Amazon Campus Mentorship Series

SPLENDID SALES

CRM model

Project Mentor: Bharat

Alekhya Tethali	**alekhyatethali@outlook.com**
Charishma Dasari	**dasari.charishma@gmail.com**
Jasmine Wadhwania	**jasmine.wadhwania@gmail.com**
Samaikhya choppadandi	**samaikhyachoppadandi@gmail.com**
Kudaravalli Tejaswi	**tejamanu117@gmail.com**

May 22, 2018

Contents

1 Project Management Plan

1.1 Overview

Splendid Sales is a CRM platform that manages the sales interaction between the company's sales team and its customers with the help of the information provided by the company, like the company's progress, customer subscription data, etc. Splendid Sales follows IDIC model i.e., Identification, Development, Interaction and Customize. Splendid Sales analyzes the customers, thus helping the employees understand the customer needs in a better way and serves the customers profitably.

1.2 Assumptions, Scope and Objectives

Assumptions:

- It is assumed that the data is received from the company on a weekly basis and the schema is designed based on what is projected onto the CRM. The static data is assumed to be updated every week.
- The Gmail API file which is used to send emails has been authenticated and given to the CRM System in a format as required by celery. This is a one-time generated file.

Scope:

- The CRM generates analysis of the customers, products and the sales for each salesman as his progress (As of now no progress)
- Provides employees the customers' interest and understand their needs
- The employee can send prompts, updates, alerts to the customers
- Provides payment transactions using blockchain technology

Objectives:

- To generate the sales of the month
- To generate the analysis of the customers visit
- To analyze the customers' needs and interact with the company

2 Requirements Specification

Build a CRM (Customer Relationship Management) Application to manage sales interaction between a company's sales team and its customers.

Phase 1

The application will allow sales team to:

1. Manage their leads by tracking their progress on a UI
2. Send customizable mails automatically on certain triggers/accomplishments
3. Track new customers on websites and reach out to them accordingly

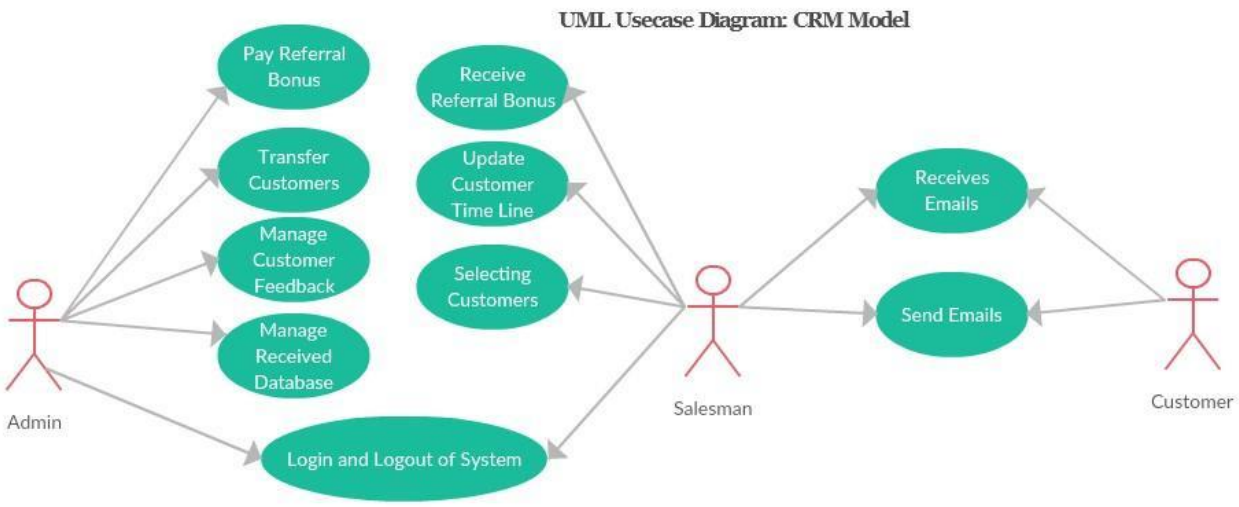
Phase 2

There could be options of exploring advanced ideas aa additional feature requirements such as:

1. Using Smart Contracts based on Blockchain for paying out a referral bonus.
2. Using Natural Language Processing for sentiment analysis on mails using Gmail API.

3 Analysis

3.1 Use Case Diagram



3.2 Use case Description

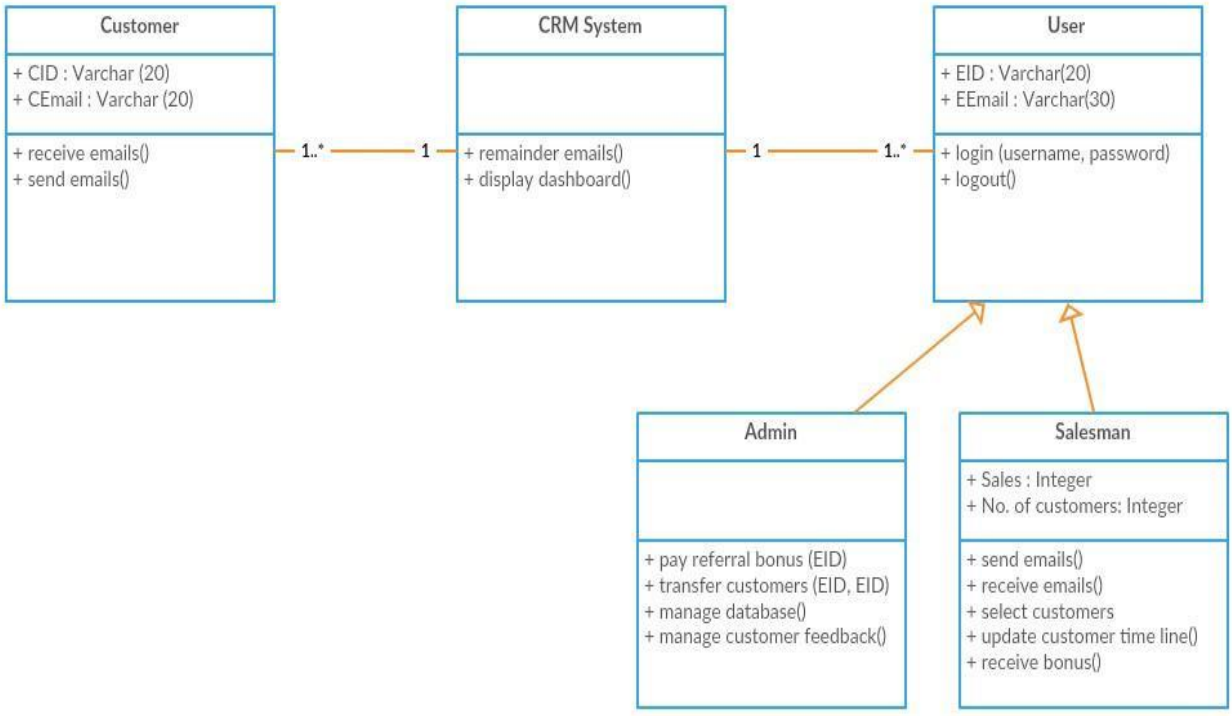
A Customer Relationship Management (CRM) model is an interactive system that provides the interaction between the customers and the sales team of the company.

The Admin (actor) will receive the database (use case) on a weekly basis from the company. Using this database, the new potential customers list is updated on the dashboard of the Salesman (actor). The Salesman must select a list of customers (use case) who he/she wishes to deal with. The Customer (actor) is sent an email containing the details of the allotted Salesman and the potential subscription plans by which the Customer can be benefited. Customer can choose to reply to the email, if he/she is interested in the deal. Customers are also sent emails reminding the expiry of their subscriptions. The Admin has the power to Transfer Customers (use case) from one Salesman to another, as per the requirement. Admin can also Pay Referral Bonus (use case) to the Salesman based on their sales performance and manage the customer reply emails (use case).

4 Design

Customer Relationship Management System Class Diagram describes the structure of a Customer Relationship Management System classes, their attributes, operations (or methods), and their relationships among objects. The main classes of the Customer Relationship Management System are customer remainders, new subscriptions, feedback analysis.

4.1 Class Diagram



4.2 Classes of CRM Class Diagram

- CRM System Class: Manages all overall CRM system operations
- Customer Class: Manages all operations of Customer
- User Class: Manages all operations of User
- Salesman Class: Manages all operations of Salesman; generalization of User class
- Admin Class: Manages all operations of Admin; generalization of User Class
- CRM Database Class: Manages all operations of Database

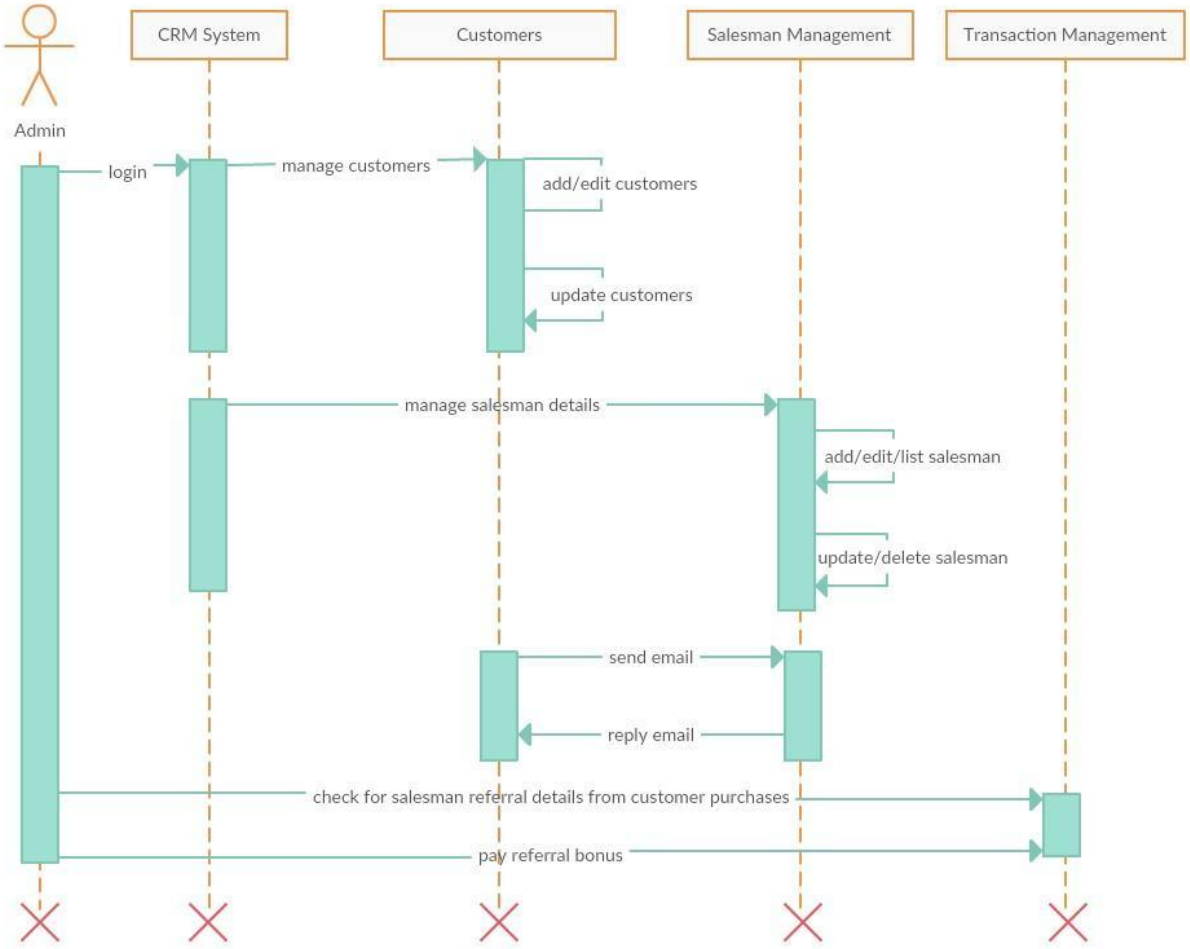
4.3 Classes and their attributes of CRM Class Diagram

- Customer Attributes: CID, CEmailID
- User Attributes: EID, EEMailID
- Salesman Attributes: Sales, No. of Customers

4.4 Classes and their methods of CRM Class Diagram

- CRM System Methods: remainder emails(), display dashboards()
- Customer Methods: receive emails(), send emails()
- User Methods: login(), logout()
- Admin Methods: pay referral bonus(), transfer customers()
- Salesman Methods: send emails(), receive emails(), select customers(), update customer timeline(), receive referral bonus()
- Database Methods: getData()

4.5 Sequence Diagrams



5 Implementation

5.1 Software technologies used

1. Python 3.5
2. PyCharm Community Edition 2017.2.1
3. PostgreSQL 9.6
4. Psycopg2 (database driver)
5. Visual Studio Installer
6. Redis (used as a message broker)
7. Node.js

5.2 Execution instructions and machine/system configuration used:

1. django-admin startproject ‘project_name’
2. python manage.py startapp ‘app_name’
3. python manage.py runserver
4. python manage.py migrate
5. python manage.py makemigrations
6. python manage.py instancedb
7. pip install ‘package_name’

6 Source Code

Please find the source code in the attached GitHub link:

http://github.com/crmwebsite/spendidsales

7 Database Schema

Auth_User Table

id	integer
password	character varying(128)
last_login	timestamp with time zone
Is_superuser	boolean
Username	character varying(50)
First_name	character varying(30)
Last_name	character varying(150)
Email	character varying(254)
Is_staff	boolean
Is_active	boolean
Date_joined	timestamp with time zone
Sales	numeric

Registration Table

cid	character varying(20)
first Name	character varying(20)
last Name	character varying(20)
phone No.	bigint
age	integer
gender	text
Cemailid	character varying(20)
Location/City	character varying(20)

Resource Table

rid	character varying(20)
rname	character varying(20)

Ex: Instance of Resource Table

rid	**rname**
AWS001	EC2
AWS002	Quick Sight

AWS003	Lambda
AWS004	S3
AWS005	RDS

Membership Table

mid	character varying(20)
rid	character varying(20)
mname	character varying(20)
Mexpiry	integer
Price	integer

Ex: Instance of Membership Table

rid	**mid**	**mname**	**Mexpiry (months)**	**Price**
AWS001	MAWS001	Monthly	1	Rs. 1000
AWS001	YAWS001	Yearly	12	Rs. 10,000
AWS001	PAWS001	Premium	60	Rs. 48,000
AWS002	MAWS002	Monthly	1	Rs. 1000
AWS002	YAWS002	Yearly	12	Rs. 10,000
AWS002	PAWS002	Premium	60	Rs. 48,000
AWS003	MAWS001	Monthly	1	Rs. 1000

Customer Table

Cid	character varying(20)
rid	character varying(20)
mid	character varying(20)
Eid	character varying(20)
dop	date
cexpiry	date
Status	character varying(20)

Visitors Table

Id	integer
Product	character varying(20)
Visitors	character varying(20)
Avg_time	character varying(20)