

Custom Application Development Of BMRCL In Cloud Environment: Using Salesforce CRM

Sunil B S¹, Prof. Madhuri J²

Department of Computer Science and Engineering, Bangalore Institute of Technology.

Abstract

In today's world, providing good customer satisfaction is very important for the companies. For this purpose many companies are turning their face towards customer relationship management (CRM). The companies are making the relationship between the customer and the company stronger by leveraging CRM. In this paper we propose a custom application for Bangalore Metro Rail Corporation Limited (BMRCL) with which we can support some common functionalities of original application. Our main focus is to utilize the CRM features provided by Salesforce. As the part of the CRM we propose an intelligent chatbot through which users can make use of features like purchasing a smart card, checking their balance, getting answer to common FAQ's and checking roadmap. The application we build follows the Model View Controller (MVC) architecture. For building the view we make use of Visualforce, the model is made up Salesforce records and the control is written in apex programming language supported by Salesforce.

Keywords: CRM, Salesforce, Einstein bots, Case management, Apex, Visualforce.

1. Introduction

The Salesforce is a cloud provider who leads in providing CRM (customer relationship management) products. It supports three clouds namely the service cloud, marketing cloud and sales cloud. We make use of the service cloud which the Salesforce offers. The original BMRCL does not focus on the customer satisfaction and does not proactively support their customer on their needs when they have some cases or queries to be solved. The custom application we develop will overcome the drawback of the original application which falls behind in providing customer satisfaction. Our system would provide functions to the user like purchasing a smart card, checking for balance, checking for road maps and so on. This feature can be accessed through an intelligent chatbot. Appropriate dialogs for this functionalities are provided which the customer can make use of. Security for the user record at the top most level is done through taking user credentials like email, user id and OTP. The chatbot is build based on natural language processing. The developer may not worry about building the code for this bot functioning rather writing control statement is required. Apex and flows can be used to write the control as to what process should be followed when a particular dialog is selected. The bot builder is used to write logic to the bot.

Along with this case management is proposed where the cases raised by the user through email of the customer are converted to cases in the backend by Salesforce. This happens when a user sends a mail to a designated mail address which is used by the system we have built. This are carried out by auto response, auto assignment rules which we would have defined. Through the use of Omni channel or live channel we will assign the cases to appropriate queues. The cases will be taken by the agents belonging to this queues. The agent can continue conversation with the user through the case itself as Salesforce offers this feature. The user will receive an acknowledgement as to his registered case with details containing information about the case number, subject of the case along with some other

2. Literature Survey

Jigar Patel, Ankitchauhan et al. [1] All the existing concepts about software, how the applications are built and handled and the way products are built which will change all the existing views on software. Many of the IT companies are moving their computation on to the cloud and when it is about custom relation management they are turning their face towards Salesforce. The prosed work tries to show the salient features of Salesforce and in what way we can make use of Salesforce platform to change the existing business to help customers have a better experience when they avail the services from them. Krutarth Soni et al. [2] This paper talks about security offered by Salesforce in detail. The authentication are handled through two factor authentication and single sign on authentication. User level security is provided through permission settings, profile access restrictions. The row level security is provided with field level security on both standard and custom objects. Salesforce offers something known as role hierarchy which will create a hierarchy of roles which determine who sees what type of records and who has default access to records. To better handle the authentication Salesforce keeps asking to change passwords every ninety days. Neroida Selimi, Marika Apostolova Trpkovska et al. [3] The main objective of this paper is to utilize the CRM tool of the Salesforce and come up with a model for CRM which would increase the productivity of the institute. The CRM prototype developed would benefit the SEEU institute to have a long term relationship with the students of its organization. This approach is not only followed for students but staff are also in picture for satisfaction. Am Rahman et al. [4] this paper points out the problems while designing the chatbot and the difficulty one faces when programming. Since bot is designed through artificial intelligence the implementation requires minimal to no code change. One big advantage of chatbot is that it does not require user to install software packages to

start it running. The chat bot usage is often very simple to use with simple interface. The conversation messages between the user and the bot are transferred immediately. If the omni channel is supported by chatbot only then the end user can see the status of the agent otherwise the end user will not have knowledge about the agent's status. The paper also gives an overview of cloud-based chatbot technologies along with programming of chatbot and difficulty of programming in today's and future era of chatbot. Yuxion chai et al. [5] Based on the censorship outcome in this paper utterance system will filter out the response of chatbot and censorship results are given based on the feedback of the bot. The architecture which is proposed is a self-purifying one where we during bot training give penalty if it captures offensive words and we provide rewards to the bot if it does not do that. But the limitation here is that censorship needs to be updated regularly. However this paper does not explain about FAQ and intent sets of bots. Aneta Poniszewska Maranda, Radosław Matusiak et al. [6] This paper talks about how the organizing tournaments, popularization of mentioned disciplines is done on the cloud by having an application designed by the Polish Billiards and Snooker Association, further as (PBSA). It also talks about how Salesforce provides real-time service systems and supports them. The access for this cloud application and usage of it is provided only to the employees from the snooker organization. Nor Alina Binti Ismail et al. [7] For establishing a long term relationship with customers it is important to have a virtual community in today's world. Once the post purchasing of tickets was done by the customer it was found out that the satisfaction for the customer was about $\beta = 0.99$. This indicates that the airline service have been providing good CRM to their customers and hence the number of users using the service of the airlines are increasing. This shows the research questions asked where appropriate and on point and where meeting the customer requirement standard.

3. Related Work

The related work mainly makes use of trailheads which helps one to learn about Salesforce and its features in detail. Trailhead is the go to place if one wishes to learn about Salesforce and its cloud usages. It consists of various trails and you get points on finishing tasks. There are various trails to learn about individual features. For example if one chooses Visualforce trail then we get to learn about its structure, its usage, how the controller is written and how the browser renders it and so on.

APEX: Apex is language developed by the Salesforce.com which they own. Strongly typed, object-oriented programming language is a feature of Apex and it will help the developers to transaction control statements and also executing the flows in the Salesforce platform server by making calls to the Force.com API in conjunction. Apex has a Java-like syntax and will work like database stored procedures. Inclusion of most system events with the business logic, including the button clicks, updating related records and Visualforce pages are enabled to the developers. Apex code can be started by Web service requests and from triggers on objects. In the Unlimited Edition, Enterprise Edition, Performance Edition, and the developer Edition Apex is included.

Features of Apex as a Language are:

Integrated: Apex includes built in support for DML operations like INSERT, UPDATE, DELETE and also DML Exception handling. Apex also supports for inline SOQL and SOSL query handling which returns the set of sObject records.

Java like syntax and easy to use: Apex is easy to use as it uses the syntax like Java. For example, variable declaration, loop syntax and conditional statements are similar to Java.

Strongly Integrated With Data: Apex is designed by keeping in mind that it can execute multiple queries, DML statements parallel and also it is data focused. Apex issues multiple transaction statements on Database.

Strongly Typed: Apex is a strongly typed language. Apex uses direct reference to schema objects like sObject (Salesforce object) and any invalid reference quickly fails if it is deleted or if is of wrong data type.

Multitenant Environment: Apex runs in a multitenant environment. Also Apex runtime engine is structured to guard closely against runaway code, preventing it from monopolizing shared resources. Any code that violates limits fails with simple and clear error messages.

Upgrades Automatically: Apex is upgraded as part of Salesforce with new releases. The orgs don't have to upgrade it manually.

Easy Testing: Apex provides built-in support for unit test creation and execution, including test results that tell how much code is covered, and which parts of your code can be better structured.

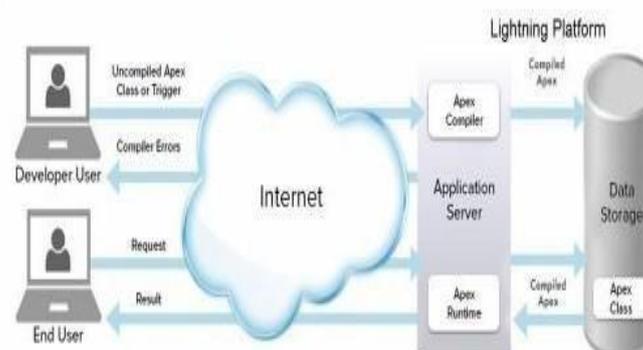


Figure 1: Flow of Apex

PERMISSION SET:

A permission is nothing but collection of settings and permissions that are given to users. The permission set determines what all settings or details the user has access into using or reading it. This is particularly useful for security purposes. We can determine the profiles having access to particular data or setting by adding them with specific permission sets. Hence the access of data is restricted. For example, consider the user with sales profile. He will have access to read or write opportunities but not all profiles with sales access will be able to have transfer opportunities. Hence we can restrict specific accesses through profile sets.

CASE:

Case is nothing but a question or a query a customer has probed and wishes get it resolved. The way in which cases are handled are different based on what sort of case it is. Suppose it is a case related to support then support agent picks it up and if it is a sales case then the sales agent picks it up and will try to resolve it. If the case is answered and resolved the customers will become happy and hence we provide them a better service.

CHAT TRANSCRIPTS:

The discussion among client and Agents can be logged through visit transcripts and Salesforce mechanized this procedure where once the talk between agent and client closes chat transcript is generated. After the visit window closes, the record is held as a transcript record in database. The transcript can be related with a case, account, lead and so on. If required we can connect them with other custom articles too. The chat transcript record holds data with respect to what extent the visit was open, from which city or nation the talk has begun, which route has been followed and others. This data can be utilized to upgrade the exhibition of how we handle our business in giving best client assistance to clients. The record of chat transcript can be queried through SOQL or Apex. Through the bot builder setup we can set up the chat transcript.

SALESFORCE OBJECTS:

They are the building blocks of Salesforce records. There are two types of objects in Salesforce namely, custom objects and Salesforce standard objects. We can add additional field to standard objects according to our requirements. The custom objects are built from scratch and it can hold various types of data like pick list, text, number and more. If we want to check for a particular field to have populated with a field before inserting the record into the Salesforce database or we wish to do some similar validation then we can go for validation rules. Not only through validation rules but we can also go for workflow rules which could be designed to have various activities or actions done like a trigger. The records of these objects can be called and shown in Visualforce, or through a trigger be modified or write an apex to have a control functionality. Specific fields can be made accessible to some profiles and some may be restricted through the usage of FLS and permission sets. Page layouts can be created and the records can be displayed in a customizable way to users. The editable, read and write permissions can be set very well in Salesforce.

KNOWLEDGE ARTICLES:

Knowledge articles help in the making of information base which contain reports of data. Administration specialists with more experience and composing articles are left to the inner authors. The articles are then distributed and can be utilized inside or remotely in a scope of channels. One can distribute articles in client and accomplice networks and open sites or offer articles in social posts and messages. We can have a command over what we can distribute and share dependent on the article page designs, client profiles, activities, and different settings. One can utilize Knowledge in Salesforce Classic just as Lightning Experience. Organizations new to Knowledge use Lightning Knowledge, which is commonly accessible. In the event that your organization as of now utilizes Knowledge in Salesforce Classic, consider utilizing the Lightning Knowledge Migration Tool and changing to Lightning Knowledge.

OMNI CHANNEL:

If you desire to not to code but still wish to have a customizable, flexible and one which can configure then we can go for omni-channel. Omni channel helps in handling more important things and give priority to them. We hence can restrict how much work an agent can handle which helps in even distribution of work load. We can set the agents to take up as much work which they can handle by configuring the omni-channel. We can split up the work among the agents that is for example we can have one set of agents handling queries regarding opportunities and other set handling sales. The functionality of routing work to the queues is automated by Salesforce which we can avail. Effort and brainier can be saved by not making the agents to choose the job instead by having these queue structure in place we can automate which agent gets assigned which job. The rate at which agents respond to the customers can increase exponentially because they now get jobs auto-assigned and hence can work at a faster rate.

Some of the tools required to achieve our goals are:

EINSTEIN BOT BUILDER:

If we wish to have a point and click setup tool then Einstein bot builder is where we need to get. Builder also makes us to build dialogs in a bot. Bot builder helps us to have different types of tasks that can be configured in the dialog we create. One can Question and gather information which we desire and can return the output to the chat window of the user after sending response to the users. To be more specific, for companies to deliver good customer service we can use.

AI backed up bots in the ways as follows:

Reducing customer waiting time: Instead of waiting for an email, making a phone call, or to an agent making a response from another channel, we can provide immediate answer to customers and also we can decrease the time the customer waits for the

responses (information such as getting to know the status of the order, knowing the store hours, fetching the location of the customers etc.,).

Providing agents with leads: The more qualified sales representative plays a key role where we can provide him lead which we obtain with the help of Ai Chat bots that will help in opening up conversations that the sales representative needs to follow up. We ask questions which are related like a series and gather relevant information like getting to know the email address, name and many more information. This is done by having initial support interactions with prospective customer and the customer.

Resolving support cases: We can have a much faster support by resolving support cases at a much faster rate. This is done by the bot that answers straight forward questions and finally as result the customers will be satisfied more.

Handling efficient redirects for customer inquiries: We can quickly place customers to the dialogs they are looking for through intents and we can hence greet them appropriately for example we can give them a welcome message with the brand of the company.

FLOW BUILDER:

It's a tool for building flows in Salesforce. A flow is nothing but a series of actions that would take place. Some of its features include:

Button Bar:

Here we manage the flow as it is build. The Run icon runs the most recent saved version of the flow that we have opened. If the flow contains Sub-flow elements, the active version of its referenced flow is executed. If the referenced flow has no active version, then the sub-flow element runs the latest version of its referenced flow. Debug lets you enter values for the flow's input variables and display debug details while running the flow.

To the left of the Run button, you will see:

1. A pill identifying whether the version is active or not
2. How long ago the version was saved
3. A warning icon and an error icon, if any apply. To check the warnings associated with the specific versions or errors associated we can click the respective icon.

Canvas:

To build the flow by adding elements we can make use of the canvas. We can also add elements to the canvas and associate them together, it provides us a visual diagram of the flow we have built.

Toolbox:

From the Elements tab of the toolbox we can add new elements, like Screen and Create Records, to our flow. From the Manager tab of the toolbox we can create variables, stages, choices, and other resources to use in your flow. Or view a list of all elements and resources that we have added to the flow.

4. Design Issues and Goals

The BMRCL has the following drawbacks:

- Bengaluru metro (BMRCL) is a fast travel that serves the city of Bengaluru. The Bengaluru Metro has a web application permits clients to buy and purchase smart cards, find close by metro stations and furthermore gives data identified with stopping, train recurrence, course guide, and passage subtleties.
- A significant disadvantage of the BMRCL application is its CRM which does not have any connection with the clients.
- There is no legitimate case handling system in the BMRCL, where the client can raise a case and the BMRCL operators to see it.
- The clients can't collaborate with the operators continuously with chatbot.

The objective of this work is to give CRM highlights to the clients of BMRCL application. Here, we do as such by presenting case management and giving a chatbot to the client to get service by asking questions and connecting to live agent if required. The case management is accomplished by different automated rules. Through case management all the issues of the clients can be attended. As a result of a chatbot (Einstein bot here) allows the client to toss their questions. It uses concepts of intent sets to increase the performance of the chatbot. Using intent sets the intent of customer can be predicted and hence reduction of his dialog clicks decreases.

5. Proposed System

With this work the end user can make use of functionalities like purchase smart card, Recharge smart card, check roadmap, check balance etc.,. As soon as the user tries to access these functionalities the request is hit on the Salesforce servers and the chatbot which is running in the Salesforce server is invoked and communicates appropriately with the user. The Salesforce provides security through the two way authentication method. The bots in turn may invoke appropriate apex methods/flows based on the functionality requested. That is for example when the purchase smart card dialog is chosen by the user, purchase card method is called upon to handle user request. The BMRCL website is completely designed using Visualforce language which is mark-up

language of Salesforce. The code for chat bot is embedded in its script. The apex method/flows that were invoked may access data from the records that are part of the BMRCL. The architecture of the proposed system is given by figure 2.

The Salesforce provides through Salesforce predefined objects and also custom system objects were also created. For example custom object called smart_card__c was created to hold smart card records. They are accessed by apex methods through SOQL calls. The apex runs on the cloud and provides the result back to the integrated system through the cloud itself. The BMRCL website we have created tries to replicate the original metro app. Also along with it case management is provided where the user without using the website can directly create a case which can be handled by the BMRCL agents by sending a mail to a designated mail address created by us. In response to the created case we send an email through the auto-response rule we have set up which contains information regarding the case.

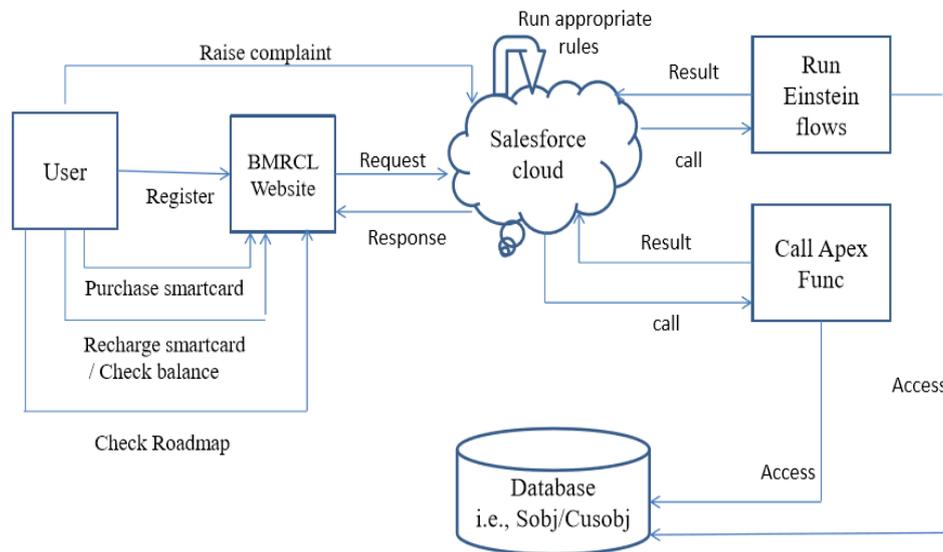


Figure 2: Architecture of proposed system

Some of the important modules in our custom application are:

User Interface:

The user interface for our system is the BMRCL application which is deployed on the Salesforce cloud.

The user can make use of this application to raise complaints, check metro future plans, recharge smart card , purchase smart cards, chat with an agent/bot etc.,

The application will be build using the Visualforce framework and using apex language to implement business logic.

Visualforce:

- One of the web development framework is Visualforce and it allows the users to create better custom UI for desktop and mobile apps that can be hosted on the Lightning Platform. Visualforce enables developers to increase Salesforce’s built-in features, replace them with new functionality, and build completely new applications.
- If somebody accesses a page the server performs any data processing required by the page, renders the page into HTML and returns the results to the browser of the user for displaying the contents. . It can also be joined along with any JavaScript framework or standard web technology which will provide for a more animated and richer UI.
- Visualforce app development is similar to anyone who has built web apps. Developers can create Visualforce pages by composing the components, HTML, and Visualforce will allow for more number of styling elements in their web page. Every page is accessible by a unique uniform resource locator (URL).

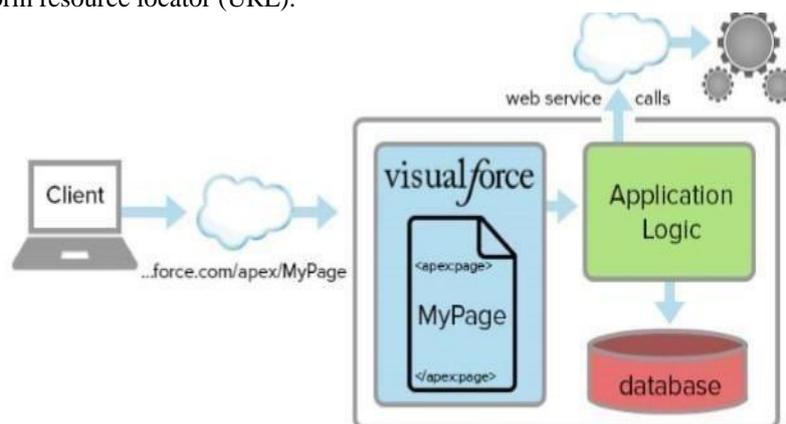


Figure 3: Principal of working of visualforce

Einstein Chatbot:

Einstein chatbot is built on the basis of natural language processing and they are used to have a human like communication with the end user. The type of data exchanged between the user and the bot are mostly text messages. These text messages can be used to train the bot as well. Through the use of intents, utterances are used to train the bot which will form the test data. The result is that the common questions the user might look to get sorted by the bot is found out. Hence communication to an extent can be automated but still if the customer wishes we can transfer complex questions to the live agent.

Some features of the chatbot are:

Dialog: Dialogs are conversation snippets that control what a bot can do. Each dialog includes a dialog intent, which is optionally available for NLU training to understand different kinds of customer responses. During a conversation with a customer, a bot moves between several different dialogs. Each dialog handles a portion of the conversation. For example, Welcome, Main Menu, Order Status, Location and Hours, and Transfer to Agent are individual dialogs that a customer might experience as part of a single conversation with a bot.

Dialog Intent: Dialog intents are the customer's reasons for interacting with your bot. For example, buying a solar panel, returning a solar panel, getting store hours. Dialog intents are actions and generally start with verbs, such as buy, return, find, look up, cancel, edit, etc.

Entity: Entities are a type of data that you want to collect from a customer. Salesforce provides the following system entities: Text, Date Time, Date, Money, Number, Person, Location, Organization, Percent, Boolean, and Object (standard Salesforce or custom). You can create custom entities as needed.

Variable: A variable is a container that stores a specific piece of data collected from a customer. You must associate each variable with an entity. Since variables are containers of information, you can use them within dialog actions as both inputs and outputs and insert them as text in messages.

Raise Complaints:

Case management means organizing customer cases into one place and making sure they go to the right person, for the right answer, by the right time. Service Cloud does all that behind the scenes with automation tools. Service is easier, faster, and better with a little auto-magic. This can be achieved through queues, assignment rules, auto response rules and escalation rules. The users can put up cases through mail in the form of cases. The cases in the backend will be created because of the rules we have defined. The cases will be routed to the correct service agents who will look it up. These feature is not present in the original BMRCL application.

Recharge/ purchase smart card:

Recharging a smart card online is the one of the most important BMRCL feature. Our system implements it by making use of custom object that is related to the smart card associated with each user. Purchasing a smart card is another feature in the BMRCL application. He has to first validate his user credentials like Email, User Id and authenticate himself by entering OTP sent to his Email. The user smart card and his details are stored as user records. Not only recharge/purchase, our bot also provides the user to check balance, check FAQ and register as a BMRCL agent as well. The authentication process for the other functionalities is also same that we do it by getting mail and asking for OTP. The FAQ is provided with the creation of record types of knowledge articles. An article is itself a record which will hold information regarding the questions which would be usually asked by the users. In our case the articles would contain information to questions like how to check balance, how to connect to an live agent and so on.

Check future metro plans and road maps:

This is additional applications provided by the BMRCL application where user can check out the connectivity of the city by the metro.

User can check out future metro plans too that are going to be taken up.

Information for both of these are taken from the main website.

Case management:

Case management implies sorting out client cases into one spot and ensuring they go to the ideal individual, for the correct answer, by the perfect time. Service Cloud does all that in the background with rules we have defined. This can be accomplished through lines, task rules, automated message rules and heightening guidelines. We essentially utilize Assignment rule which causes us to allocate the approaching mail to one of the queue. Through automated message rule we send case creation email to the client. The sent email will have insights about the case like its case number, the explanation it was made. We utilize auto assignment rules for the automated message.

Through the use of task rules we have made the way toward doing out the work to a specialist robotized and now there is no requirement for an operator to get a case to work upon. What's more, through automated message rule we have given client a sense that we would be taking care of the case. This is significant with respect to tell the client that they are not ignored. The case subtleties that we have send to the client as a feature of the pre-programmed message can be utilized by them to gaze toward the case and know the current status of the case and who is working upon it. The Agents can communicate through sends with the client on the grounds that Salesforce gives the alternative of reacting to the client inside the case. At the point when the client sends an answer to a similar case another case won't be made rather the sends will be added to the first case email string.

Live chat:

We have actualized live visit with the utilization of omni-channel usage which is given by Salesforce. On the off chance that we structure various lines dependent on the kind of cases they handle, we can separate the work and can advance the work to the correct operator. This outcomes in a quicker exchanges and exertion taken by the Agents in the determination of the work is skipped. The specialist utilizes the nearness status to show whether he wish to take up the case or not. On the off chance that one specific specialist of the line isn't gotten an operator it is re-coordinated to another client. In the event that no operator is accessible, at that point a message is shown to the visit client expressing that no Agents are available.

Omni channel helps in taking care of progressively significant things and offer need to them. We consequently can confine how much work a specialist can deal with which helps in even dissemination of outstanding task at hand. We can set the specialists to take up as much work which they can deal with by arranging the omni-channel. We can throw up the work among the specialists that is for instance we can have one lot of operators taking care of inquiries with respect to circumstances and other set taking care of deals. With the utilization of essence arrangements it is workable for the operator to feature his status whether he is on the web, disconnected, away and others. In the event that none of the operators is accessible in the specific queue, at that point the client will get a message that no Agents are accessible from the bot when non agent is online.

6. Results

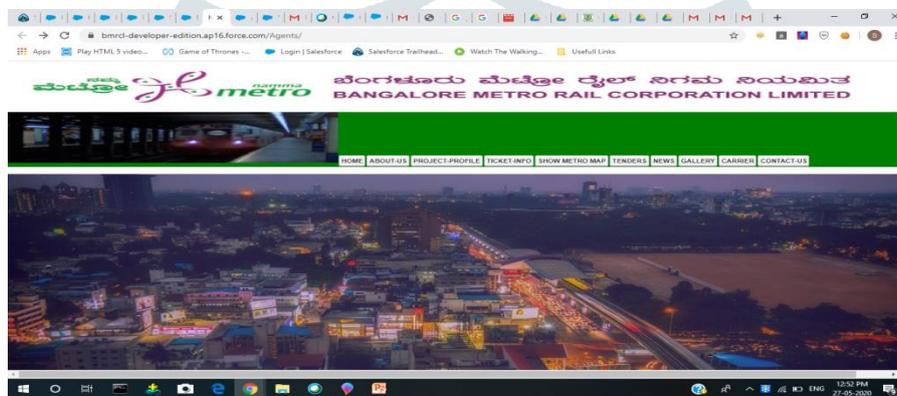


Figure 4: BMRCL Home Page

The site can be accessed by any user with the link <https://bmrcl-developer-edition.ap16.force.com/Agents/> but in-order to access the bot functionalities like recharging card etc., the user has to be an registered customer of our site. The figure 4 gives the BMRCL home page of which only contact us and check road map are clickable.

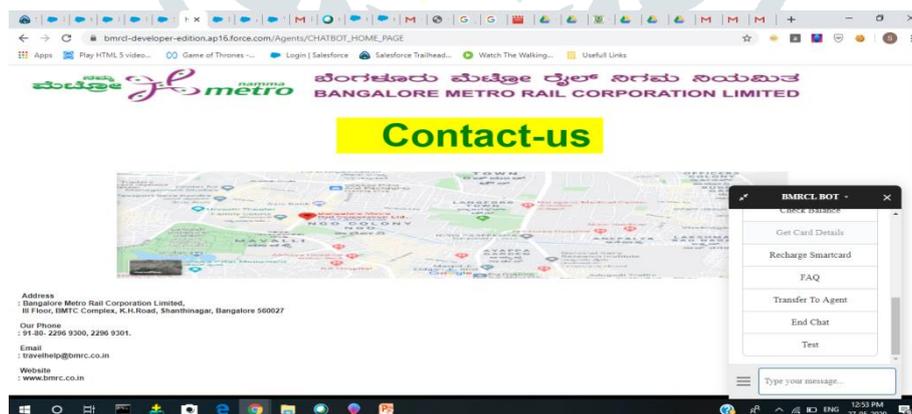


Figure 5: Chatbot after chat button opens

On clicking the bot button a bot window opens. Initially connecting ... is shown. On successful opening of the bot, the user is shown a welcome message. And then a main menu is displayed for the user to choose from as shown in the figure 5. On pressing on any of the dialog individual flow for appropriate dialog is run. They can select any of the dialog like purchase smart card, recharge smart card, check balance, FAQ, transfer to agent etc.,

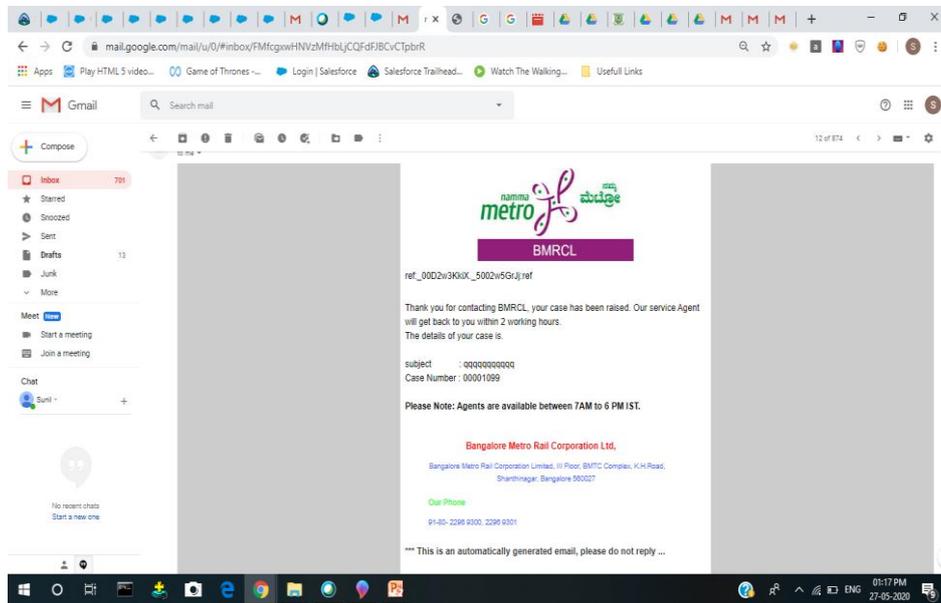


Figure 6: Case management handshake

bangaloremetrorailcorporation@1qpil77v4ag6wm93vx85nhs1yomcosrmd0totn5hgs1et0oqq3.2w-3kkixeas.ap16.case.Salesforce.com

Any user can send a mail to custom BMRCL site we have created and log a case using the email address provided above. Once a case is logged an handshake mail is sent to the user email which contains information related to the case like case number, subject etc.,

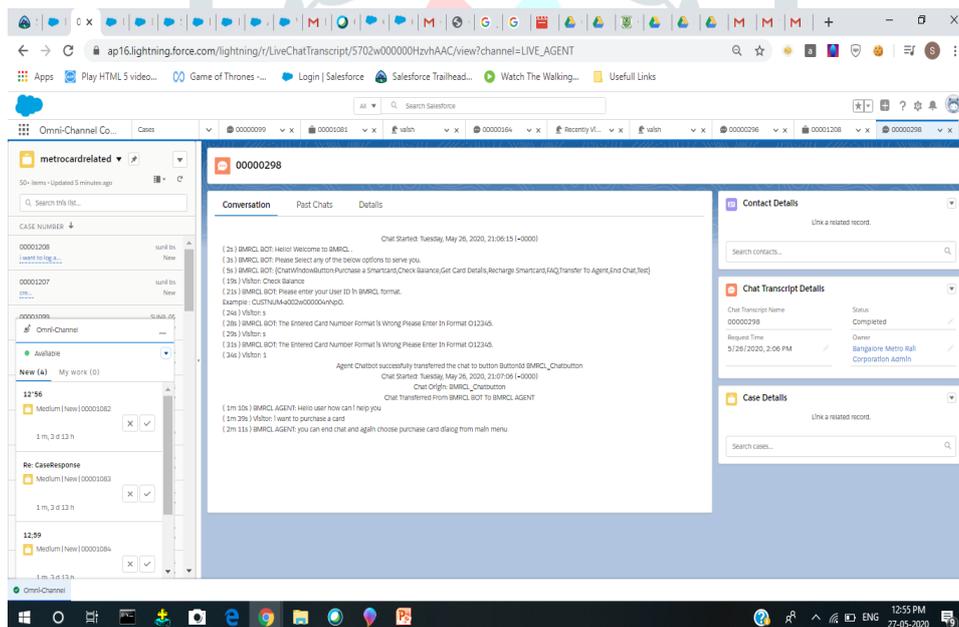


Figure 7: Live chat as seen from Agent perspective

Once the transfer is made from the bot by an user, the case shows up at the omni channel/live agent if he is online. If he is offline no agent is available is shown to the user.

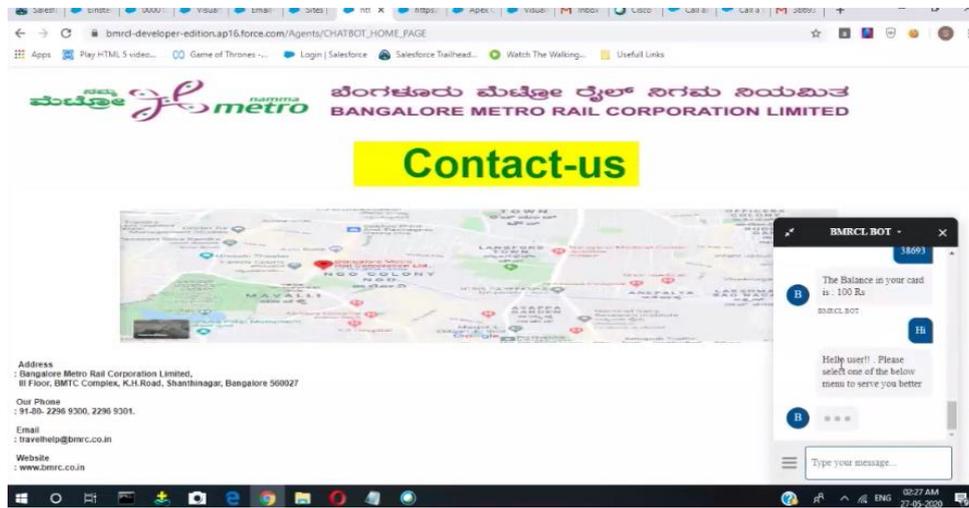


Figure 8: Intent recognition

The intents are recognized by adding utterances to intent sets. The above figure 8 shows one such intent “greeting” being recognized by the bot through NLP. If the user sends a greeting then they are greeted back because of the utterances we have added.

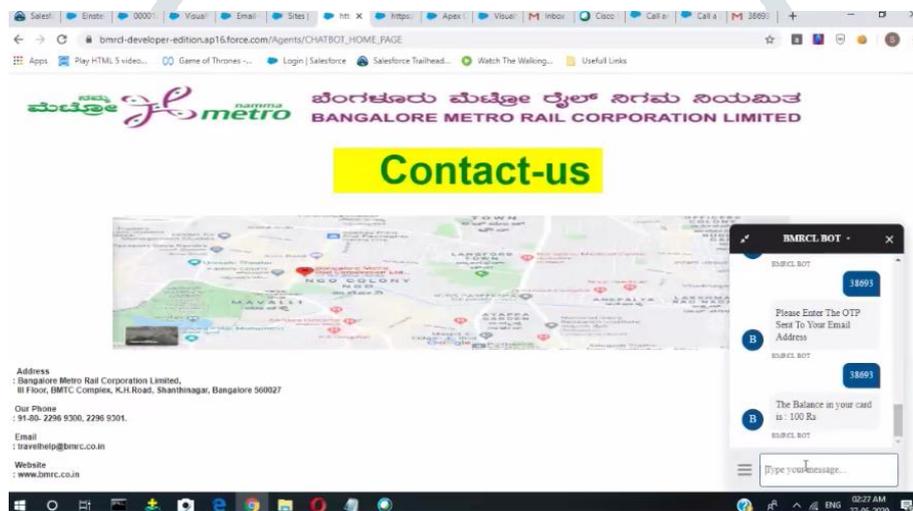


Figure 9: Balance being displayed when check balance flow is run

When the check balance flow is run user is asked for credentials like user id, email and also validation of email is also done through OTP. After all the validation the user is displayed his balance as shown in Figure 9.

7. Conclusion and Future work

We have designed a system which would overcome the downside of the first BMRC application which was tied in with fulfilling the clients. The issue was BMRC application where neglecting to keep up a decent client support as far as taking care of the case. The system which is created utilizes the CRM offered by Salesforce to give consumer loyalty to its clients. Here an intelligent bot is proposed which can be utilized by the clients to buy a smart card, check balance, check for FAQ's, etc. For progressively complex issues the client has the alternative of moving to a human operator. Through the use of utterances we can capture and respond to common questions that are asked. Along with this case management is offered to the client where auto assignment rules are utilized to allocate cases to queues and made agents to work upon. Through automated assignment rule emails as response to case creation are sent to the client.

- Further wish to include functionalities that help the client like including an approach to get to metro bicycles and other metro administrations. This is as of now not executed as the geo map utilization isn't available in basic version.
- Creation and utilization of lightning component is proposed, which will give better look and feel to our BMRC usage.
- Making utilization of geography tags offered by Salesforce where we can include it for the Visualforce page. This would assist the client with visualizing the map offered by BMRC better.

References

- [1] Jigar Patel, Ankitchauhan, "An Approach to Introduce Basics of Salesforce.com: A Cloud Service Provider", IEEE, 2016.
- [2] KrutarthSoni, "Roadmap to Salesforce Security Governance & Salesforce Access Management", IEEE, 2015.
- [3] NeroidaSelimi, MarikaApostolovaTrpkovska, "Utilization of Customer Relationship Management (CRM) Theory, Prototype and Tool for Improved Strategic Marketing in HE", MCEE, 2017.
- [4] Am rahman, "Programming challenges of Chatbot: Current and Future Prospective", IEEE, 2018.
- [5] Yuxion chai, "Utterance Censorship of Online Reinforcement Learning Chatbot", IEEE, 2018.
- [6] Aneta Poniszewska Maranda, Radosław Matusiak, "Use of Salesforce Platform for Building Real-time Service Systems in Cloud", IEEE, 2017.
- [7] Nor AlinaBinti Ismail, "The Effect of E-CRM features on Customers Satisfaction for Airline E-ticket Services in Malaysia", , IEEE, 2016.
- [8] Shufeng Zhang, Zengliu, "Research on the Construction and Robustness Testing of SaaS Cloud Computing Data Center based on the MVC Design Pattern", IEEE, 2017.
- [9] Anuradhamanohar, Ankitchauhan, "Salesforce CRM: A new way of managing Customer Relationship in cloud environment", IEEE, 2016.
- [10] Ali Ibrahim, madri, Arisprativi, "Measuring Customer Satisfaction Using CRM Scorecard in Canteen FASILKOM UNSRI", ICECOS, 2016.
- [11] Abdallah A.Z.A. Ibrahim, "PRESENCE: A Framework for Monitoring, Modelling and Evaluating the Performance of Cloud SaaS Web Services", IEEE, 2018.
- [12] Chao Feng, Naiding Yang, Ting Lei, "Research on method of emergency cases retrieval based on case-based reasoning", , ISCID, 2016.
- [13] Neelkumar P. Patell Devangi R. Parikh, "AI and Web- Based Human-Like Interactive University Chatbot (UNIBOT)", IEEE, 2016.
- [14] Borislav S. Ćorčević, Slobodan P. Jovanović and Valentina V. Timpenko, "Intent Detection and Slots Prompt in a Closed Domain Chatbot", IEEE, 2016.
- [15] Bavika R ronali, sanjaysingh, "Chatbot for University Related FAQs", IEEE, 2018.
- [16] Albert Verasius Dian Sano, "The Application of AGNES Algorithm to Optimize Knowledge Base for Tourism Chatbot", IEEE, 2018.
- [17] Shanglei pie, "Application of Data Mining Technology in the Tourism Product's Marketing CRM", IEEE, 2016.
- [18] SodamBaek, Kibae Kim, JornAltmann, "Role of Platform Providers in Service Networks: The Case of Salesforce.com AppExchange", IEEE, 2015.
- [19] Xian Zuh, "Design of mobile system for metro value added services", IEEE, 2017.
- [20] HenkieOngowarsito, "The Impact of e-CRM Implementation to Bank Performance Through e-Service Quality", IEEE, 2016.