Customer Support Tool

Author: Francisco J. Castillo Rivera
Advisor: Othoniel Rodriguez, PhD
Electric and Computer Engineering Department

Abstract

Small companies rely on after-sales support for achieving customer satisfaction. A common problem for these small companies is the lack of adequate staffing to provide good customer support, it could get even worse when companies sell technology which demands a high level of after-sales support. A customer-support application tool could help overcome these limitations by automating many of the functions involved in after-sales support and making more efficient utilization of the existing human resources. This software tool will include functions such as: handling available personnel and assigning tasks to them, sending notifications of new customer calls, keeping track of unresolved problems, and even reminding the support personnel when proactive tasks such as preventive maintenance are needed. In addition, to helping retain existing customers, this Customer Support Tool will easily scale-up to handle additional new customers with only small increments in human resources.

Introduction

The Customer Support Tool is a web-based application with the capability of storing, pending or resolved, day-to-day customer calls, notifying the available technician via email or text message when a new call occurs, notifying personnel when a preventive maintenance is needed for a specific equipment and guide them through the process, help sales persons to submit new quotations inquiries, notify sales persons when the quotation is ready for the customer, storing lead customers for future follow-up, and generating reports for customers leads and day-to-day customer calls.

The system main components are the SQL database, and the content management software. The content management software allows the development of graphical user interface using languages like JavaScript, PHP, HTML, and CSS and provides an easier way to handle users.

The content management software is responsible for storing the user input in the SQL database component. Also, it is responsible for handling user authentication and authorization in order to keep the Customer Support Tool data safe. Both components are installed in the same web server.

Requirements

- The system stores and allows access to customer calls in a database.
- The system keeps customer calls as pending or closed.
- The system notifies the available technician via email or text message of a new entry using REST API.
- The system generates a report of stored customer calls.
- The system allows sales user to request new equipment quotes.
- The system provides a way to notify the sales user when a quote is ready for the customer using REST API.
- The system stores and allows access to customer leads in an organized matter.
- The system stores and allows access to customer visits in order to keep customer follow-up history.
- The system generates a report of customer visits stored in the database.

Problem

Nowadays, not every company is able to afford a large number of personnel and invest in automated tools that help grow their business. Instead, companies try to do more with less, and that's were the real problem arises. Our solution is meant for these companies because is an affordable way to have a small number of personnel and still be able to offer clients a great customer service.

Methodology

The Customer Support Tool is designed with extensibility and scalability in mind. We are taking great care in designing a framework which can be updated easily. Many of the anticipated changes to our system in future phases will only require adding new types of data and changing the user presentation code to make use of these new data. The design will only require "plugging in" these new types of data without refactoring the logic that passes the data over the network, retrieves and updates the database, etc. There are three basic, logical components of the system: Database Engine, Server Application, and Client Applications.

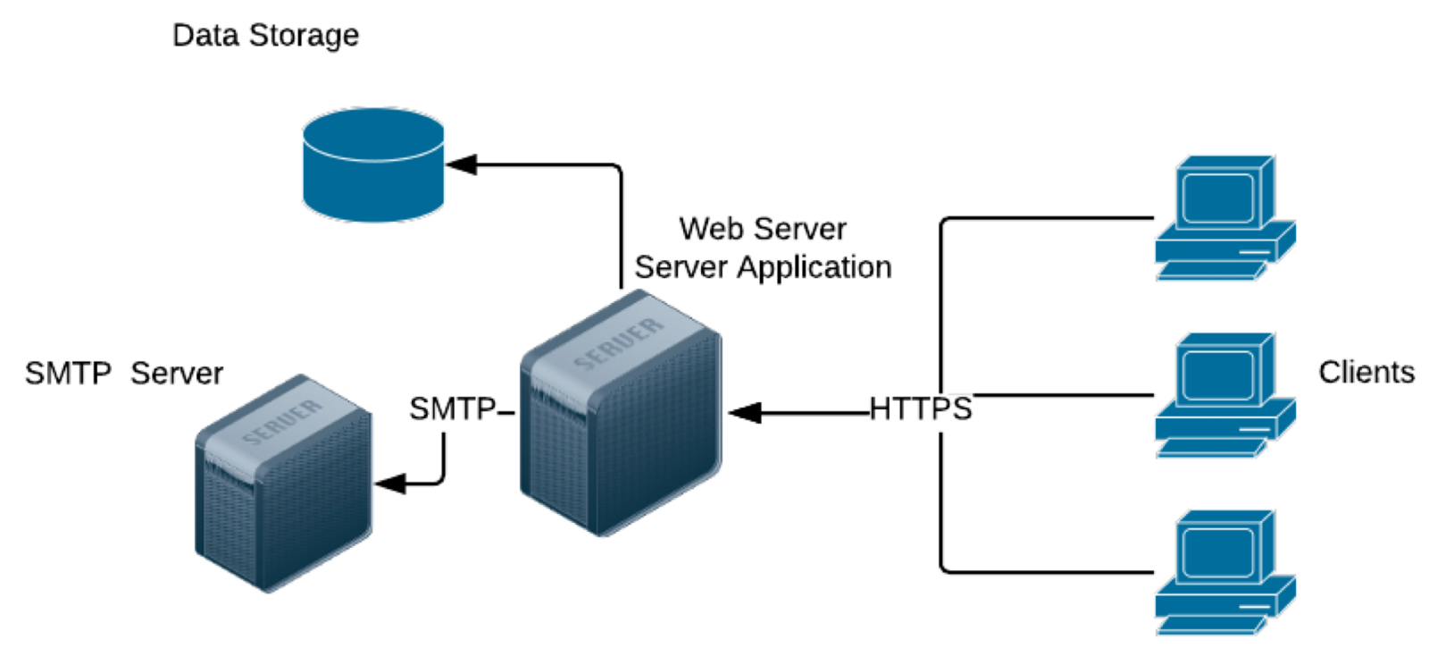


Figure 1 Overall System Architecture

Database Engine – SQL data source used to handle data regarding users and service information.

- Existing open source software: phpMyAdmin
- Hosts the backend database which is used for central data storage.

Server Application –Application that resides in a web server. Implemented in HTML, PHP and Javascript

Provides methods and procedures that can be invoked remotely by a client application.

Retrieve data.

- Update data.
- Create new data.
- Generate reports.

Central process which can make all decisions that arise due to the distributed nature of this application.

- For instance, when a client wishes to update a data, there may be conflicts that need to be resolved if another client has updated the same data.
- The server can coordinate conflict resolution with the client application.

Client Application – Existing application to interact with server application and end user.

- Existing Browsers.
- Contains all presentation logic.
- Interacts exclusively with the user.
- Communicates with the server application through HTTPS.

Results and Discussion

The CMS is in charge of the operation and interaction between the user and data. It decides the system behavior depending on the user input. Interaction with the CMS is only available for registered users.

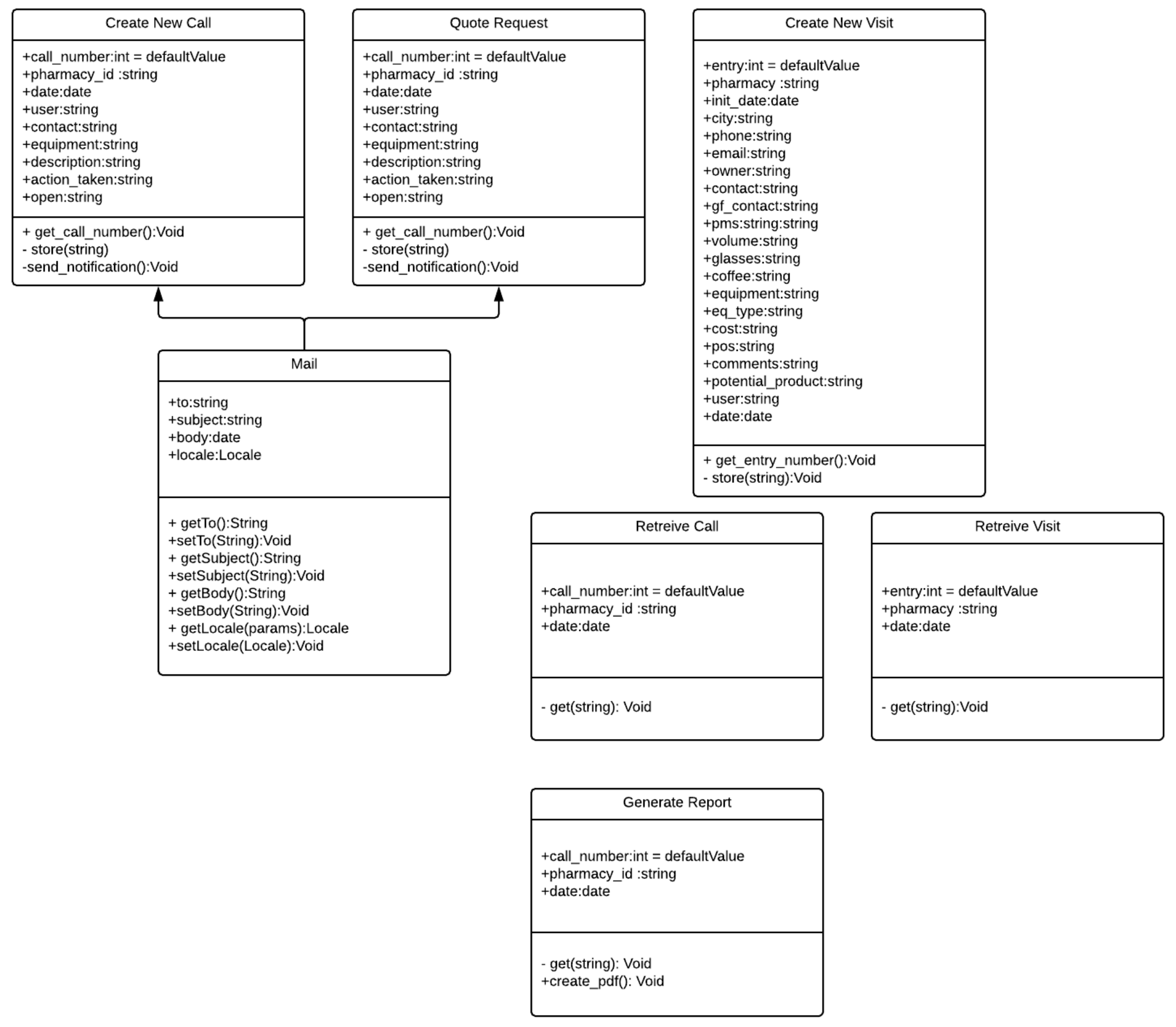
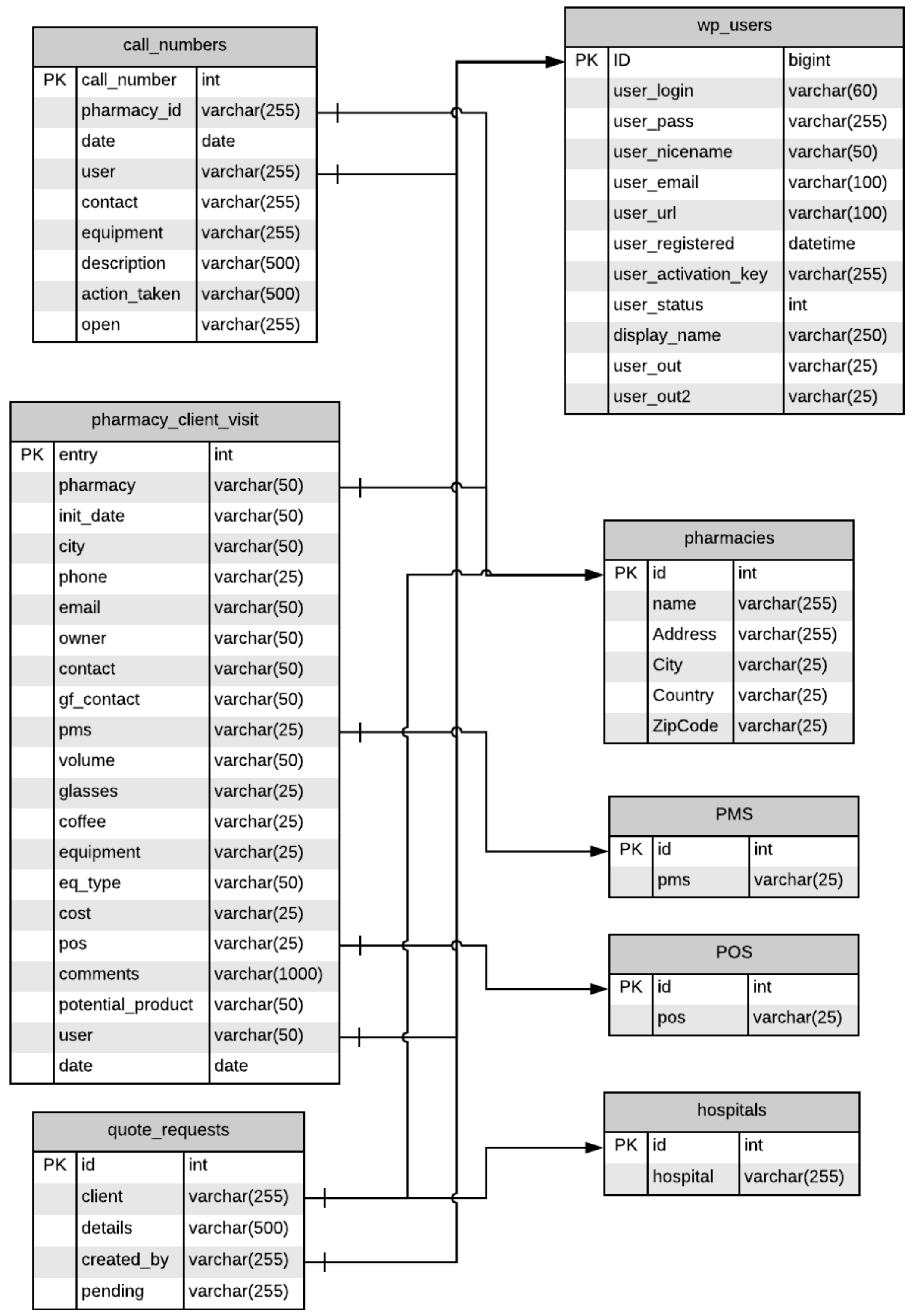


Figure 2 CMS Class Diagram



call_numbers		
PK	call_number	int
	pharmacy_id	varchar(255)
	date	date
	user	varchar(255)
	contact	varchar(255)
	equipment	varchar(255)
	description	varchar(500)
	action_taken	varchar(500)
	open	varchar(255)

wp_users		
PK	ID	bigint
	user_login	varchar(60)
	user_pass	varchar(255)
	user_nicename	varchar(50)
	user_email	varchar(100)
	user_url	varchar(100)
	user_registered	datetime
	user_activation_key	varchar(255)
	user_status	int
	display_name	varchar(250)
	user_out	varchar(25)
	user_out2	varchar(25)

pharmacy_client_visit		
PK	entry	int
	pharmacy	varchar(50)
	init_date	varchar(50)
	city	varchar(50)
	phone	varchar(25)
	email	varchar(50)
	owner	varchar(50)
	contact	varchar(50)
	gf_contact	varchar(50)
	pms	varchar(25)
	volume	varchar(50)
	glasses	varchar(25)
	coffee	varchar(25)
	equipment	varchar(25)
	eq_type	varchar(50)
	cost	varchar(25)
	pos	varchar(25)
	comments	varchar(1000)
	potential_product	varchar(50)
	user	varchar(50)
	date	date

pharmacies		
PK	id	int
	name	varchar(255)
	Address	varchar(255)
	City	varchar(25)
	Country	varchar(25)
	ZipCode	varchar(25)

PMS		
PK	id	int
	pms	varchar(25)

POS		
PK	id	int
	pos	varchar(25)

quote_requests		
PK	id	int
	client	varchar(255)
	details	varchar(500)
	created_by	varchar(255)
	pending	varchar(255)

hospitals		
PK	id	int
	hospital	varchar(255)

Figure 3 Data Storage Schema

Implementation

This section outlines the process to be taken for the Customer Support Tool implementation. The application will be implemented on a web server with enough requirements for hosting the server application and the data storage. Once installed, a data entry will be performed in order to have the latest company's list of clients and users in the data storage. Employee roles will be defined, and access will be determined by the role.

Also, due to the complexity of maintaining a local application server and smtp server, a web hosting service will be implemented to reduce down time risk. A three-month support period will be provided to assure the application is working as expected and fix any issues that may have occur during the implementation phase.

If a problem is found during the implementation, a re-execution of the test case involving the feature being implemented, will be made in order to keep record of changes. After re-execution of the test case, the feature will be implemented.

Conclusions

The Customer Support Tool was successfully implemented and deployed. All components have passed their Unit Tests and their respective Acceptance Tests, after executing these tests we have seen no issue within the tests performed.

The process of data entry through the first days ran smoothly, considering high volume of calls being received in a daily basis.

The company's staff is very happy with the results of the new system and they have high hopes for their future as a sales and customer service company.

Future Work

- Develop an employee time entry system to record their working hours.
- Create payroll based on the entered working hours and the employee rate to facilitate the payroll process.
- Develop a payroll email notification once the payment has been processed.
- Develop mobile devices applications to access the Customer Tool instead of using a Web Browser.

Acknowledgements

We would like to thank Professor Dr. Othoniel Rodriguez for being my mentor through this whole process. Also, I'll like to thank Modern Tech Associates for giving me an opportunity in order to make this project a reality. Finally, my family, who played a very important part during my two years of being graduate student at the Polytechnic University of Puerto Rico.

Team

Name	Role
Francisco Castillo	Developer/Document Manager

Table 1 Team Members