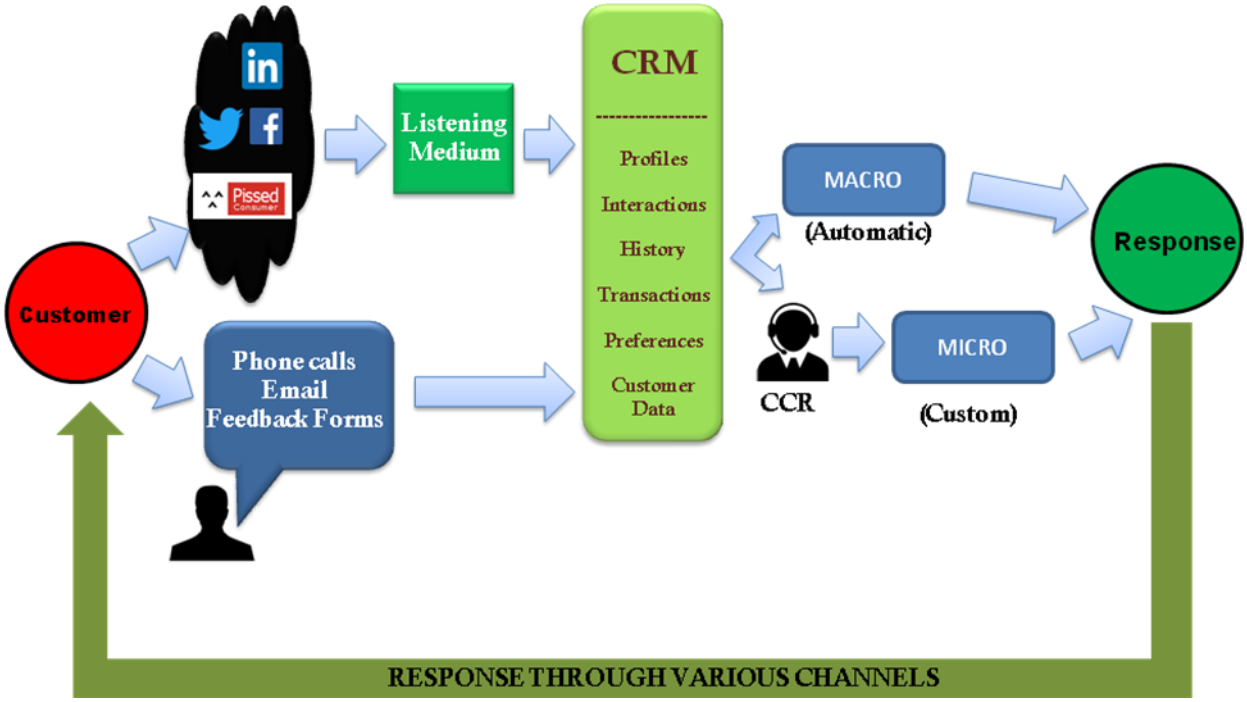

Figure 3.3: System architeccture

Figure 3.3 shows the architecture of the project. Customer has the option of using traditional CRM or SCRM methods to raise their complaints or issues. In case of SCRM, a listening tool is used to extract customer complaints given to a particular company. Phone calls, email and feedback forms come under the category of traditional CRM. Both types of complaints and communications are stored in a database. Various other details are also caputed by the system. An immediate response called a macro response is sent, It is usually generic and just indicates the user that their complaint has been registered. A micro and custom response is sent to the client by a CCR.

3.4 Data Flow Diagrams

Data flow diagrams indicate the flow of data from external entities into the system and later the logical flow of data within the system using standard predefined universally graphical notations. The DFD is part of the structured analysis and design paradigm and is used to show the way that data flows through a system including the processes, data stores and entities that relate to the data. A data flow diagram (DFD) illustrates how data is processed by a system in terms of inputs and outputs. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled.

3.4.1 Data Flow Diagram – Level 0

DFD level 0 is also commonly referred as context diagram and provides an abstract view of the entire system under discussion. It generally comprises of a single process describing the entire system with the external entities interacting with the system and relationships as well as the exchanges between them.

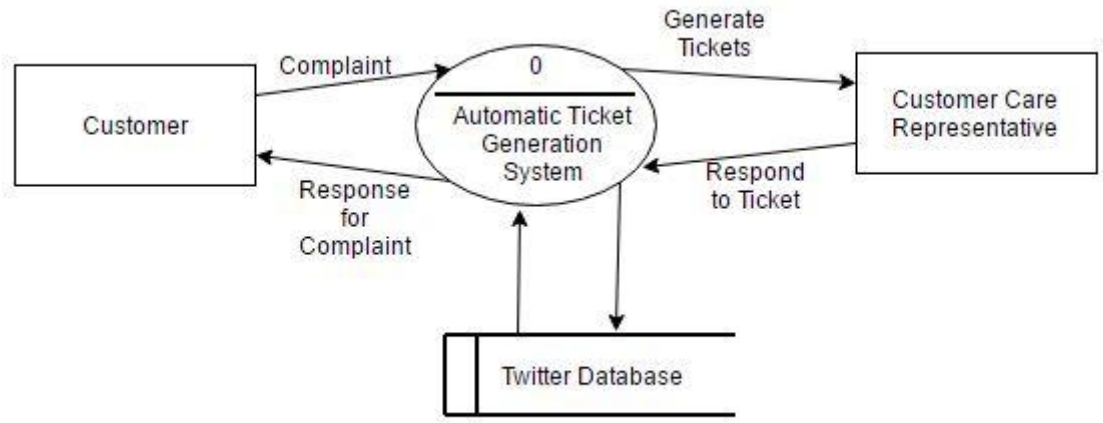

Figure 3.4: DFD level 0

In the Figure 3.4, represents the DFD level 0 of the entire system. The external entities are the customer and the CCR. The data is stored in the Twitter Database. Customer raises the complaint on social and the system captures it automatically raises a ticket to be

handled by the CCR. The CCR responds to ticket using the system that is in turn communicated to the customer

3.4.2 Data Flow Diagram – Level 1

DFD level 1 goes one step deeper into the context diagram and indicates all the key processes involved in the system. DFD level 1 gives a high-level abstraction of all the interactions between the participating entities and the major processes in the system.

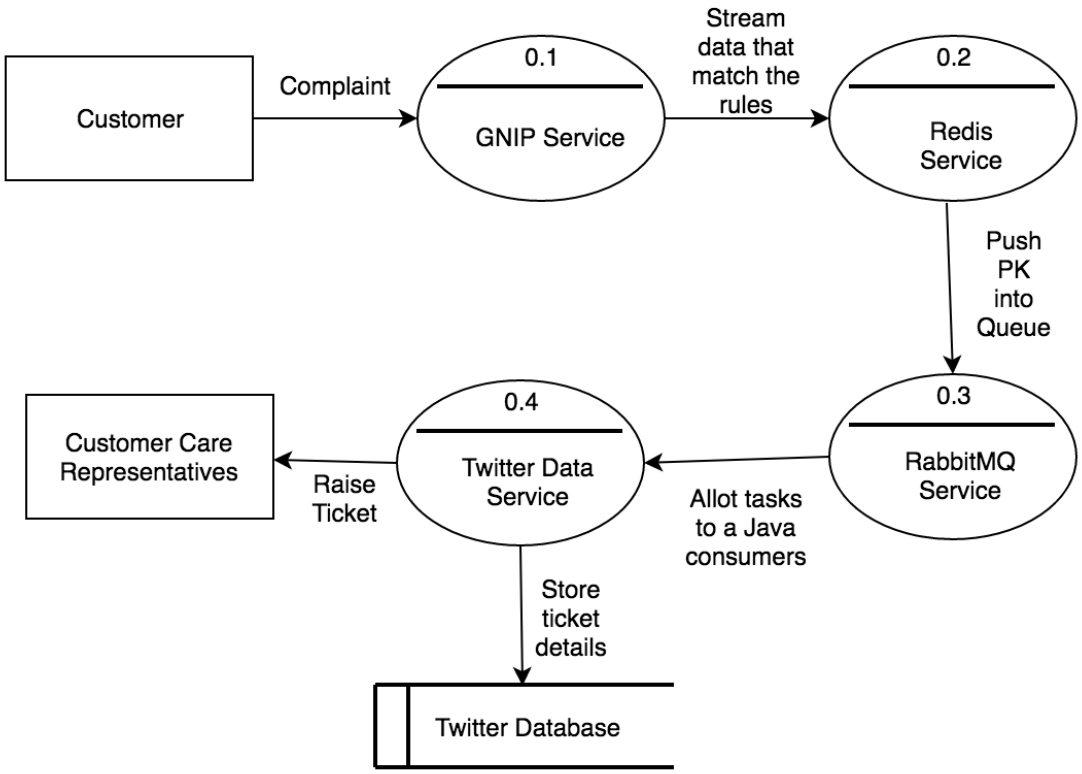

Figure 3.5: DFD level 1

In the Figure 3.5 it shows the DFD Level 1 of the system. There is a single sequential flow of the important modules in the system. The external entity Customer raises a complaint on a social media platform like Twitter, Facebook, etc. GNIP service streams the data matching the rules specified in the GNIP console and feeds the Redis Service. In Redis, a primary key is the brand name and the customer information and the JSON activities are sorted in a chronological order. Only the primary key is pushed into the RabbitMQ Service.

Each primary key is allocated to a single Java Consumer for further processing and the primary key is locked till the complete processing is complete. The Twitter Data Service is responsible for processing the data and raising the tickets on the incoming data and also store it into the database. The ticket details are returned to the CCR for further actions.

3.4.3 Data Flow Diagram – Level 2

DFD Level 2 expands on the major processes in the DFD Level 1 and provides a more in-depth analysis of the data flow. DFD Level 2 goes one step deeper compared to DFD Level 1. They generally provide a detailed explanation of the core processes of the system.

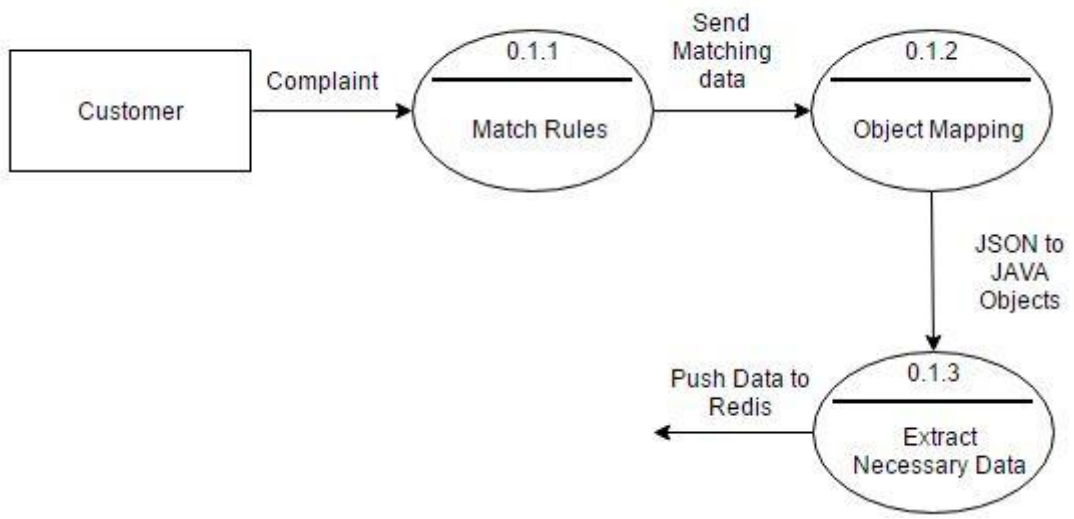

Figure 3.6: DFD level 2 – GNIP service

In the Figure 3.6, the GNIP Service is explained in detail. The complaints raised by Customers are captured by the GNIP Streaming API and checked for matching rules specified in the GNIP console. If the rules match then, the data in JSON are converted to equivalent Java objects using object mappers. The necessary data is extracted and pushed into Redis Service.

In Figure 3.7, the data from GNIP is pushed into Redis . A primary key is generated for the data pushed from GNIP which is a combination of brand name and customer details. The data is sorted to be chronologically to accurate. Only the primary keys are pushed into the RabbitMQ Service.

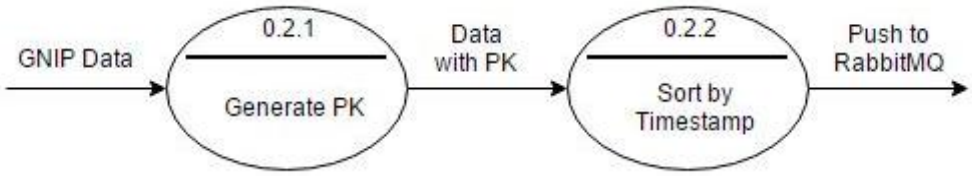

Figure 3.7: DFD level 2 – Redis service

In Figure 3.8, the primary keys from Redis are pushed into the queue. Each key is allotted a Java consumer for further processing. The primary keys are locked until the processing is complete. The Java consumer essentially executes the tasks specified in Twitter Task Service and also the data is stored in the database. Once the data is processed, the complaint tickets are raised and the lock on the primary key is removed and data is deleted from Redis.

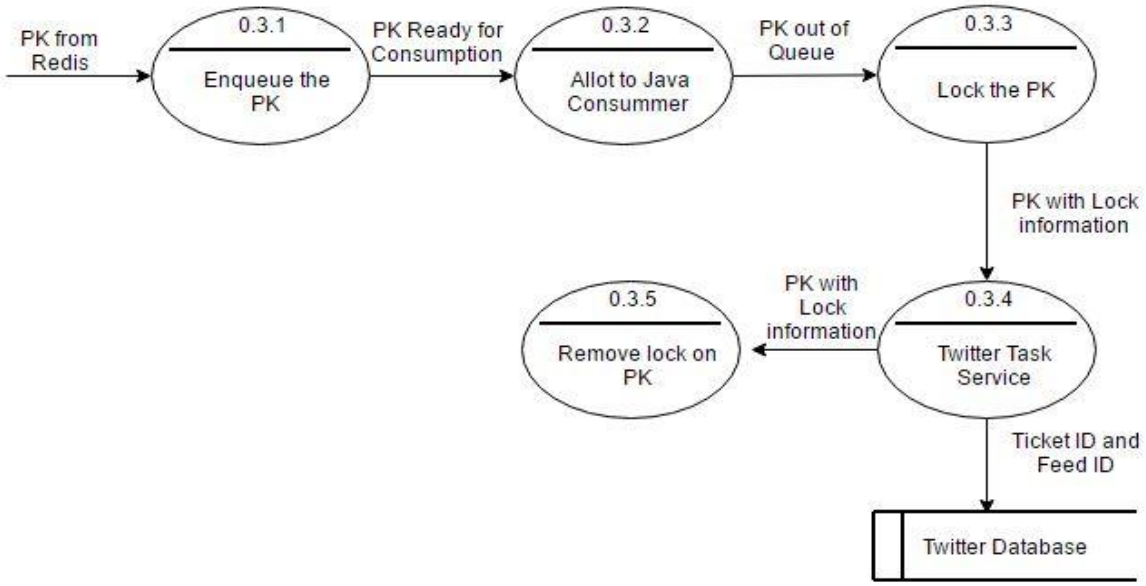

Figure 3.8: DFD level 2 – RabbitMQ service

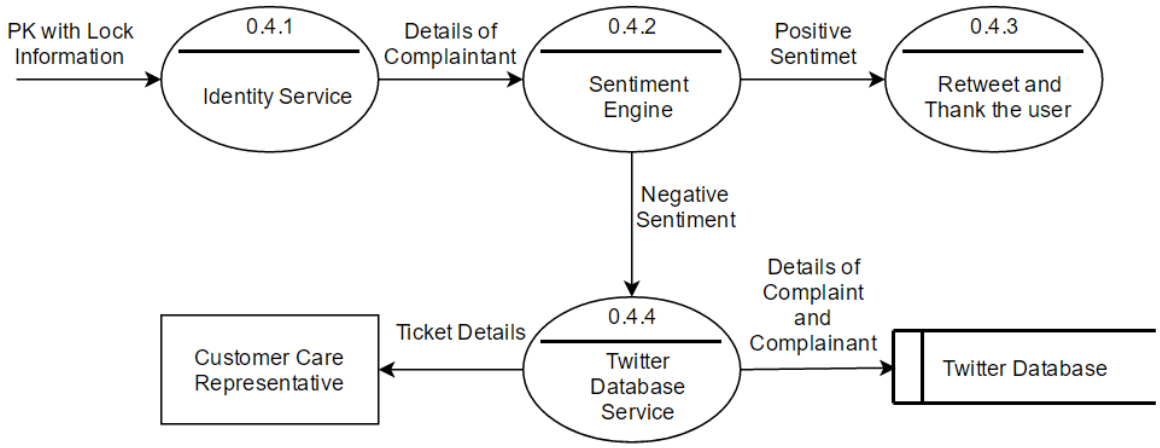

Figure 3.9: DFD level 2 – Twitter Data service

In Figure 3.9, the Twitter Data Service is explained in detailed. The primary key is sent from the queue is taken in by the Identity Service to determine the data corresponding to each primary key. Identity Service determines details about the customer. The data extracted is sent over to the Sentiment Engine to determine if it is a positive or negative. If the data is positive, the customer is acknowledged and thanked by the CCR. If the data has a negative sentiment, a ticket is raised on the data and the details of the ticket raised are returned to the CCR to further handle the complaint. Once the tickets are raised, their details are also stored in the database.

3.5 Summary

High level design provides an abstract level explanation of the entire system. Right from the development methodologies all the way to the data models, every aspect is discussed in the high level design. The data models depict how data is processed by the system. This constitutes the analysis level. The notations applied above represent functional processing, data stores and data movement amongst the functions. The purpose of the chapter is to describe major high-level processes and interrelation in the system. DFD and an architectural diagram are used to explain the design.

Chapter 4

DETAILED DESIGN OF AUTOMATIC TICKET GENERATION FOR COMPLAINTS RAISED ON SOCIAL MEDIA

Detailed design of the system is the only design activity left prior implementation begins. The hardest design problems must be addressed by the detailed design or the design is not complete. The detailed design is an abstraction compared to actual source code but should have enough detail to ensure that translation to a source is a precise mapping instead of a rough interpretation.

This chapter explains the project using structure chart, control flow diagram and flowcharts.

4.1 Structure Chart

A structure chart is a graphical representation showing the breakdown of the system to the lowest manageable parts. It represents the system in more detail than DFD. It breaks down the entire system into lowest functional modules, describes functions and sub-functions of each module of the system to a greater detail than DFD. Structure charts are used in structured programming to arrange program modules into a tree. Structure chart represents the hierarchical structure of modules. At each layer, a specific task is performed. Structure charts follow a top-down design approach. Each module is represented by a box, indicating the module's name.

The relationships between modules indicating data transfers are shown using arrows and the whole thing is represented as a structure chart. The structure charts are read from left to right and often move from leaves up to understand the functionality of each module.

In the Figure 4.1, the system is represented in the form of a structure chart. Starting from the left, the GNIP Service is the first the process to be executed. On the social media platforms, all the data that match the rules specified in the GNIP console is streamed to us.

The JSON data are converted to Java objects by object mappers and sent to Redis for generating the primary key and sorting the data is a chronological order. The primary key is sent across to RabbitMQ to be allotted to a Java consumer for further processing. A lock is imposed on each primary key to avoid multiple consumers operate on the same data. Based on the primary key, the data is pulled from Redis to determine the identity of the Customer who has generated the data in the Identity Service. The data is analyzed to determine if a ticket has to be raised or not by the Sentiment Engine. If the sentiment is negative, the data is stored and a ticket is raised by the Twitter Data Service. The details of the ticket are forwarded to the CCR to be handled personally. The lock on the primary key is released and the data is deleted from Redis.

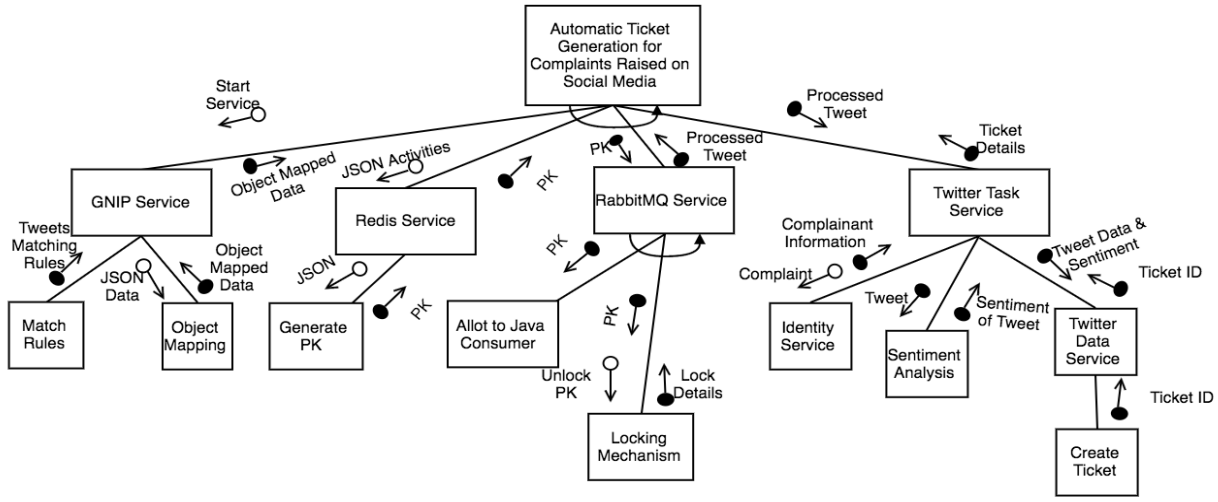

Figure 4.1: Structure chart of the system

4.2 Control Flow Diagram

Control flow diagram indicates the flow of control in the system or a business process. Each control flow diagram may comprise of loops, conditional statements, etc. It is one of the most common techniques for project planning.

In the Figure 4.2, it is observed that the control flow in the project is purely sequential. The Customer Tweets the complaint on Twitter. GNIP captures the data streams it across the internet. Redis and RabbitMQ Services are preparing the data received to be processed. The Java consumers handle each tweet individually and pass it through the Identity Service, Sentiment Engine and Twitter Data Service to process and raise the tickets.

The WIFE(Work Integrated Flow Engine) service is responsible to handle the tickets further. The WIFE service comprises the Rules and Priority Engine along with the GUI. It is responsible for coordinating all the other services.

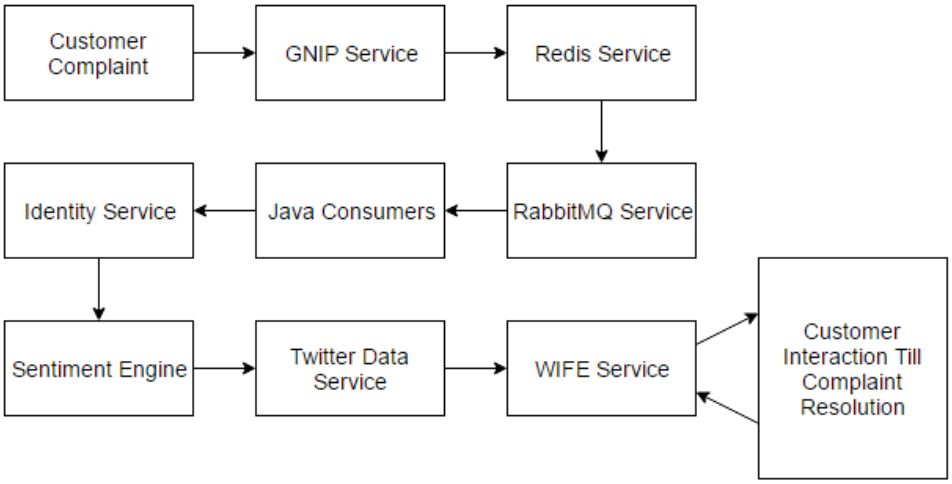

Figure 4.2: Control flow diagram of the system

4.3 Functional Description of the Modules

Functional description of the modules is usually done using flowcharts. A flowchart is a visual representation of the logical sequence of steps and decisions needed to perform a process or a series of processes in order to achieve a particular predefined goal. Each step in the sequence is noted within a diagram shape. Steps are linked by connecting lines and directional arrows.This diagrammatic representation illustrates a solution model to a given problem in an orderly and well-defined manner to avoid confusion. The flowchart is a good tool to explain the workflow to individuals who do not possess in-depth knowledge of the topic under discussion.

4.3.1 GNIP Service

- Purpose: To stream matching Tweets from Twitter.
- Functionality: This module is responsible for streaming data using the GNIP streaming service that match the rules specified in the GNIP console. The data received is in the form of JSON and is converted to Java objects by using object mappers.

- Input: None
- Output: JSON mapped to Java Objects
- Flowchart: In Figure 4.3, the GNIP Service is discussed in detail. GNIP is a streaming API that provides data from multiple social media platforms. The rules specified in the GNIP console are used to instruct GNIP stream data that match the requirements required by the system. The data that satisfy the rules are streamed over the internet to our server for further processing. Since the internet is not foolproof, the GNIP Replay feature to restream data that might have been missed during the downtime. Once the data is available, the JSON data is converted into Java objects by object mappers and only the required fields are extracted for further processing.

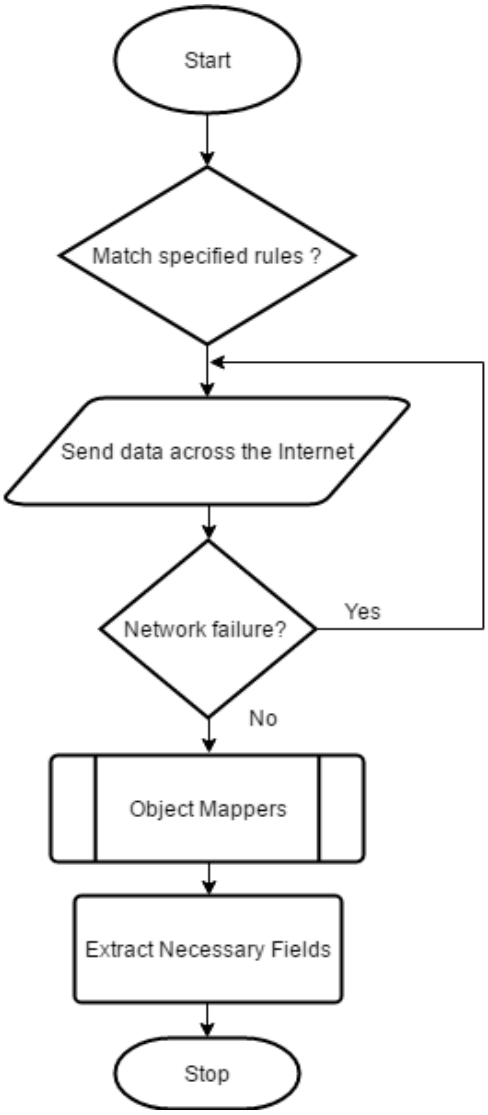

Figure 4.3: GNIP service flow chart

4.3.2 Redis Service

- Purpose: To create, store the primary key and sort the data in chronological order.
- Functionality: A primary key is created by combining the brand Id and customer Id. The data is sorted chronologically. The primary key is pushed into the RabbitMQ.
- Input: JSON mapped to Java Objects
- Output: Primary keys in RabbitMQ
- Flowchart: In the Figure 4.4, the flowchart throws light on the Redis Service. The data sent over from the GNIP service is assigned a primary key. The primary key is a combination of the Brand ID and User or Customer ID. The data is sorted so that they are in chronological order. Only the primary key is pushed into the RabbitMQ.

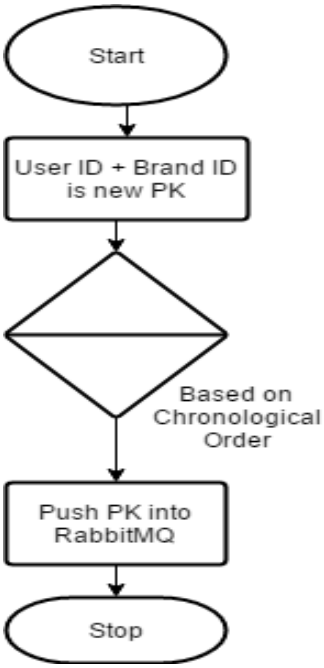

Figure 4.4: Redis service flow chart

4.3.3 RabbitMQ Service

- Purpose: Allot primary key to Java consumers for further processing
- Functionality: The primary keys are individually popped out of Redis and assigned to a Java consumer and locked to prevent other consumers from accessing it. Data corresponding to each key is extracted from Redis and given to Twitter Task Service

for further processing. After the processing is complete, The primary keys and the data associated with it is deleted and the lock is released.

- Input: Primary keys from Redis
- Output: Ticket details
- Flowchart: In the Figure 4.5, the RabbitMQ Service is discussed. Individual primary keys are popped out of the queue one after the other and allotted to Java consumers. On allotting to a Java consumer, a lock is imposed on the primary key to ensure that the same data is not processed multiple times. The Java consumer fetches data corresponding to each primary key for further processing. The data is forwarded to Twitter Task Service to raise tickets and return Ticket and Feed ID. On completion of processing of the data, the primary key and corresponding data are deleted from Redis. The lock imposed on the primary key is also lifted.

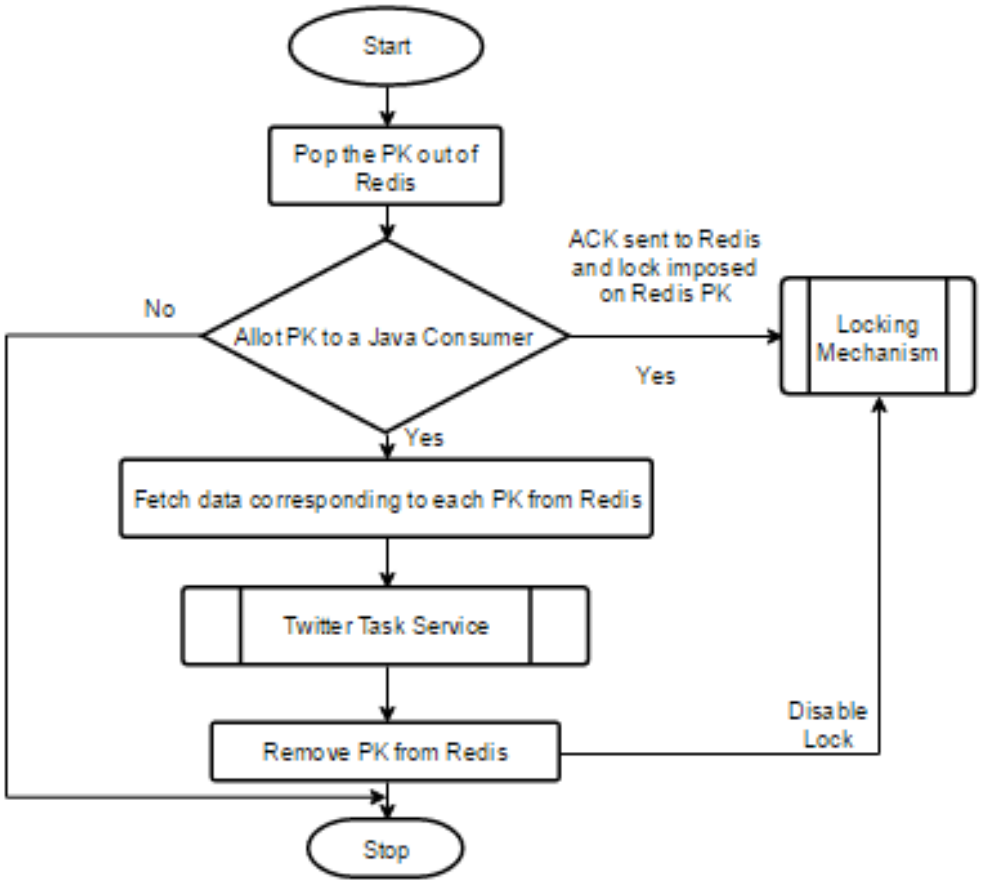

Figure 4.5: RabbitMQ service flow chart

4.3.4 Twitter Data Service

- Purpose: To process the data and automatically raise tickets if necessary
- Functionality: The Identity Service is responsible for extracting data user. The Sentiment Engine determines the sentiment of the data tweeted. If the sentiment is negative, the data is stored in the database with the help of Twitter Data Service and the ticket details like Ticket Id, Feed Id etc are returned to the customer care executives to handle the ticket of their end.
- Input: Primary key and data associated with the primary key
- Output: Ticket Details
- Flowchart: In the Figure 4.6, the workflow of the Twitter Data Service is explained. The data is sent to the Identity Service to gain more knowledge about the Customer who is responsible for generating the following data. The Sentiment Service uses a linear 2 class SVM to determine the sentiment of the data provided. If the data is negative, the data is pushed into the database. The Twitter Database Service is responsible for raising a ticket on the data and returning the ticket details to CCR to handle the complaint manually.

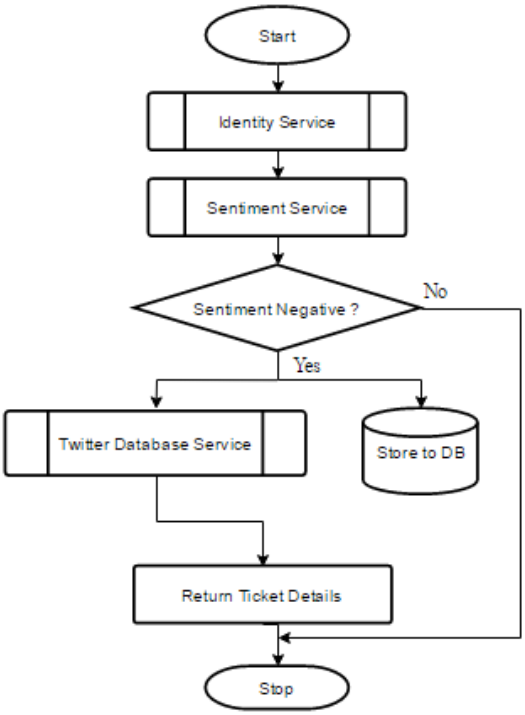

Figure 4.6: Twitter Data service flow chart

4.4 Summary

The internal working of the project with the necessary flow chart has been described in this chapter. A clear view on control flow within software was conveyed by the structure chart and other tools explained in the chapter.

Chapter 5

IMPLEMENTATION OF AUTOMATIC TICKET GENERATION FOR COMPLAINTS RAISED ON SOCIAL MEDIA

Implementation is the process of execution of the plan and design decisions taken in the previous chapters. In this chapter, the low-level designs are transformed into their equivalent language specific constructs based on the specifications provided in the Requirements Specifications. This chapter looks into the actual execution of all the designs discussed earlier. This chapter gives an insight into the platform selection and programming language selection as well as steps to execute the projects.

5.1 Programming Language Selection

Programming language selection is a critical part of the implementation. All the positives and negatives must be carefully compared before coming to a conclusion. Multiple languages may suit the project requirements, however, it is necessary to keep in mind of some factors like scalability, portability, performance, etc. Just one programming language cannot be used for building an entire project, it is necessary to identify if a given language provides support and has stable dependencies for the remaining support languages.

Java is a general-purpose high-level language that has a high degree of portability. Java 7 has been used throughout the codebase. Though a stable version of Java 8 is available, Java 7 has been chosen since there are stable as well compatible integrations with the remaining packages used in the project.

Spring is a Java based MVC framework commonly used for development of websites and other enterprise grade applications. The beans can be configured in multiple ways using XML, annotations, etc. Spring framework is best known for the property of dependency injection and also supports life-cycle management of all the Java components. The complexity to configure Spring increases with more features and is bound to get error prone. Spring Boot is a framework from the team at Pivotal, designed to simplify the bootstrapping

and development of a new Spring application. The framework has a rather opinionated approach and is designed to support the most common use case. It implements the most common use case of a particular task with little or no fuss of configuring Spring.

Gradle is an open source advanced build management system that revolutionised JVM-based build tools. It takes best practices from both Ant and Maven and it elevates to the next level.

Hibernate is an ORM tool that is used to exchange data between Java and a relational database. Hibernate is a lightweight open source tool and provides a non-invasive approach to the way the programmers interact with the database. Hibernate can run within or without a server, ie it will suitable for all types of Java applications.

Apache Tomcat is web container that allows to run Java Server pages and servlet based web applications. Tomcat by default runs on 8080 and can also act as an HTTP server. The Apache Tomcat is an open source implementation that supports JSP, Java WebSocket technologies, etc.

Log4j is a simple, reliable, fast and flexible logging framework that equips developers with detailed context for application failures to debug at runtime without any changes to the application binary. Log4j supports hierarchical logging.

Twitter API is an open source API stack released by Twitter to enable interaction of various web applications with Twitter without actually logging into their website. The REST APIs provide a mechanism to systematically read and write data on Twitter.

Twitter4J is an open source unofficial Java library for the Twitter API. With Twitter4J, one can easily integrate their Java application with the Twitter service.

MySQL is an open source relational database management system where data is stored in relational tables and widely used for web applications. MySQL is the most popular language for adding, accessing and managing content in a database. MySQL is renowned for its quick proven reliability, processing, ease and flexibility of use.

Redis is an open source in-memory data store that can be used as a cache, message broker or a database. It only supports strings, hashes, lists, sets, sorted sets with range

queries, bitmaps, hyperloglogs and geospatial indexes with radius queries. It has an inbuilt functionality to handle data duplication, LRU eviction and multiple levels of on-disk persistence.

GNIP is a streaming service that aggregates data over multiple social media platforms and feeds to the necessary user. The data supplied is based on the rules specified in the GNIP console. GNIP provides real-time as well as historical social data that will power businesses that are heavily reliant on social media data.

RabbitMQ implements Advanced Message Queuing Protocol (AMQP) with delivery guarantees. RabbitMQ is a commonly used message broker in many enterprise grade applications. RabbitMQ has support for multiple languages and supports cross-platform message transfers and communication.

5.2 Code Conventions

Coding conventions need to be followed during a successful implementation of a project. The conventions followed for this implementation are described in the following subsections.

Coding conventions refer to known or acknowledged standards that have been implemented in the code base to develop code in a well-structured, systematic and readable way. This section discusses the coding standards followed throughout the project. It includes the software applications that are necessary to complete the project. Proper coding standards have been followed. This makes it easier to understand any part of the code without much difficulty.

5.2.1 Naming Conventions

The naming convention is a set of predefined rules in use to systematically name identifiers, variables, methods, classes, interfaces and constants.

The names of classes have the first alphabet of each word capitalised. When more than one word is there in the class name, there is no space in between the multiple words.

Eg; TwitterApi

Methods are also named in the same fashion as the classes.

Eg; GetTwitterHandleByResourceId()

Variables are all defined using only lowercase characters. Variables comprising of multiple words are separated by an underscore. Simple and meaningful words or phrases have been chosen for variables. Eg; number_of_retweets

The constants are all defined using uppercase alphabets. The constants that have multiple words are separated by an underscore. Eg; DB_CONNECTION_PORT

5.2.2 Comments

Comments are statements that are not executed by the compiler. Comments are part of the code intended to provide information or explanation about the variable, method, class or any statement present in the code. It can also be used to hide program code for a specific time. Comments are extensively used throughout the project to improve the code readability.

The author is mentioned at the start of any class along with the date it was first created. General style is given below.

```
/**
*   @author <author name>
*   Date : <DD/MM/YYYY>
*/
```

Prior to each method, a multi-line comment has been included with line 1 with the name of the method. The next few lines have a short description of the method. Then the input parameters have been specified with the data types. The last component of the multi-line comment is the return type of the function with the data type. General style is given below.

```
/**
*   Name of the method
*   Description of the method
*   @param <parameter name> : datatype
*   @param <parameter name> : datatype
*
```

```
*    @return <data returned> : datatype
*
*/
```

Single line comments are also widely used throughout the code to give any necessary information that is necessary in between the code.

```
Eg;  //  This comment is to help you understand the need for a callback
```

5.3 Difficulties Encountered and Strategies Used to Tackle

A project of this size has multiple dependencies that lead to roadblocks in the development process. This section explains the issues and the clever strategies incorporated to work around the existing issues.

Differentiating between a new tweet thread and a reply to a tweet on Twitter is quite a challenge. Twitter implements a parent child relationship among the original tweet and reply tweets. GNIP only provides the parent id of the Tweet. There may be a series of replies over duration of time and needs to be stored in a fashion such that it resembles the display in Twitter itself. While processing the data received from GNIP, we add an additional field to identify the most recent Tweet and append the new tweet to that based on the last entry in the database.

In the initial stages of the project implementation, it was seen that many tweets forwarded by GNIP was being missed are getting lost in the processing phase. One of the key reasons for the failure was that the data generated was so large that the numbers of workers in place were not sufficient enough and the data was getting discarded. A locking mechanism as well as a message broker like RabbitMQ was incorporated to solve the problem. RabbitMQ has delivery guarantees that ensure that the task is completely processed before being deleted. Also RabbitMQ has the feature to hold the data in case if all the workers are busy.

5.4 Summary

This chapter deals with the programming language and development environment followed during implementation of the project. It also explains the difficulties encountered in the course of implementation of the project and strategies used to tackle them.

Chapter 6

SOFTWARE TESTING OF AUTOMATIC TICKET GENERATION FOR COMPLAINTS RAISED ON SOCIAL MEDIA

The aim of Software Testing is to detect defects or errors by testing the components of the system individually. During testing, the components are combined to form a complete system. At this stage, testing is concerned to demonstrate that the software meets the required functional goals, and does not have any anomalies. The test cases are chosen to assure the system behaviour can be tested for all combinations. The proper functionality of the system has been tested with an array of test cases and different combinations of inputs are given to check if the desired behaviour is being performed by the system. Appropriate error messages are being displayed to provide developers with a clear idea if there is a failure or it was successful.

In this chapter, several test cases are designed for testing the behaviour of all modules. When all modules are implemented completely, they are integrated and deployed on Amazon server. Test cases are executed under the same environment. The testing begins with unit testing and functional testing. Then integration and regression testing is done. Load testing has also been done to assess if the system does not go down if multiple users hit the same API continuously.

6.1 Testing Objectives

Testing is a straightforward process. Test the code and check for bugs. Get the bugs fixed and test again. In simple terms, the only objectives are to find bugs, fix them, test again and ensure that the product meets specifications.

6.2 Test Environment

A variety of tests have been performed and different configurations and settings are in use for each test. The testing environments are changed as per the test in consideration. For

any type of testing, the application(s) or microservices need to be running in the system at the time of the test.

Postman is a REST client and it is used for testing the individual services or APIs. For testing features involving multiple functionalities, breakpoints were put in the code and run in debug mode. The code is executed step by step in debug mode from the breakpoints. All the parameters are monitored after execution of each step.

Load testing is done using the command line tool popularly called AB(Apache Benchmark). A particular API can be executed to check the rate limit and the maximum concurrency of the system for that particular operation. Few metrics like average response time, minimum response time, maximum response time, total number requests, the number of request failures, etc as a part of the result.

Always the final results are shown in the GUI. The user interaction with the system is through the websites. With web browser like Google Chrome, the network calls as well as their responses can be monitored. It is one of the simplest ways to test proper functioning of the entire system. Again breakpoints can be included in the browser and the code can be tested.

6.3 Unit Testing

Unit testing is the lowest form of testing in which a single feature is being tested. The test is done as a preliminary test to check how the system performs independent of rest of the project. The key idea behind unit testing is to verify the proper functionality of each feature.

This test is aimed check if the system is able to capture the tweets sent across from GNIP to the system and properly process it. Table 6.1 indicates this test.

This test is aimed to check if the reply from the system has been displayed in Twitter. Table 6.2 indicates this test.

Table 6.1 Capture tweet test

SL. NO.	1
Name of the Test Case	Tweet Capture
Feature Being Tested	GNIP streaming service and tweet processing
Description	To verify the operation of both the GNIP service and Twitter Data service to raise a ticket
Sample Input	Tweet with negative sentiment and matches rules specified in GNIP console
Expected Output	Tweet captured by system and display in UI
Actual Output	Tweet captured by system and displayed in UI
Remarks	Success

Table 6.2 Reply to source test

SL. NO.	2
Name of the Test Case	Tweet Reply
Feature Being Tested	Tweet replied from UI
Description	To verify if a reply from UI to a tweet captured by system
Sample Input	Reply to tweet in the UI with any text
Expected Output	Reply tweet has to be displayed in Twitter website
Actual Output	Reply tweet displayed in Twitter website
Remarks	Success

Many a times a user may tweet something to a particular brand and delete the same before the ticket is resolved. It is necessary to check each time whether the ticket exists before replying to the user. Table 6.3 indicates this test.

Table 6.3 Check tweet exist test

SL. NO.	3
Name of the Test Case	Check tweet exist
Feature Being Tested	Reply to a tweet
Description	Before replying to a tweet, it is necessary to check if the parent tweet to be replied exists or not. When a reply is made an error message is displayed in the UI.
Sample Input	Delete a tweet for which a ticket has been raised
Expected Output	Error message is displayed
Actual Output	Error message is displayed
Remarks	Success

Table 6.4 Suggestive tweet handle test

SL. NO.	4
Name of the Test Case	Suggestive Twitter handles
Feature Being Tested	Suggestive Twitter handles in the reply
Description	Some companies have multiple support handles that needs to be suggested to CCR as it is not possible to remember the exact handle for all the users
Sample Input	@ followed by at least one valid alphabet in any of the stored suggestive handles
Expected Output	Display top 5 relevant Twitter handles in a drop down list
Actual Output	Displayed top 5 relevant Twitter handles in a drop down list
Remarks	Success

Table 6.4 indicates the test to provide suggestive handles to each brand when replying to the complaints of the customer. It is easier for the customer care executives to reply if they

don't have to remember all the handles by heart and suggestions pop up as and when the CCR enters more characters in the text area. This test is solely designed for that purpose.

Table 6.5 Manual ticket creation test

SL. NO.	5
Name of the Test Case	Manual Ticket Creation
Feature Being Tested	Manual Ticket Creation from Feeds
Description	To be able to raise tickets from neutrally classified tweets.
Sample Input	In Twitter feeds section, raise a ticket for any neutrally classified tweets
Expected Output	Ticket must be created and ticket number must be generated
Actual Output	Ticket was created and ticket number must be generated
Remarks	Success

The sentiment engine is not perfect and has its fair share of drawbacks. Any tickets with neutral sentiment are stored in a separate section from where the CCR can raise a ticket manually. Table 6.5 describes the same test in detail.

Any conversation by a user with a brand through direct message in Twitter is treated a problem and a ticket is automatically raised. Table 6.6 checks this functionality. Polling method is used to retrieve the data from the DM and it runs every 5 minutes. The conversation between the user and CCR is not immediate because of the pooling in place.

Table 6.7 describes the test to check if a reply from CCR from the UI is displayed in the DM of the customer. It will take a few minutes for the reply to be seen on the screen since the DM is polled every 5 minutes by the system. The details of the test are highlighted in the table.

Table 6.6 Capture DM from customer test

SL. NO.	6
Name of the Test Case	Generate ticket from DM
Feature Being Tested	Ticket creation for any DM on Twitter
Description	Ticket must be generated for any DM with a brand on Twitter
Sample Input	Follow a brand on Twitter and send them a DM
Expected Output	Ticket must be raised for DM automatically
Actual Output	Ticket raised for DM automatically
Remarks	Success

Table 6.7 Reply to DM test

SL. NO.	7
Name of the Test Case	Reply to DM
Feature Being Tested	Reply to DM from UI
Description	Reply to DM from UI
Sample Input	Ticket is raised from DM. Reply to the DM from the UI
Expected Output	Reply displayed in Twitter when replied from UI to a DM
Actual Output	Reply displayed in Twitter when replied from UI to a DM
Remarks	Success

Table 6.8 highlights the test performed to check if all the media files supported in Twitter is captured by the system and able to render the same in a proper fashion in the UI.

Table 6.9 is devised to check if the system is able to send across the Twitter supported media file types from the GUI to the customer in the form of a reply.

Table 6.8 Capture media files test

SL. NO.	8
Name of the Test Case	Capture multiple media files
Feature Being Tested	Support for media files
Description	Twitter supports various media files to be uploaded. The project must be able to capture these media files and display the same in the UI
Sample Input	Tweet to a brand with negative sentiment along with a Twitter supported media file
Expected Output	Capture media file and display in UI
Actual Output	Media file was captured and displayed in UI (GIF, video and image)
Remarks	Success

Table 6.9 Reply with media files test

SL. NO.	9
Name of the Test Case	Reply with media files
Feature Being Tested	Support for media files
Description	Twitter supports various media files to be uploaded. The project must be able to capture these media files and display the same in the UI
Sample Input	In the reply, send a media file
Expected Output	Media file is shown in Twitter along with tweet
Actual Output	Media file is shown in Twitter along with tweet
Remarks	Success

Table 6.9 is devised to check if the system is able to send across the Twitter supported media file types from the GUI to the customer in the form of a reply.

Table 6.10 Insights real time analytics test

SL. NO.	10
Name of the Test Case	Insights Real Time Analytics
Feature Being Tested	Reporting up to date
Description	To check if the calls for real time analytics is sending the correct timestamp
Sample Input	Choose the Today option in any tab in Insights
Expected Output	Send current timestamp for end date
Actual Output	Current timestamp was sent for end date
Remarks	Success

When the today option is selected , it essential to check if the time from the GUI in the form of a UNIX timestamp is the current time for the end date field as the system cannot compute results of events that are about to happen during the course of the day that are yet to happen. Table 6.10 indicates this test.

6.4 Integration Testing

Integration testing is done by combining two or more independent features together and checks whether they are able to function properly. The interaction and exchanges with the individual units are monitored.

Table 6.11 shows the test that is intended to check if the integration between the backend and front end is proper. The test is intended to monitor the exchanges between the GUI and reporting service. The test checks if the GUI displays the data sent by reporting is properly rendered.

Table 6.11 Integration of Akoshav3 and Reporting API test

SL. NO.	11
Name of the Test Case	Integration of Akoshav3 and Reporting API
Feature Being Tested	The proper functioning of Akoshav3 and Reporting API project is being tested
Description	Akoshav3 comprises of the UI. The Reporting API project provides all the data that is populated in the UI
Sample Input	Run both the projects simultaneously
Expected Output	Data from Reporting API must be displayed in UI
Actual Output	Data was properly rendered in the UI
Remarks	Success

6.5 Load Testing

Load testing is a form of performance testing used to understand the behaviour of the system when it is stressed. Loading is done on both peak and normal conditions and their performance are monitored.

Load testing is done to check if the system does not fail under stress. The Apache Benchmark command line tool has been employed for this sole purpose. A single API is chosen at random and the API is hit multiple times with various concurrency levels. The system fails for very large concurrency since the system exceeds the Twitter specified rate limits. Table 6.12 shows the load testing done on the system. The system worked smoothly even at high concurrencies and large number of requests.

Table 6.12 Load testing

SL. NO.	12
Name of the Test Case	Load Testing
Feature Being Tested	Load testing any one API
Description	Check system does not go down for successive API hits with large concurrency
Sample Input	Any API is hit with various loads
Expected Output	Zero requests should fail
Actual Output	Requests fail when the concurrency is very large and the user reaches Twitter API rate limit
Remarks	Partial Success

6.6 System Testing

System testing is devised to check the complete functionality of the system. All the individual modules are integrated and validated for overall functionality. The main aim of system testing is to verify if the system under test matches the specific requirements of the system mentioned in the SRS.

Table 6.13 describes System testing. System testing is the last phase of testing before deployment in the production environment. Each and every feature is tested rigorously and checked if the software meets the specifications. All the features have been thoroughly checked their functionality has been verified. There are no errors in the system. All the behaviour of the system as it is intended.

Table 6.13 System testing

SL. NO.	13
Name of the Test Case	System Testing
Feature Being Tested	Functioning of the entire system
Description	All the services are deployed in a docker and tested for any issues
Sample Input	-
Expected Output	Smooth functioning of the system
Actual Output	Smooth functioning of the system
Remarks	Success

6.7 Summary

This chapter includes the general testing process, which starts with unit testing of modules followed by integration testing wherein the modules are merged together. System testing is done last where the entire system is tested for its functionality and correctness was performed.

Chapter 7

EXPERIMENTAL ANALYSIS AND RESULTS OF AUTOMATIC TICKET GENERATION FOR COMPLAINTS RAISED ON SOCIAL MEDIA

Experimental analysis was done for deciding the model for analysing sentiment as well as measure both customer and CCR metrics. . The results were closely analysed before the most suitable model was chosen for sentiment engine. Key insights about the system has been discussed . These results are useful for further development of the system.

7.1 Experimental Dataset

```
1	The Da Vinci Code book is just awesome.
1	this was the first clive cussler i've ever read, but even books like Relic, and Da Vinci code were more plausible than this.
1	i liked the Da Vinci Code a lot.
1	i liked the Da Vinci Code a lot.
1	I liked the Da Vinci Code but it ultimatly didn't seem to hold it's own.
1	that's not even an exaggeration ) and at midnight we went to Wal-Mart to buy the Da Vinci Code, which is amazing of course.
1	I loved the Da Vinci Code, but now I want something better and different!..
1	i thought da vinci code was great, same with kite runner.
1	The Da Vinci Code is actually a good movie...
1	I thought the Da Vinci Code was a pretty good book.
1	The Da Vinci Code is one of the most beautiful movies ive ever seen.
1	The Da Vinci Code is an * amazing * book, do not get me wrong.
1	then I turn on the light and the radio and enjoy my Da Vinci Code.
1	The Da Vinci Code was REALLY good.
1	i love da vinci code....
1	i loved da vinci code..
1	TO NIGHT:: THE DA VINCI CODE AND A BEAUTIFUL MIND...
1	THE DA VINCI CODE is AN AWESOME BOOK....
1	Thing is, I enjoyed The Da Vinci Code.
1	very da vinci code slash amazing race.
1	Hey I loved The Da Vinci Code!..
1	also loved the da vinci code..
1	I really enjoyed the Da Vinci Code but thought I would be disappointed in the other books & # 8230;.
1	I do like Angels and Demons more then The Da Vinci Code.
1	The Da Vinci Code was a really good movie.
1	yeah, da vinci code is an awesome movie i liked it pretty interesting.
1	I really like The Da Vinci Code.
```

Figure 7.1: Training dataset

Figure 7.1, shows a snapshot of some part of the training dataset used ie, Twitter dataset. Twitter data from Kaggle is used for training of the sentimental analysis classifier. The same data is also used to calculate the accuracy of the classifiers used. Data was split into training and testing data using stratified K fold (K = 3). Training dataset is only necessary for this sentiment engine. The "1" on the left of each text in the Figure 7.1 indicates the text has a positive sentiment. For negative text, "0" is used to represent it in the dataset.

7.2 Performance Analysis

This section provides the performance analysis of few key components of the system. Visual representations are used to model the data and help users clearly understand and analyse the trends.

7.2.1 Linear SVM Model Selection

The accuracy for two class SVM classifier is around 84 percent and the accuracy of three class pair wise SVM is around 74 percent.

The various SVM models that were experimented along with their accuracy of prediction. It can be seen that two class SVM has the highest accuracy. This model was extended to Three class SVM to incorporate more emotions. This model however shows a dip in accuracy however it is still sufficient for the application under development.

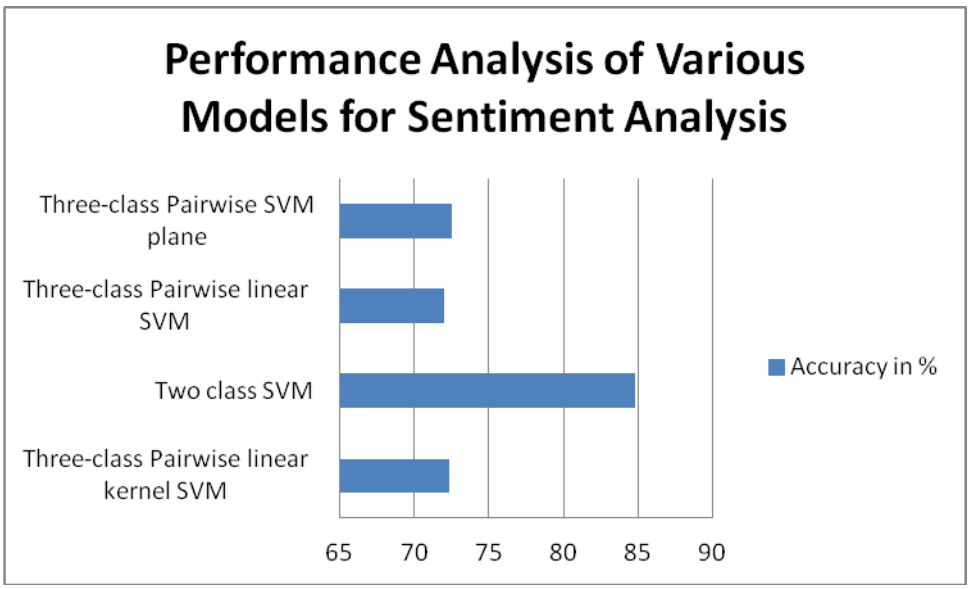

Figure 7.2: Performance analysis of various models for sentiment analysis

Figure 7.2 summarises all the performance of the different types of models tested. From the graph it is evident that two class SVM has the highest accuracy among all the models listed above in the graph

7.2.2 Ticket Count Based on Priority

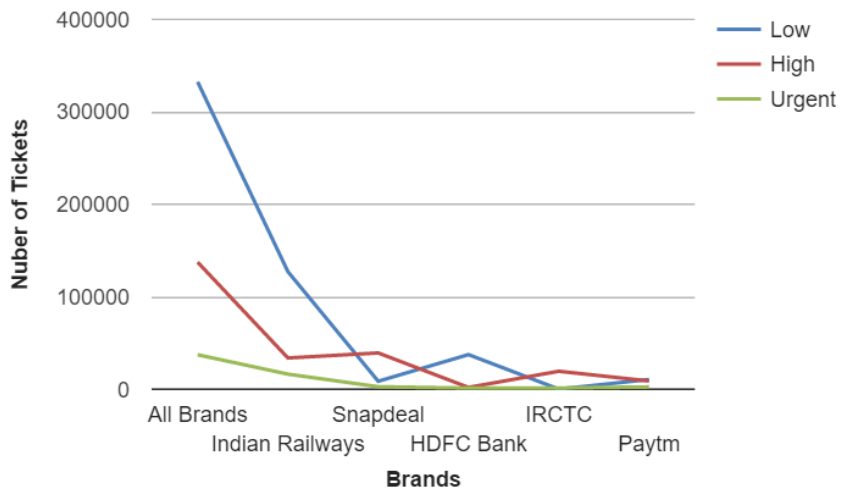

Figure 7.3: Ticket count based on priority

From the graph in Figure 7.3 it is clear that most of tickets raised from social media are of low priority. These issues correspond to minute repairs and minute issues may be facing. Urgent and high priority complaints are few in number and provide extra time for the CCR to resolve them.

7.2.3 Ticket Count Based on Sentiment

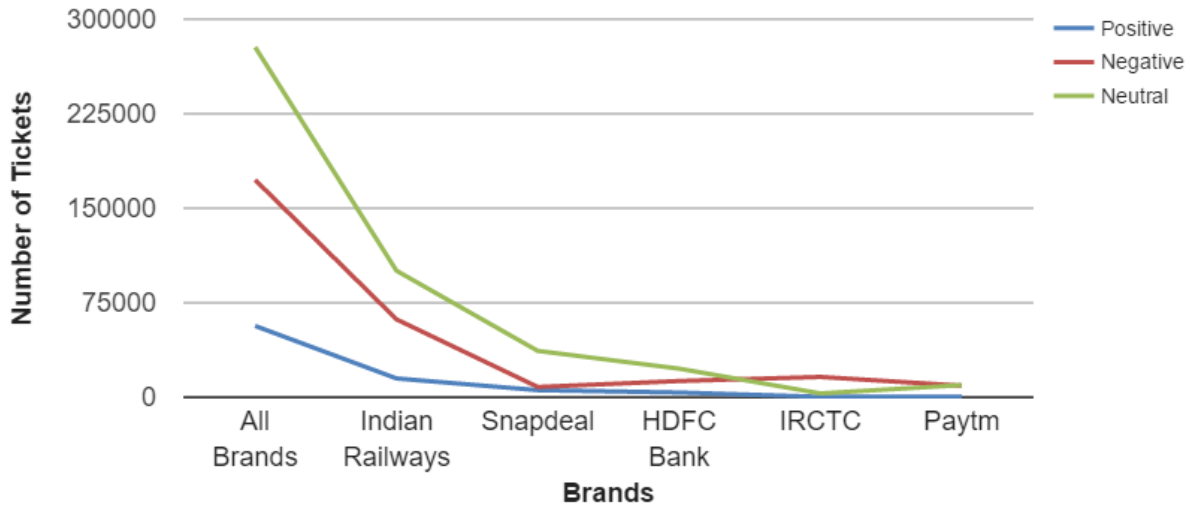

Figure 7.4: Ticket count based on sentiment

From the graph in Figure 7.4 it is evident that most of tickets raised from social media of neutral sentiment. The main reason is that, the user is correctly highlighting the problem and the sentiment engine fails to identify any of the sentiment. Most of the times, the users just refer the various brands in their posts for no specific reason.

7.2.4 New Users

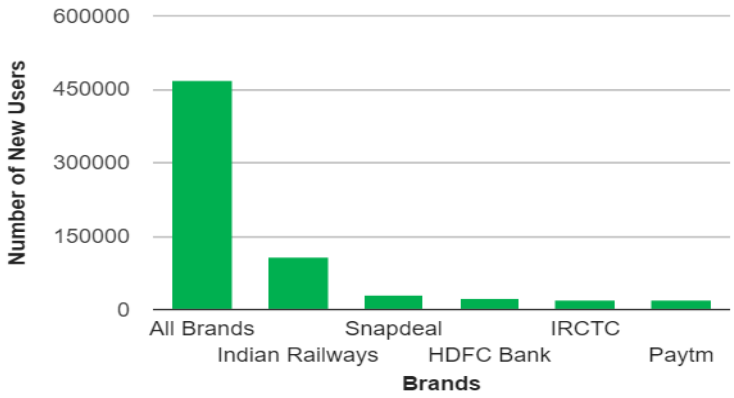

Figure 7.5: Tickets generated by new users

The graph shown in Figure 7.5 is a very important and vital stat indicating the growing popularity of SCRM in India. Day by day more and more users are taking social media to express their opinions and dissatisfaction. It is also metric to prove that the users are in control and not the company anymore for choosing the CRM experience.

7.2.4 Daily Tickets Generated

The graph shown Figure 7.6 indicates the number of tickets generated by system everyday between Jan 1st 2017 and April 1st 2017. It was astonishing to see the number of tickets that were generated every single day. The peaks in the graph refer to the weekends and also signify that most complaints are usually raised on the weekends. Also there is general trend of slow increase in the number of users taking to the social media to voice out their opinion on companies.

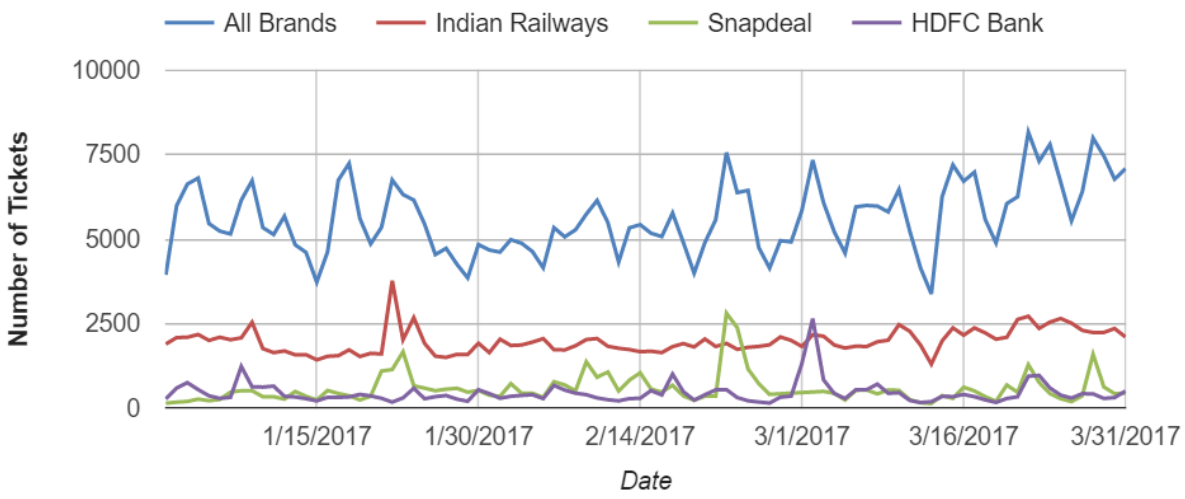

Figure 7.6: Tickets generated by daily

7.2.5 Tickets Deleted

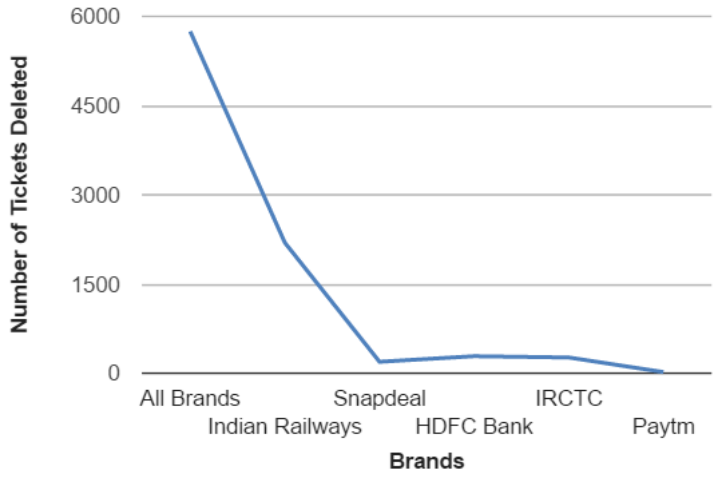

Figure 7.7: Tickets deleted

A major platform with raising tickets from social media platforms is that there is a high possibility that the user may anytime delete their posts for a multitude of reasons. Handling this situation has its own set of challenges both in the reporting end and also there is no clear picture if the CCR was able to resolve the issue. Approximately 10% of tweets are usually deleted after ticket creation. Graph in the Figure 7.7 indicates the trend where more than 30% of all tickets deleted are done by customers using the Indian Railways.

7.2.6 Time to Ticket Creation

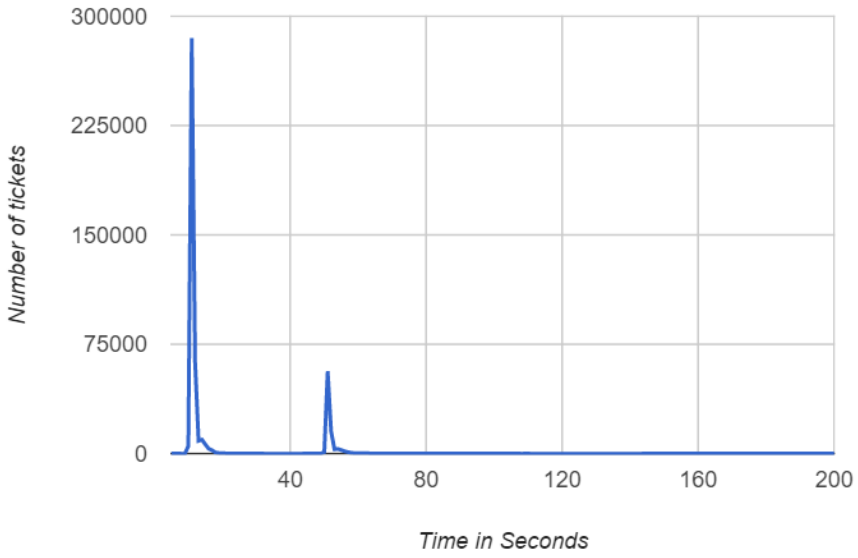

Figure 7.8: Time to ticket creation

Time to ticket creation is a very important metric for consideration. This is a clear performance indicator and efficiency of the system can be monitored. This is what ensures that the CCR can have an extremely quick FRT. The graph shown in Figure 7.8 shows that majority of the tickets are created under 11 seconds from the time a Tweet is made by a user. Another spike is observed at around 50 seconds. This spike exists because in few situations all the Java workers are busy and need to wait till the resources become free. More than 98% of tickets are created under a minute.

7.2.7 Average FRT and Average Resolution Time

Figure 7.9 indicates the average FRT and average resolution time of the companies with most tickets raised during which this metric has been evaluated. All the branches have a very good average FRT that is under an hour except for the case of IRCTC. Most of a company's take a lot of time to resolve the issue. Simple issues are resolved immediately. The more complex problems take significant amount of time and needs significant amount of dialog with the customer.

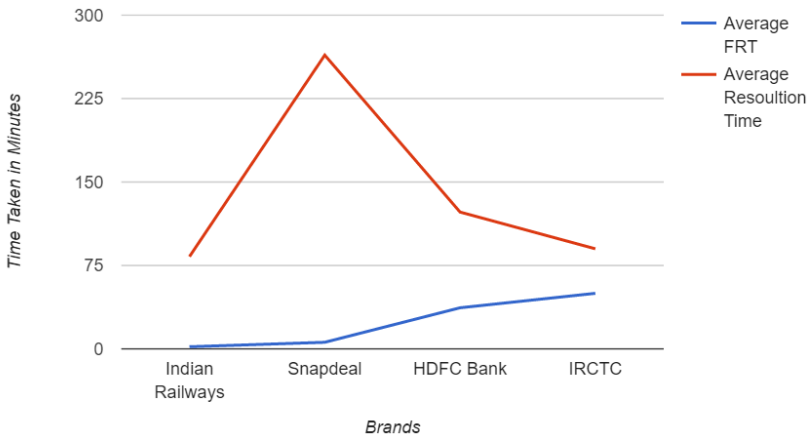

Figure 7.9: Average FRT and average resolution time

7.3 Inference from Result

Linear SVM has the highest accuracy but however pair wise three class was chosen since it can accommodate more emotions than two class. The system was further improved and extended to classify the chat into four categories.

Ticket count based on priority and sentiment was analysed. The sentiment engine has lot of scope for further improvement. The number of new users indicates the increasing use of social media and all the more reason for companies to adopt SCRM. The number of tickets generated is steadily growing suggesting the system must be highly scalable and more resources have to be allocated to properly run the system. A few tickets are deleted every now and then and not much can be done about it.

Majority of tickets are created under a minute it has been published and adhering to the specifications indicated in chapter 3. Average FRT and average response time gives an indication of the performance of the CCR.

Chapter 8

CONCLUSION

The world is developing very quickly. More and more interactions are becoming virtual. In such a situation, there is growing social presence of brands in the various social media sites. It would be foolish if the companies are not taking advantage of social media to fuel their brand popularity. Another aspect of social media is that it also empowers the customer also. Now the customer is in control they pick the method to communicate with the companies. Social media is interestingly being used by customers all over the world to show their dissent or dissatisfaction with any of the products or service they have used or come across. A single post by an angry customer can easily go viral on social media and create havoc on social media and completely destroy the company has built over time.

Until recently, there were no means to track and handle the complaints of customers in social media platforms. Thus CRM had to evolve to the new and improved Social CRM to handle the changing Web 2.0. Social CRM is catching fire in recent times and many companies are having exclusive CRM teams to handle issues raised on social media.

The system that has been implemented provides SCRM tool to handle the issues raised from Twitter. The system is able to capture and raise tickets from tweets and direct message to the brands. A mechanism has been provided to support both automatic ticket generation and manual ticket generation. The architecture and design of the system are one of the key highlights of the system. Ample time has been spent on design to ensure that system works smoothly and every possible issue is properly handled. With a good first response time and quick problem resolution time, companies are able to satisfy many customers and improve brand image and customer retention.

The process of computationally identifying and categorizing opinions expressed in a piece of text, especially in order to determine whether the writer's attitude towards a particular topic, product, etc. is positive, negative, or neutral is known as sentiment analysis. Various models were tested before the final selection of the sentiment analysis model . Incorporation of the concept of stop words, root words and identification of negative words

improved accuracy and reduced misclassification. Ruled based systems worked well only a few set of words and not easily scalable. After trying multiple sentiment analysis models, the highest accuracy with a linear SVM with 2 classes. Linear SVM with 2 classes had the highest accuracy of around 85% and was further developed and extended to Linear SVM with multiple classes.

8.1 Limitations of the Project

The project has quite a few flaws in it despite the fact that a lot of care was taken to cover all possible cases during the design and implementation phase. These limitations are mostly edge cases and occur in rare situations. A few of the key limitations have been listed in this section.

GNIP captures the tweets relevant to a brand purely based on the rules specified in the GNIP console. It is not possible to come up with a set of rules that covers every possible type of mention or reference to a particular brand.

The Twitter users are very sarcastic at times and make references to the different brands. A ticket is raised by the system as the mechanism in place is not able to identify whether the Twitter user is serious or just poking fun at the brands in his/her tweet.

The sentiment engine in place uses a standard linear two-class support vector machine with accuracy up to 85%. There is a possibility that few tweets maybe categorized in the region of the hyperplane that separates the positive and negative sentiment. Threshold has been set in the border region and those tweets are categorized as neutral tweets. These tweets have to be checked manually by the CCR and tickets have to be raised if necessary.

The system highly dependent on 3rd party services like GNIP, Twitter API, Twitter, and Twitter4j. Glitches or failures in these services have a direct impact on the performance of the system.

8.2 Future Enhancement

The system is not perfect and has scope for ample improvement in the near future. Few possible enhancements are:

- Using a better sentiment engine for classification of tweets to improve accuracy.
- Improve user interface for a better user experience.
- The system in place is very heavy in the sense, it requires a lot of resource for proper functioning. It is desirable to develop a lightweight version of the same to minimize data and resource consumption.
- Incorporate more metrics into the system to measure the performance of the CCR.
- Reduce the dependence on 3rd party services.
- For performing a single task, multiple API calls are being made. A different architecture can be put to use to execute the same task.
- Functionality can be implemented to capture and process complaints raised in languages other than English.
- Sentiment engine must be tweaked such that it can support slang, sarcastic language as well as words or phrases from other languages along with English.

REFERENCES

[1] Zeynab Soltani and Nima Jafari Navimipour, "Customer relationship management mechanisms: A systematic review of the state of the art literature and recommendations for future research", *Computers in Human Behavior 61*, pages 667-688, 2016

[2] José Martins et al , "Social networks sites adoption at firm level : A literature review", Information *Systems and Technologies (CISTI), 9th Iberian Conference*, 2014

[3] Fatima Zohra Ennaji et al, "Social intelligence framework: Extracting and analyzing opinions for social CRM", IEEE/ACS 12th International Conference of Computer Systems and Applications (AICCSA), 2015

[4] Marcel Rosenberger, "Social Customer Relationship Management: An Architectural Exploration of the Components", *IFIP International Federation for Information Processing,* 2015

[5] Nyoman Karna et al, "Social CRM using web mining", *International Conference on Information Technology Systems and Innovation (ICITSI)*, 2014

[6] Abid Hussain et al, "Social Data Analytics Tool (SODATO)", *International Conference on Design Science Research in Information Systems*, 2014

[7] Surya Nepal et al, "Special Issue on Clouds for Social Computing", *IEEE Transactions on Services Computing, Vol 3 Issue 3*

[8] Marcel Rosenberger, "Conceptualising and Exploring User Activities in Social Media", *Open and Big Data Management and Innovation*, pages 107-118

[9] Priyanga Gunarathne , Huaxia Rui and Avi Seidmann, "Customer Service on Social Media: The Effect of Customer Popularity and Sentiment on Airline Response", *48th Hawaii International Conference on System Sciences*, 2015

[10] Ifeoma Adaji and Julita Vassileva, "Social Media Strategy in the Italian Fashion Industry: A New Model of Analysis", *Workshop on Business Models and ICT Technologies for the Fashion Supply Chain*, 2016

[11] Niels Buus Lassen et al, "Predicting iPhone Sales from iPhone Tweets" , *IEEE 18th International Enterprise Distributed Object Computing Conference*, 2014

[12] Graeme Shanks et al, "Social Media Analytics and Business Value: A Theoretical Framework and Case Study", *47th Hawaii International Conference on System Sciences*, 2014

[13] Hyung Jun Ahn, "Identification of the factors that affect the user reaction to posts on Facebook brand pages", *International Conference on Computer and Computational Sciences (ICCCS)*, 2015

[14] Babak Abedin et al, "Relationship Development with Customers on Facebook: A Critical Success Factors Model", *48th Hawaii International Conference on System Sciences*, 2015

[15] David Alfred Ostrowski, "Using latent dirichlet allocation for topic modelling in twitter", *Proceedings of the IEEE 9th International Conference on Semantic Computing (IEEE ICSC),* 2015

[16] Ghazaleh Babanejaddehaki et al, "Topic modeling for social media content: A practical approach", *3rd International Conference on Computer and Information Sciences (ICCOINS),* 2016

[17] Marcel Rosenberger et al, "Integrating data from user activities of social networks into public administrations", *Information Systems Frontiers, Volume 19, Issue 2, pp 253–266,* April 2017

[18] A. Marzak et al, "Review current CRM architectures and introducing new adapted architecture to Big Data", *International Conference on Computing, Communication and Security (ICCCS),* 2015

[19] Kevin Andrey Rosales Ferreira et al, "Analysis of the behavior of customers in the social networks using data mining techniques", *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM),* 2016

[20] David Alfred Ostrowski, "An approximation of betweenness centrality for Social Networks", *Proceedings of the IEEE 9th International Conference on Semantic Computing (IEEE ICSC),* 2015

[21] Jitendra Ajmera et al, "A CRM system for social media: challenges and experiences", *Proccedings of 22nd International Conference on World Wide Web, Pages 49-58,* 2013

[22] Stephan Wan et al, "Social Media Data Aggregation and Mining for Internet-Scale Customer Relationship Management", *IEEE International Conference on Information Reuse and Integration,* 2015

[23] Kevin J. Trainor, "Social media technology usage and customer relationship performance: A capabilities-based examination of social CRM", *Journal of Business Research, Vol 67, Issue 6,* Pages 1201–1208, June 2014

[24] Jyotirmoyee Bhattacharjya et al, "Building Business Resilience with Social Media in B2B Environments: The Emergence of Responsive Customer Relationship Management Processes on Twitter", *IFIP Advances in Information and Communication Technology Vol 463,* 2015

[25] Ulrike Baumöl et al, "Dynamics of customer interaction on social media platforms", *Electronic Markets, Vol 26, Issue 3,* Pages 199-202, 2016

[26] Siaw Ling Lo et al, "Use of a High-Value Social Audience Index for Target Audience Identification on Twitter", *ACALCI: Artificial Life and Computational Intelligence,* Pages 323-336, 2015

[27] Yoav Achiam et al, "Why not scale free? Simulating company ego networks on Twitter", *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM),* 2016

[28] S. S. Askool et al, "Scoping Study to Identify Factors Influencing the Acceptance of Social CRM", *Proceedings of the IEEE ICMIT,* 2010

[29] Sanaa Askool and Keiichi Nakata, "A conceptual model for acceptance of social CRM systems based on a scoping study", *AI & Soc 26,* Pages 205–22, 2011

[30] Chunjing Xiao et al, "Measuring User influence based on multiple metrics on YouTube", *Seventh International Symposium on Parallel Architectures, Algorithms and Programming*, 2015

[31] Edward C. Malthouse et al, "Managing Customer Relationships in the Social Media Era: Introducing the Social CRM House", Journal of Interactive Marketing 27 , Pages 270–280, 2013

[32] Julia Heidemann et al, "Online social networks: A survey of a global phenomenon", *Computer Networks 56,* Pages 3866–3878, 2012

[33] Seyedreza Mousavi et al, "The Key to Social Media Implementation: Bridging Customer Relationship Management to Social Media", *46th Hawaii International Conference on System Sciences*, 2013

[34] Imre Petkovic, "CRM in Cloud", *IEEE 8th International Symposium on Intelligent Systems and Informatics* , 2010

[35] Eun Go et al, "But not all social media are the same: Analyzing organizations' social media usage patterns", *Telematics and Informatics 33*, Pages 176–186, 2016

[36] Dushyant Kumar Sharma, "A Social Network Perspective From CRM", *IEEE International Conference on Granular Computing*, 2011

[37] David Alfred Ostrowski, "Feature Selection for Twitter Classification", *IEEE International Conference on Semantic Computing*, 2014

[38] Hanno Zwikstra et al, "Connecting Customer Relationship Management Systems to Social Networks", *7th International Conference on KMO, AISC 172,*Pages 389–400

[39] Adam Czyszczoń and Aleksander Zgrzywa, "Automatic Customer Segmentation for Social CRM Systems", *CCIS 370*, Pages 552–561, 2013

[40] Dario Liberona et al, "Customer Knowledge Management in the Age of Social Networks", *7th International Conference on KMO, AISC 172,* Pages 353–364

[41] David Alfred Ostrowski, "Semantic Filtering in Social Media for Trend Modeling", *IEEE Seventh International Conference on Semantic Computing*, 2013

[42] Minbo Li et al, "A Social Collaboration Platform for Enterprise Social Networking", *IEEE 16th International Conference on Computer Supported Cooperative Work in Design*, 2012

[43] Meyliana, Budiardjo and Eko K, "The Comparison of CRM Model : a Baseline to Create Enterprise Architecture for Social CRM", *Proceedings of the First International Conference 479 on Advanced Data and Information Engineering (DaEng)*, 2013

[44] Neil Woodcock et al, "The evolving data architecture of social customer relationship management", *Journal of Direct, Data and Digital Marketing Practice, Vol.12 No.3*, Pages 249–266, 2011

[45] Niels Buus Lassen et al, "Predicting iPhone Sales from iPhone Tweets", *IEEE 18th International Enterprise Distributed Object Computing Conference*, 2014

[46] Sergio Orenga-Roglá et al, "Social customer relationship management: taking advantage of Web 2.0 and Big Data technologies", *SpringerPlus 5:1462*, 2016

[47] Lawrence Ang, "Community relationship management and social media", *Journal of Database Marketing &Customer Strategy Management*, Pages 31-38, 2011

[48] Olaf Acker et al, "Social CRM: How companies can link into the social web of consumers", *Journal of Direct, Data and Digital Marketing Practice 13*,Pages 3 – 10, 2011

[49] Nicolas Wittkuhn et al, "Social CRM Performance Dimensions: A Resource-based View and Dynamic Capabilities Perspective", *28th Bled eConference* , 2015

[50] Reinhard Jung et al, "Towards Social CRM – Scoping the Concept and Guiding Research", *26th Bled eConference*, 2013

APPENDIX

Figure A.1 shows the Brand home page. This is the first page that will be displayed when a CCR logs into their accounts. The GUI displays the number of tickets in each status. Some key performance metrics are showed below this section followed by customer happiness index.

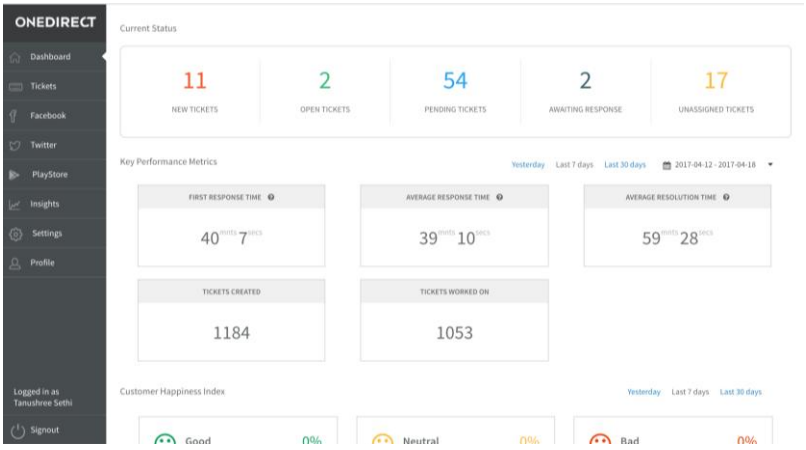

Figure A.1: Brand home page

Figure A.2 shows the Tickets page. The dashboard shows all the tickets assigned to a particular CCR. The ticket number, priority and other fields are displayed. A number of filters have been provided to help the CCR easily search for a particular ticket. All tickets assigned to a CCR can be reached by clicking on the corresponding ticket.

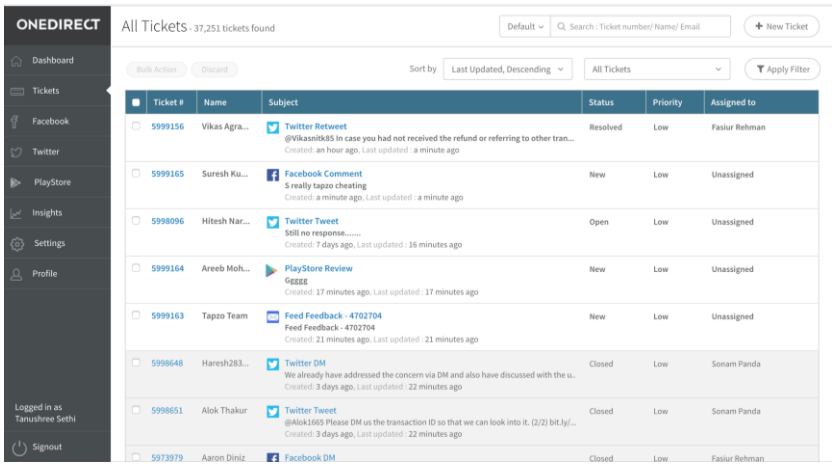

Figure A.2: Tickets page

Figure A.3 shows the Ticket details page. The following section is displayed when a particular ticket is selected from the Tickets page. On the right hand side, details about the customer, ticket, ticket history and other fields are shown. The ticket number is displayed in the middle of the page. The DM section is shown below it followed by the tweet and reply section. In the tweet and reply section, the tweets by the user is seen in blue while that of the CCR is seen with a white background.

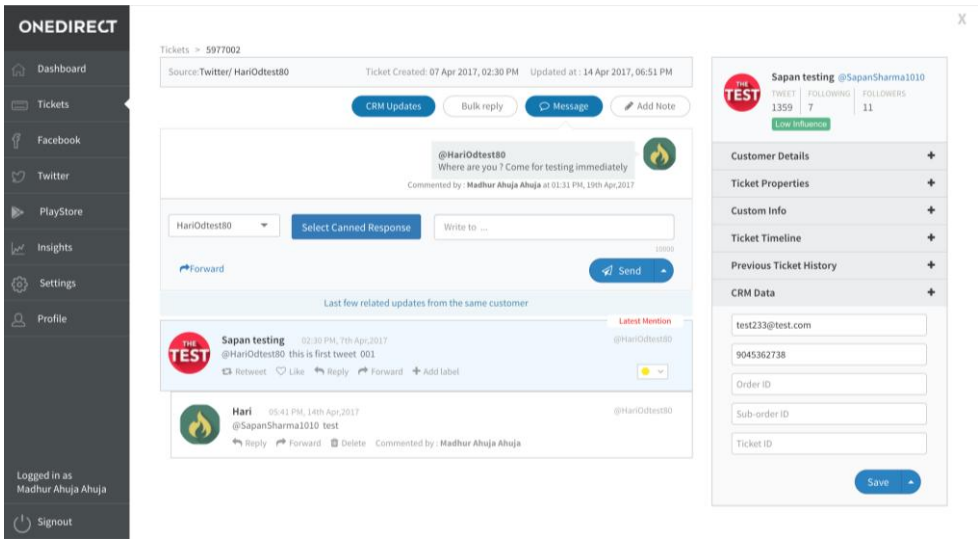

Figure A.3: Ticket details page

Figure A.4 shows the Insights page which shows the analytics and reporting of the CCR. Average response time, average resolution time and other metrics are shown to help the CCR track their performance.

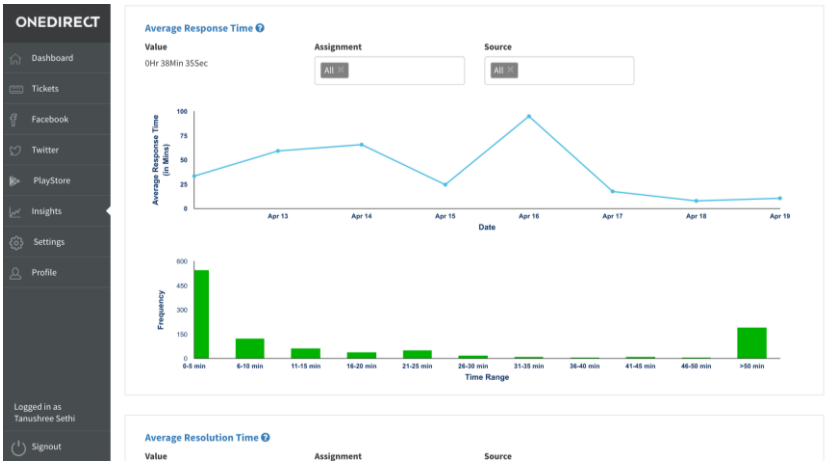

Figure A.4: Insights page