ASHESI UNIVERSITY COLLEGE

GEOGRAPHIC INFORMATION SYSTEM: CUSTOMER MANAGEMENT AND SERVICE APPLICATION FOR SMALL AND MEDIUM ENTERPRISES.

THIERRY NUKU GNANIH

2013

Applied Project

ASHESI UNIVERSITY COLLEGE

Geographic Information System: Customer Management and Service Application for Small and Medium Enterprises in Ghana.

By

Thierry Nuku Gnanih

Dissertation submitted to the Department of Computer Science,

Ashesi University College.

In partial fulfillment of the requirements for the award of Bachelor of Science degree in Computer Science.

April, 2013

Declaration Page

I hereby declare that this dissertation is the result of my own original work and that no part of it has been presented for another degree in this university or elsewhere.

Candidate's Signature:……

Candidate's Name:…….

Date:……..

I hereby declare that the preparation and presentation of the dissertation were supervised in accordance with the guidelines on supervision of dissertation laid down by Ashesi University College.

Supervisor's Signature:…….

Supervisor's Name:……..

Date:……

Acknowledgement Page

First and foremost, I would like to thank my family for supporting me throughout challenging years in school. I am grateful to my close friends, especially Kpetermeni and Frank who encouraged me to believe in my work. Without their words of encouragement, this project would not have been a success.

Special thanks goes to my supervisor, Aelaf Dafla who guided and encouraged me to deliver beyond my basic capabilities.

Abstract

The project aims at providing a user-friendly application to enable Ghanaian small-scale firms to provide better after-sales service support. The provision of effective and efficient door-to-door customer service in Ghana has been a challenge to Small and Medium Enterprises (SMEs). This can be attributed to cost of acquiring technology that can help to streamline processes. However, Geographic Information Systems (GIS) have improved significantly has become accessible in recent years. This enables SMEs to make use of it in order to save time and cost in their quest to reach desired destinations and provide customers with quality services. The implementation of such a tool in a customer management application used by SMEs will aid in the organization's provision of valuable services to its customers. The tool will help to save time and maximize the use of resources in planning and executing field service trips. The application developed provides a customer management software with Geographic Information System to SMEs. The tool can make an impact on processes used in planning after-sales service. It will provide organizations the power to provide effective and efficient services to customers.

Table of Contents

1. CHAPTER 1: INTRODUCTION

1.1 Introduction and Background to the Project

After-sales service in Ghana has proven to be a challenge to several organizations, ranging from the Small and Medium Enterprises (SMEs) to the bigger and more established companies. Many at times we are faced with multinational companies like Vodafone Ghana struggling to locate service consumers when they must make onsite visits to solve certain problems. On their way to the site, several calls are made to the customer for directions. In the situation where the service personnel are not familiar with the area or the area is remote, it becomes difficult to locate the customer. This often creates frustration for both parties. As such, a frustrated customer will only carve a negative image of the organization.

Nevertheless, technology has greatly evolved and there exists several tools that can provide easier processes to plan service trips. There are application packages that incorporate a host of tools to help companies improve their customer relations. These tools however have proved to be costly to Ghanaian SMEs. They therefore resort to manual processes, which can be time consuming and can affect their relationship with customers.

The technology industry has seen major improvements in the field of Geographic Information Systems (GIS). This system is a computer-based technology that combines geographic data and tabular data to generate

visual maps, analyze and assess real-word problems (Mandayam) (GIS Lounge, 2013). Value can then be added to the map through additional layers that provide specific information, relevant to the organization. Tools of this nature were originally used for natural disasters and weather forecasts. However, the improvement of such systems have now opened the way for several organizations to make use of GIS services, in their quest to increase efficiency in their business organizations (Drummond & French, 2008). GIS services can be of particular advantage to companies that have after-sales services as part of activities that make up their business objectives.

It is in this light that this project aims to integrate Geographic Information Systems with already existing customer management applications used by organizations. This will go a long way to reduce planning time and increase the efficient use of resources in providing quality after-sales service to customers.

1.2 Problem

To illustrate the problem an example of a company, which deals in office Multifunctional Products, and Digital Duplicators will be used. In the past, the firm has championed the Ghanaian market in providing quality after-sales services to its customers in the industry. Periodic service trips were made to several regions. Teams usually spent an average of two weeks or more on trips to several regions in the country. Occasionally, technicians would individually go on trips and would have to locate the customer on their own. On several occasion, a number of calls would have to be made

in order to get accurate directions to the customer. This cost the company in wasted technician time and frustration from the client side.

The problem can therefore be broken down into three issues:

- The inability to make on-the-spot decisions when service calls or orders are received through the phone
- The amount of time it take to plan field service trips
- Difficulties in locating customers on the field

The company is now faced with the challenge of re-championing after-sales service support trips to the remotest parts of the country in a more efficient and effective way.

1.3 Objective

The general objective of this project is to create an interactive web-based Geographic Information System, using Google Maps API, which will enable customer service personnel to seamlessly plan field service trips, as well as help the organization save time and cost in service delivery. Furthermore, the system would help the organization reduce the number of calls that must be made in order to locate customers in some of the most remote parts of the country. In the long-term, this application will also serve as a decision support tool that will aid management in decision-making. Management will easily make a decision as to how much of its resources to commit to specific field trips (number of technicians, transportation etc.) as well as make informed statements when calls are received.

The development of the product will be conducted with a Ghanaian SME that was used to illustrate the problem. As a client organization, which has the provision of quality after-sales service support as one of its key business drivers. The procedures used by the organization to process service calls will be a guide, as to the specifics that the system should be able to perform. The company has been in the office machine industry for more than twenty-five (25) years. The organization has several years of experience in providing after-sales services to customers in some of the most remote parts of the country and hundreds of kilometers away from the office. As such, their motto says it all; 'For the company, after-sales service is a religion'.

1.4 Motivation

The motivation for the use of Geographic Information System is from previous experiences with major service providers like Vodafone Ghana, Electricity Company of Ghana (ECG) or smaller organizations that need to provide door-to-door services.

An example many people can relate to is when we have electrical faults or network issues in our offices or homes. Getting service providers to the location can be a hassle. Giving directions over several telephone conversations and leaving important duties to spend hours waiting for the service provider to arrive at your destination can be very irritating. As such, productivity is lowered on the side of the customer as well as on the side of the service provider.

Moreover, having worked several months with a company where different customers were visited on a daily basis, and given that it is not always easy to follow directions given by a person, a GIS system would be most valuable in solving this problem.

As I have used Google Products on countless occasions, features offered by Google Maps is something that fascinates me. Google Maps provides users with a simple to use interface that enables one to make use of in-depth geographical information to generate 'reports that speak'. These stories can make life easier for organizations where staff need to move around on several occasions.

1.5 Outline of Dissertation

This dissertation, divided into distinct chapters, will provide an insight into the different components needed for the development and implementation of the Geographic Information System. Chapter two (2) will provide a review of the role of efficient after-sales service of an organization. Chapter three (3) will focus on design specifications and details of the product by listing features as well as functional and non-functional requirements of the application. Next, chapter four (4) will focus on requirements for implementing the solution. Details will be provided as to what technology and tools will be used, as well as the platform on which the application will be deployed. Chapter five (5) will give an analysis of tests performed and results obtained from the trials. It will also outline recommendations on what can be done in the future and the limitations of the application.

2. CHAPTER 2: DESIGN

2.1 Similar Applications

2.1.1 Miracle Service

Miracle Service is Service Management Software aimed at streamlining sales and service operations. The use of this service helps increase productivity and efficiency (Miracle Service, 2013). Miracle Service has Routing and Mapping features that enable organizations to easily plan service calls. Below are a few screenshots showing what the application looks like.

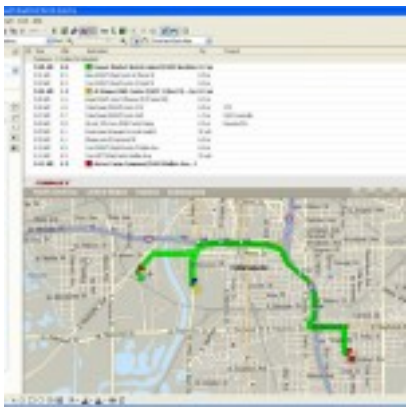

Figure 2.1 - Screenshot of Miracle Service

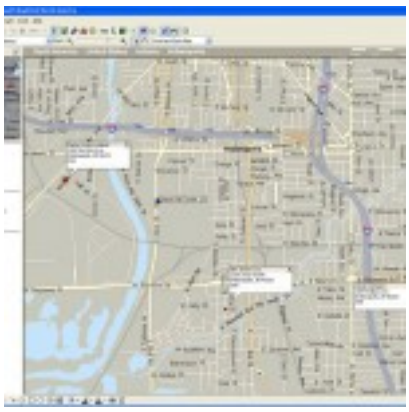

Figure 2.2 - Screenshot Miracle Service

2.1.2 Oracle Maps

Oracle has several ways in which mapping features can be used. There are readily available suites, as well as database applications with which Maps can be easily integrated, often without writing a single line of code (ORACLE, 2008).

Oracle Maps is a full suite of technologies that enables one to develop high-performance Web-based mapping applications (ORACLE, 2001). Oracle has the ability to integrate Google Maps with its Field Service application. The aim of this development by Oracle is to aid service providers locate customers more quickly (ORACLE, 2001).

2.2 Functional and Non-Functional Requirements

2.2.1 Functional Requirements (Geographic Information System)

1. **Register a new client** – This feature accepts customer information submitted by service personnel goes online to retrieve geographical data and stores it in the customer table in the database.
2. **Display customer location** – This feature uses input information from service personnel as well as Geocoded information to plot customers on a map using markers.
3. **Calculate distance to client location** – The system uses coordinates generated by the Geocode and the distance matrix to calculate distance from the service provider's office to the client's location.
4. **Render Directions** – This feature draws a road path from the service provider's location to the customer's location. It selects the shortest route available. Additionally, this feature displays a panel with step-by-step directions from the service provider's location to the customer's location.
5. **Plot all customers** – This feature plots all customers of the database on one single map. This will enable management to see

the geographical distribution of their customers across the country.

2.2.2 Other functional requirements

1. **Add complaints** – The application allows users to register complaints from customers. Complaints will be recorded based on products and their serial number. The counter reading of the machine will also be recorded, as a complaint is being filed.
2. **Generate Job Card** – The application will generate a job card report that will display relevant information for the field technician. The card will include: a table with customer information, the static map with directions, distance to customer location, and approximate travel time by road.

2.2.3 Non-functional requirements

1. **Internet Connectivity** – Most features of the GIS features in this application must make requests to Google Servers. Therefore Internet connectivity must be at least stable for an effective and efficient use of the system.
2. **Accuracy –** Information generated by requests to Google APIs must be as accurate as possible. Inaccuracy of data may lead to management of the service providers to take wrong decisions.
3. **Performance** – The application makes calls to online scripts. Therefore, as it will be deployed in an environment where Internet connectivity is not always stable, a high performance and 100% uptime cannot be guaranteed.

4. **Legislative requirements** – The GIS section of this application must abide to the rules and regulations of Google Maps API.
5. **Authentication** – Users of the application shall authenticate themselves using the username and passwords assigned to them by the IT administrator. This will enable management to track records in terms of users.

2.3 System Architecture

This section of the dissertation provides details of the application's design and system requirements specifications.

The GIS application is based on a three-tier client-server model where the client side will be based on a Web Browser. The application logic will be hosted on the Apache Web Server, which will contain all JavaScript and PHP, codes serving to run requests or queries to the MySQL server as well as the Google Maps Servers. The Google Maps server will then return results, which will be displayed, to the end-user through the Web Browser. The end-user, depending on his or her privileges, can then decide to save the information provided or print it out as a report.

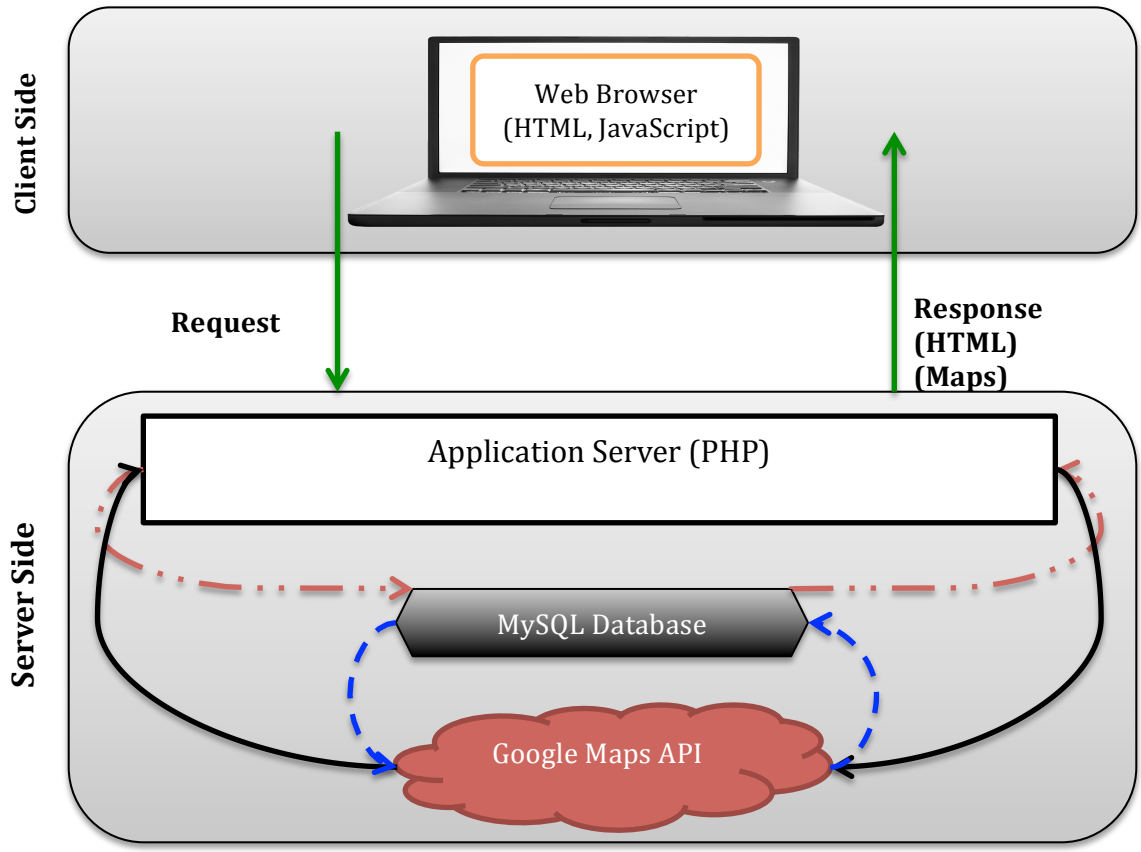

Figure 2.3 – System Architecture Diagram

2.4 Scenarios

1. Adding new customer

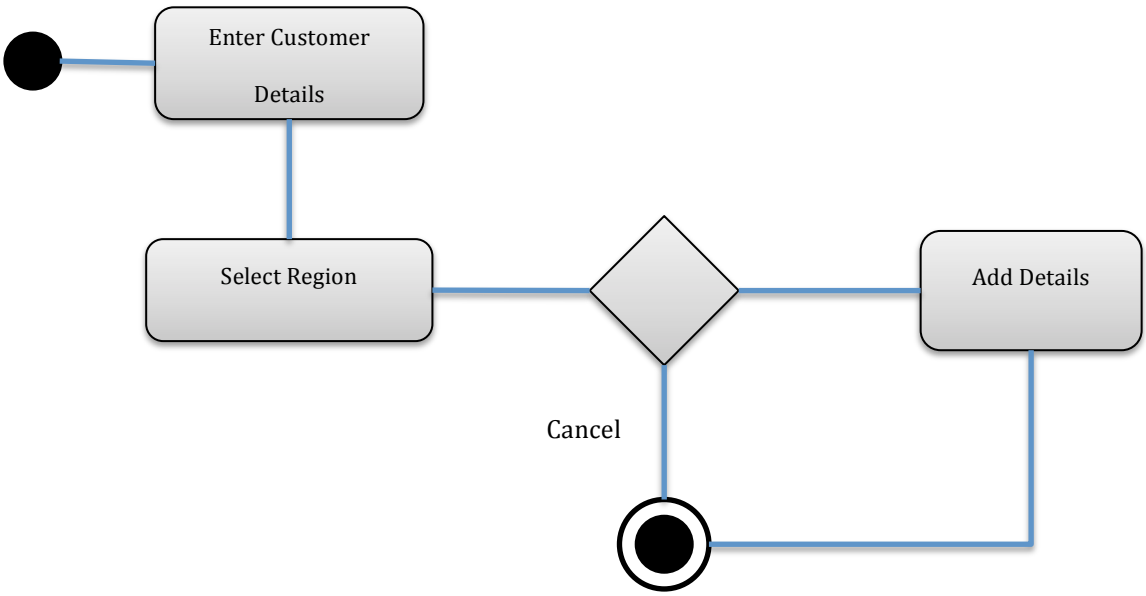

Figure 2.4 - Activity diagram for steps in adding new customer

2. View Customer

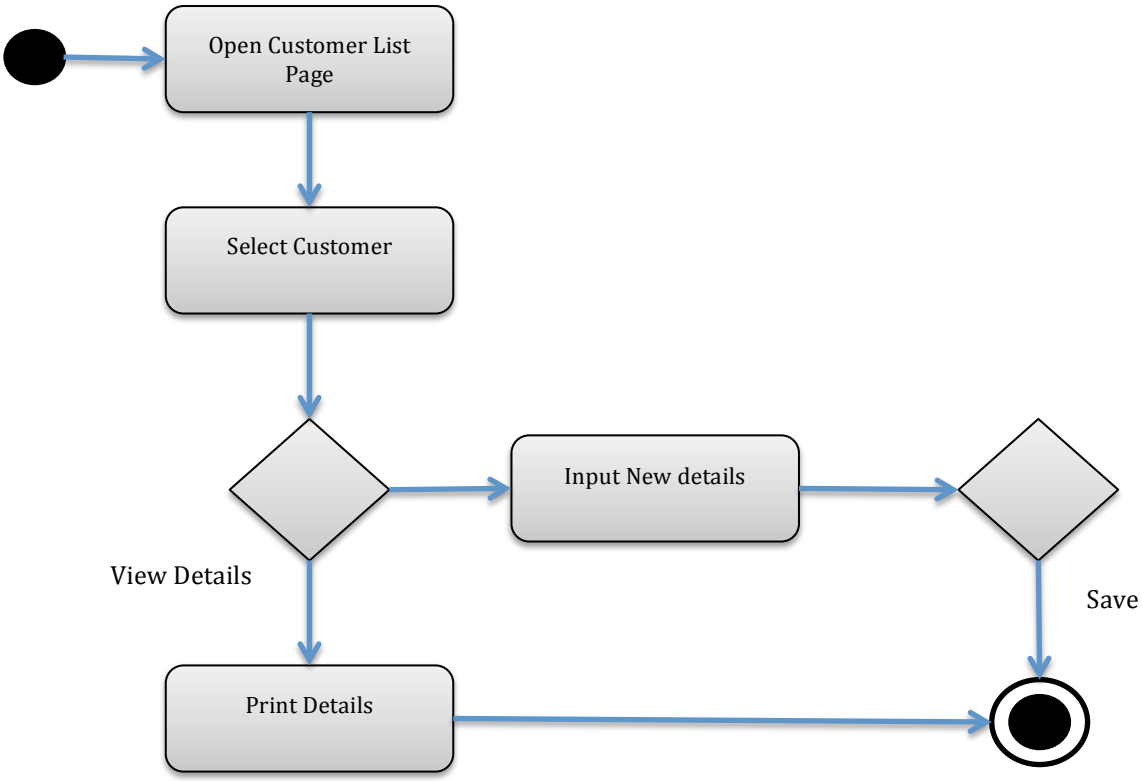

Figure 2.5 - Activity diagram showing steps to view customer

3. Adding new complaint

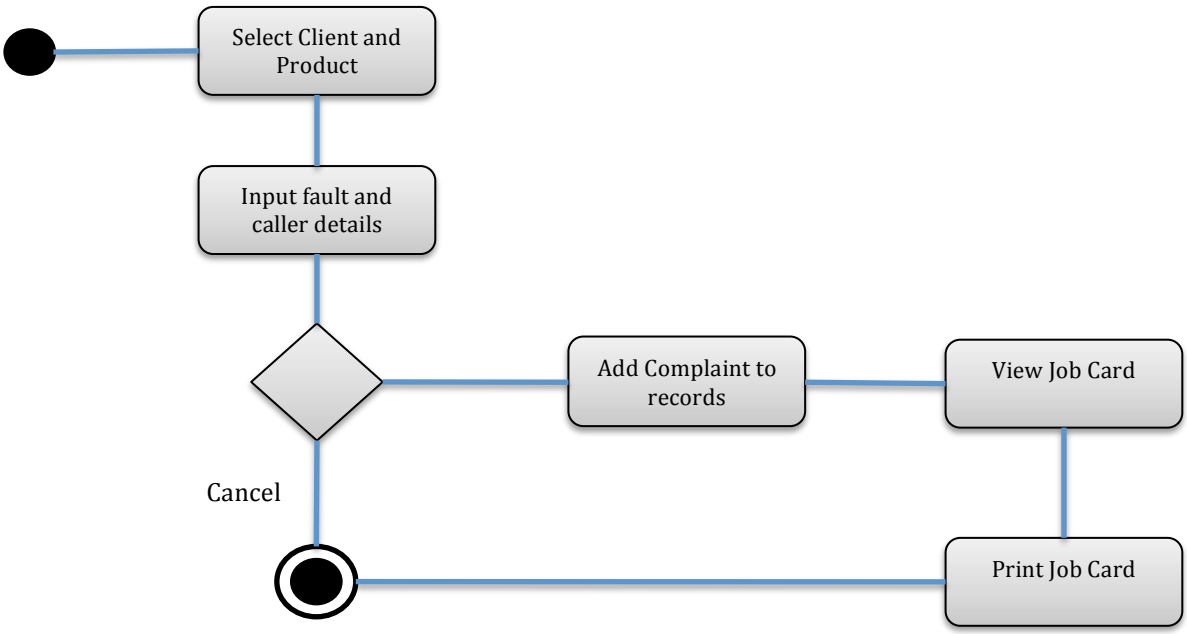

Figure 2.6 - Activity diagram showing steps to add new complaint

2.5 User Classes and Characteristics

The following groups of people will make use of the application on a daily basis:

- **Service call receiver –** This group consists of any service personnel who receives and records calls for field servicing. After recording call and complaints, he/she prints out a report consisting of the product in question and customer details (location, directions, distance). This person will have permissions to make edits to records.
- **Technical/field service personnel** – This group will be made up of technicians who will have specific permissions on the application. They will not be allowed to make edits. Their permissions will be limited to viewing.

2.6 Functionalities of the Map

The maps embedded in the application will have an 'info window' displaying the customer associated to the particular marker. Users will not be able to relocate the marker. The marker can only be moved when customer details are edited.

A path will be drawn from the service provider's location to the customer's location.

3. CHAPTER 3: IMPLEMENTATION

The application will be implemented by adding Geographic Information System tools to an already existing customer management application. The existing application will be modified accordingly to suite the GIS. Below are details as to how the application was implemented.

3.1 Technology and Tools Used

Table 3.1 - Tools and technology used

Tool	Description
PHP (Hypertext Preprocessor)	This is a robust server side scripting language used for creating dynamic Web pages. PHP is open source. As such it has a reliable and low cost online support. ▫ Detailed support from developers around the world. ▫ Affordability makes it the preferred programming language for companies around the world. ▫ Encompasses well-written libraries that can be integrated with applications. ▫ Vigorous scripting language that can be easily integrated with HTML pages to create dynamic applications that perform actions specified by user (Web Design Library, 2013) As the GIS application will fully make use of a central database, it is important to have a tool that can easily fetch from it and perform required actions. PHP is commonly used to execute operations on the MySQL database.
MySQL	This is another open source tool with great online support. It is the most popular free-to-access and high-performance database management system that can be integrated with web-based applications.

	▫ High-performance - Exclusive architecture that enables developers to configure the database server for specific applications that require unique needs. As technology systems are critical to businesses today, MySQL has the capability of meeting the most demanding performance expectations of any system. ▫ Web and data warehouse strength - The GIS application will make use of Google Maps features stored in the cloud, it is important to have a database system with strong compatibility with web functions provided by Google. ▫ Lowest Total Cost of Ownership - For organizations ranging from SMEs in Ghana to the larger corporations, cost is something that is carefully scrutinized. MySQL provides high-end data management features that are comparable to offers from other costly systems. (ORACLE)
HTML5 *CSS* *JAVASCRIPT*	The Graphical User Interface (GUI) of the application will be designed using the HyperText Markup Language (HTML) with Cascading Style Sheets (CSS) to customize the look of the interface. Lastly, Javascript functions will be used to perform more complex actions in the browser. These actions will be executed on the user's computer as opposed to the server on which the website is hosted.
GOOGLE MAPS API	The Google Maps JavaScript Application Programming Interface (API) allows one to embed interactive Google Maps on a website or application. The Google Maps API provides several utilities to create dynamic maps. Utilities range from generating distance to adding markers and several other layers.
JSON	JavaScript Object Notation is a lightweight data-interchange format.

JQUERY	JavaScript library.
fPDF library	fPDF is a PHP class which allows users to generate printable PDF reports with ease. 'F' stands for free. As such, there are no restrictions as to what it can be used for. Below is a list of the main features it offers: ▫ Measure unit, page format and margins ▫ Page header and footer management ▫ Automatic page break ▫ Automatic line break and text justification ▫ Image support ▫ Colors ▫ Links ▫ Page Compression (FPDF Library, 2013)

3.2 Platform

The application is web-based, accessible on all browsers, but preferably on the Google Chrome browser. It will however be restricted to the service provider's intranet and will not be accessible to the public. The application will make use of HTML 5 features combined with JavaScript functions. It will be hosted on an Apache Web Server. Its database will make use of the XAMPP MySQL server as well as the PHP functionality.

3.3 Database architecture

The database architecture employed for this application will be based on customer information relevant to decisions made by the organization. The format of data provided in the database must be very clear and concise, as it will be used to generate information from the Google API libraries. With precise formats, queries will be executed faster and will provide better results. The initial database without the Google Maps API features

contains most of the tables and columns. Certain changes and additions were made throughout the project to suit the GIS application.

Originally, the 'Customer' relation of the database includes the following columns:

Table 3.2 – Columns of the Customer relation of the database

Table Columns	Description
Customer Identification Number	A unique number will be assigned to the customer for identification
Customer Name:	The customer name will be the name by which the service provider knows them. It may be an acronym or any short form of the organization's name.
Customer Address:	The customer address will be selected from a database of places available on Google Maps, based on the autocomplete feature of the text field. This will use the Google Places feature.
Customer Phone Number:	Customer phone numbers will be obtained from customers.
City:	The city will be selected from the autocomplete list provided by libraries of Google Places.
Status:	The status of the customer will specify whether they are active or inactive. Active means that the customer is still actively engaged in business transactions with the service provider and inactive means otherwise.

Additional columns described below were added as required by the Google Maps API as well as per requirements of the functionalities desired in the application.

Table 3.3 – New additions to the customer relation of the database

Table Columns	Description
Latitude and Longitude:	Coordinate values will be Geocoded by the Google Maps API. These values will then be used for the Reverse Geocoding feature of the application.
Category:	The category field will specify the kind of institution or organization the customer is. The type of marker used on the map will depend on the category.
Distance:	The distance from the service provider's location to the customer's location will be obtained using the Distance Matrix of the Google Maps API. The distance will be displayed in Kilometers.
Region:	This field will be based on official regional demarcations. This field will be useful to cluster markers by region. This will help service personnel to narrow down searches.

Below is a diagram showing the various tables of the database being used.

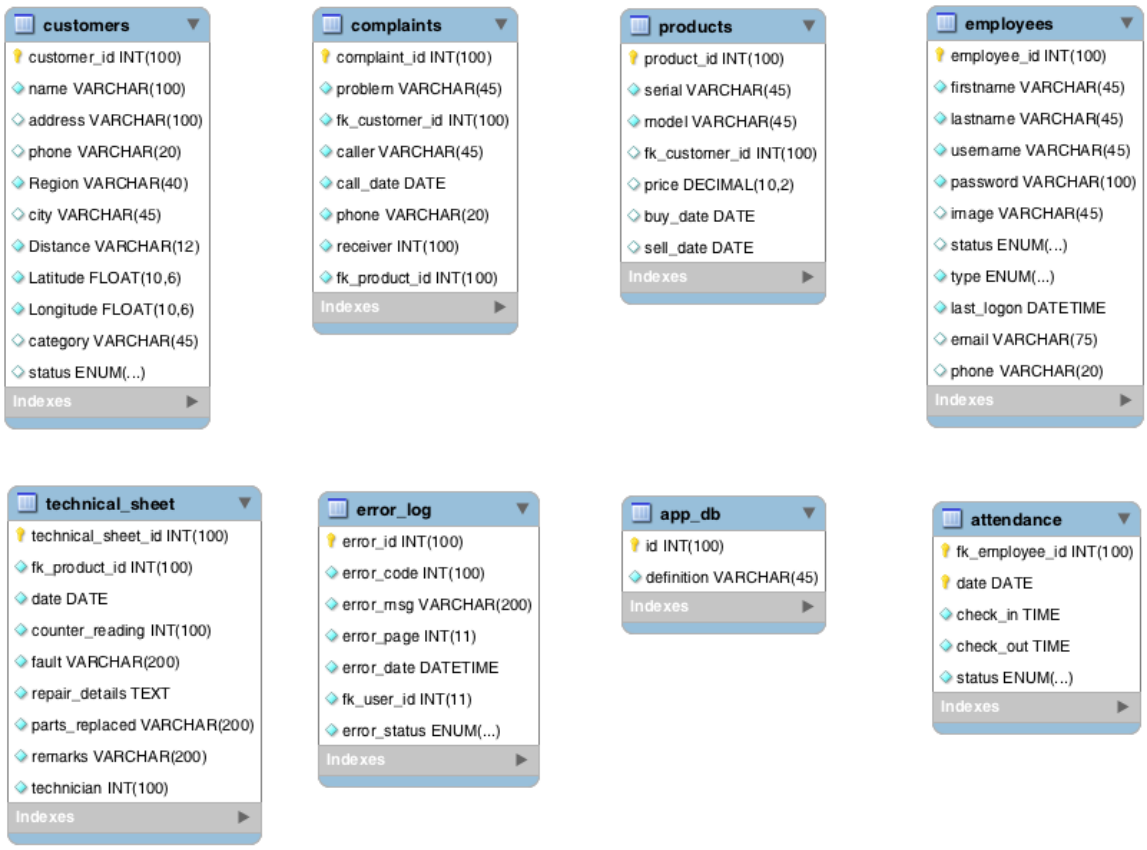

Figure 3.1 – Diagram showing different tables of the application database

The figure above displays all other tables in the entire database. The tables include *complaints*, *products*, *employees*, *technical_sheet*, *error_log* and *attendance*.

3.4 Integrating Google Maps API with existing application

To integrate Google Maps on any application, the Maps API JavaScript must be included using the `script` tag. Below is a snippet:

```
<script type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jquery/1/jquery.min.js"></script>

<script src="http://maps.google.com/maps/api/js?sensor=false"
type="text/javascript"></script>
```

Geocoding requests are processed through the following code format:

```
http://maps.googleapis.com/maps/api/geocode/output?parameters
```

The output is either in JSON or XML. The parameters must be `address`, `latlng`, or `components`, and `sensor`. The sensor indicates whether or not the request comes from a device with a location sensor, while the component restricts the results to come from a certain area. Based on the requirements of gLoc, below is a snippet:

```
$geocode=@file_get_contents('http://maps.google.com/maps/api/geocode/
json?address='.$add.',+'.$city.',+'.$state.',+'.$country.'&components=
country:GH&sensor=false');
```

3.4.1 Map Initialization

All initializations of the map including layers and different functionalities are declared in the jQuery `ready()` method. As such, an initial map with any data appears once the page has loaded. The main latitude and longitude variables to be used are that of the service provider, and that of the customer. The service provider's coordinates are declared as the variable `latlngBase`. On the other hand that of the customer is declared as `latlngCust`. In certain cases however, these variables are declared as Global Variables. Coordinates of the service provider are set to a default value, as we are certain it will not change. Initial bounds of the map are also set to the coordinates of each corner of Ghana's boundaries.

```
$(document).ready(
    function () { // Define the latitude and longitude positions
            // Latitude get from above variable
        var latitude = parseFloat("<?php echo $_POST['newLat']; ?>");
            // Longitude from same
        var longitude = parseFloat("<?php echo $_POST['newLong']; ?>");
        var latlngBase = new google.maps.LatLng(5.596278,-0.204062);
        var latlngCust = new google.maps.LatLng(latitude, longitude);

        var defaultBounds = new google.maps.LatLngBounds(
            new google.maps.LatLng(10.984335,-2.790527),
            new google.maps.LatLng(6.075011,1.120605));
```

3.4.2 Geocoding

```
<?php //code to find latitude and longitude when specific address is entered into form
    $add = urlencode($_POST['address']);
    $city = urlencode($_POST['city']);
    $state = urlencode($_POST['state']);
    $country  = urlencode($_POST['country']);
    $latitude = $_POST['lat']; //reverse geocode
    $longitude = $_POST['long'];//reverse geocode

    //GEOCODING

    $geocode=@file_get_contents('http://maps.google.com/maps/api/geocode/json?address='.$add.',+'.$city.',+'.$
    state.',+'.$country.'&sensor=false');

    $output= json_decode($geocode); //Store values in variable

    if($output->status == 'OK'){ // Check if address is available or not
        // echo the actual latitude and longitude inside Lat and Long textboxes
        // echo "<br/>";
        //echo "Latitude : "
        $lat = $output->results[0]->geometry->location->lat; //Returns Latitude
        // echo "<br/>";
        //echo "Longitude : "
        $long = $output->results[0]->geometry->location->lng; // Returns Longitude
    }


    //REVERSE GEOCODING

    $reverseGeocode=@file_get_contents('http://maps.googleapis.com/maps/api/geocode/json?latlng='.$latitude.','.$
    longitude.'&sensor=false')or die("url not loading"); //address lookup using reverse geocoding format


    $reverseOutput= json_decode($reverseGeocode);

    if($reverseOutput->status == 'OK'){//check is latlong is available or not
        //echo address based on latitude and longitude provided
        echo "<br/>";
        echo "Location (Address): ".$count = $reverseOutput->results[0]->formatted_address; //returns address
    };
```

Figure 3.2 – Code Snippet for Geocoding and Reverse Geocoding

3.4.3 Distance Matrix

```
//DISTANCE MATRIX
    var origin = new google.maps.LatLng(5.599414,-0.203619);
        destination = latlngCust; //"<?php echo $add; ?>";
        service = new google.maps.DistanceMatrixService();
        service.getDistanceMatrix(
            {
            origins: [origin],
            destinations: [destination],
            travelMode: google.maps.TravelMode.DRIVING,
            avoidHighways: false,
            avoidTolls: false
            },
                callback
        );
```

Figure 3.3 - Code snippet for distance matrix

```
function callback(response, status) {
    var orig = document.getElementById("orig"),
        dest = document.getElementById("dest"),
        dist = document.getElementById("dist");

    if(status=="OK") {
    //orig.value = response.originAddresses[0];
    //dest.value = response.destinationAddresses[0];
    //dist.value = response.rows[0].elements[0].distance.text;
        document.getElementById("dist").innerHTML =
        response.rows[0].elements[0].distance.text;
        //alert (response.rows[0].elements[0].distance.text);
    } else {
            alert("Error: " + status);
        }
}
```

Figure 3.4 - Code snippet for callback function of distance matrix

The Distance Matrix makes use of coordinates from the origin and coordinates from the destination. The origin coordinates are set as default origin. The destination coordinates are then retrieved from the geocoded results.

3.5 User interface

The home page gives us the ability to access every section of the application. Additionally it displays the name of the user logged-in.

- **Index Page:** This page will contain a link to all other pages. The main sections are *Customer, Products, Employees, Complaints, Administration* and *Help.* The last three sections mentioned will however not be of focus for GIS version.

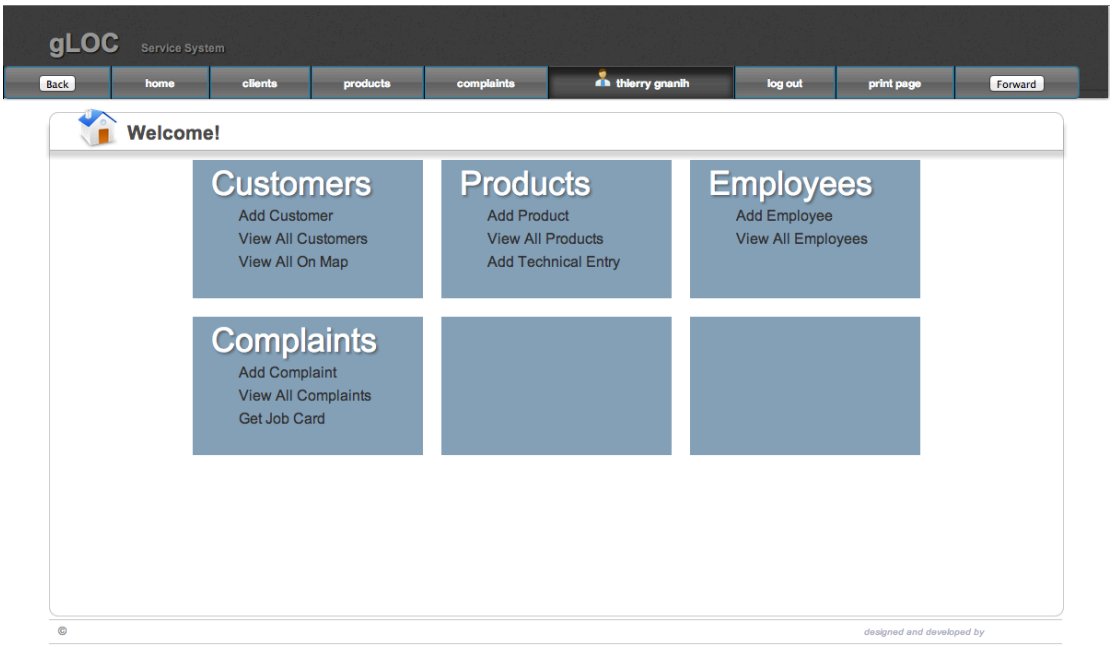

Figure 3.5 – Index page of the application

- **Add Customer:** The add customer section will allow authorized users to input new customer address, and geocode them in order to add details to the database.

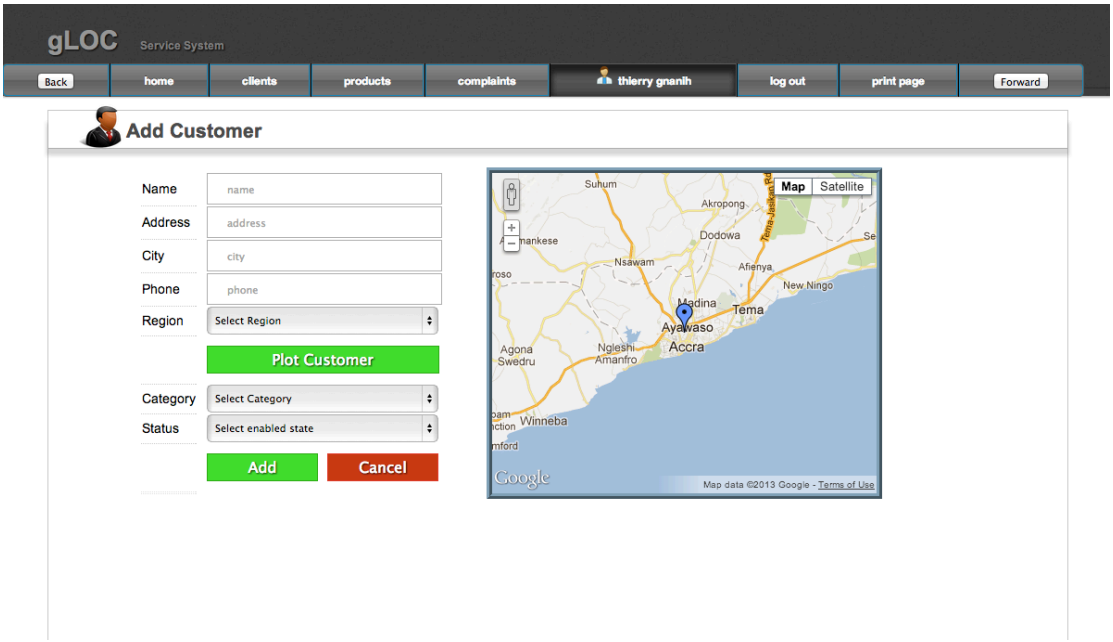

Figure 3.6 – Add customer page, which shows the initial page on load.

- **Auto-Complete feature:** The auto-complete feature used for the address and city input fields restrict searches to place in Ghana. The restriction is done when declaring the script.

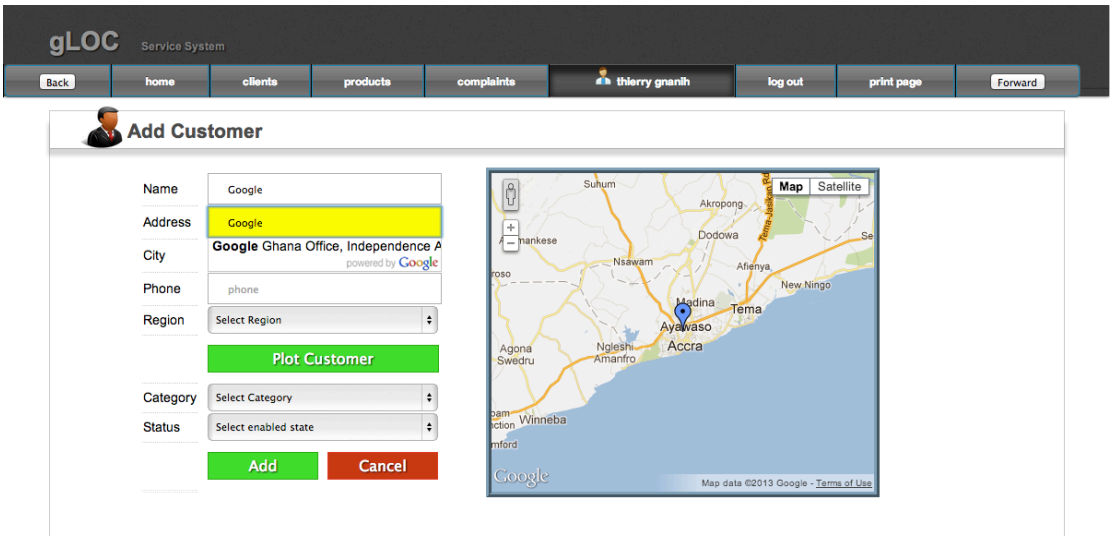

Figure 3.7 – Auto-complete feature for address input field.

- **Customer Details:** The customer details page will show four (4) different sections. The first will be a table displaying customer details such as name, address, phone etc. Next, there will be a section with a map. The map will showcase the default location of the service provider as well as the location of the customer. Both markers will be connected to each other by a trace. Detailed directions to the customer location will be provided next to the map. The directions panel will display an estimated distance and time needed to reach the destination. The fourth section will showcase details of products owned by the client.

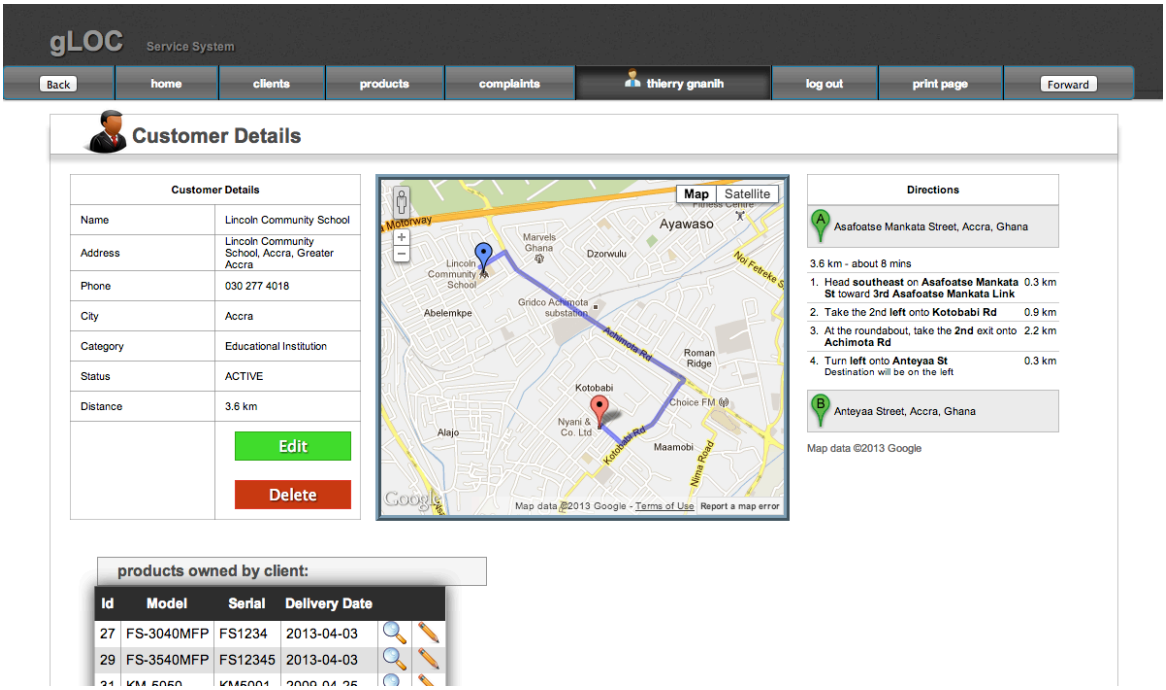

Figure 3.8 – Customer details page

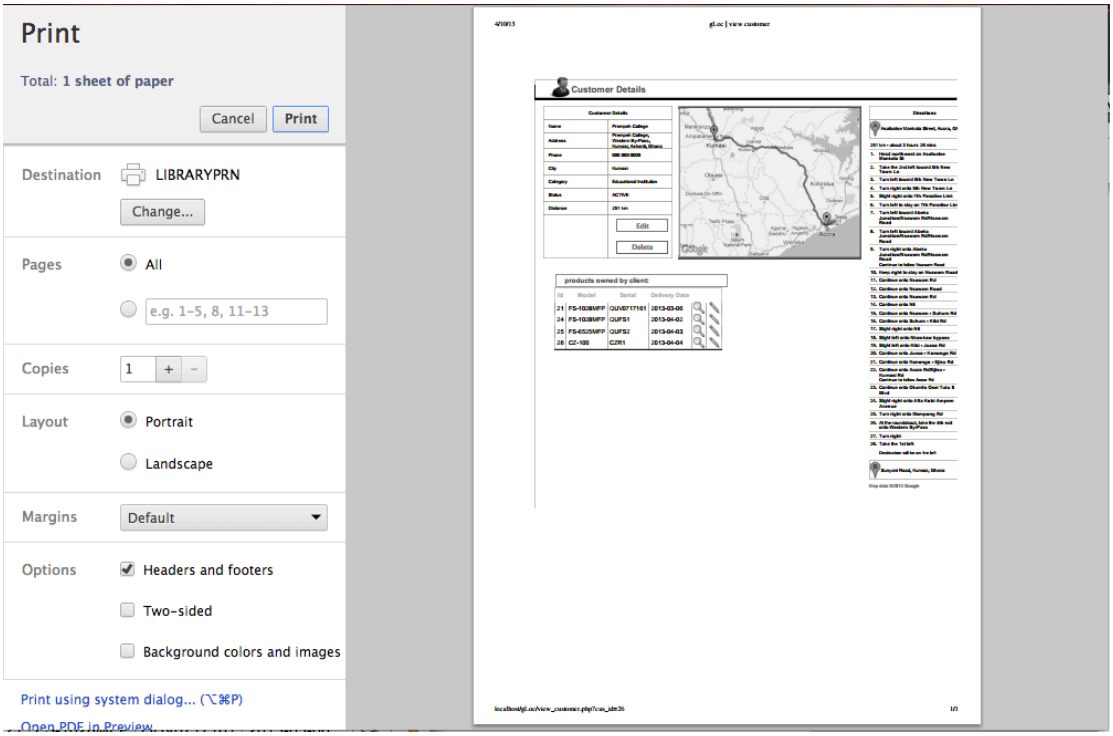

Figure 3.9 - Print-out of customer details

- **View all customers:** The view all customers page will display a map with markers for all customers available in the database. There will also be a feature that enables users to view customers by region. When the page loads, the map is displayed with bounds being around all markers. The bounds ensure that the user sees all markers. When the specific region is selected, the map will zoom in to display only that region.

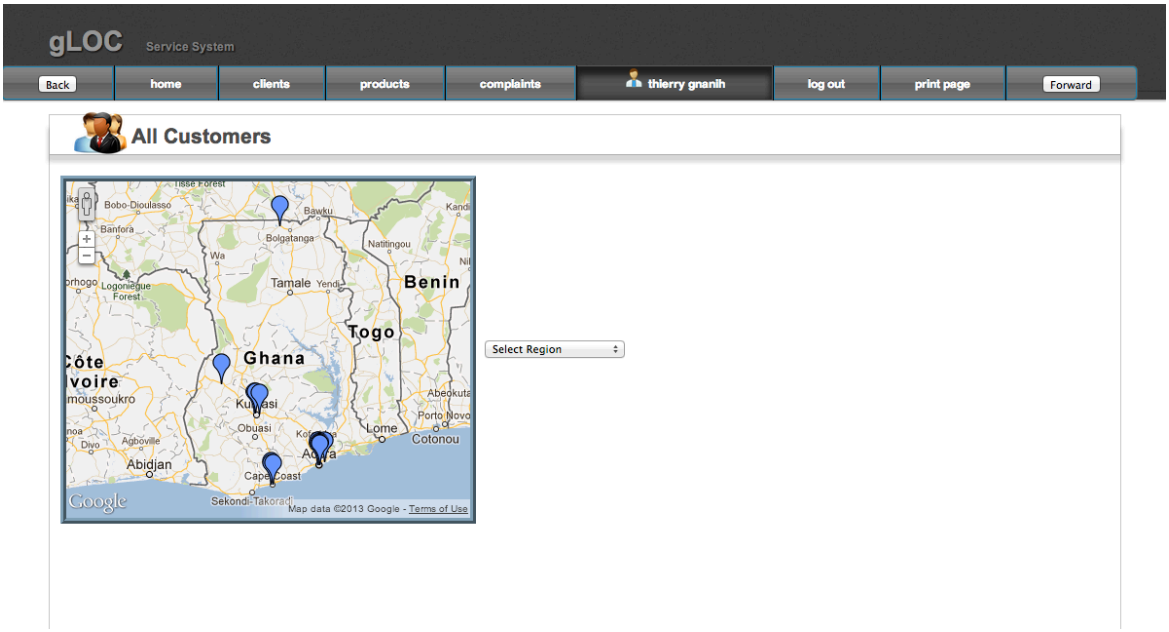

Figure 3.10 – Map displaying all customers in the database

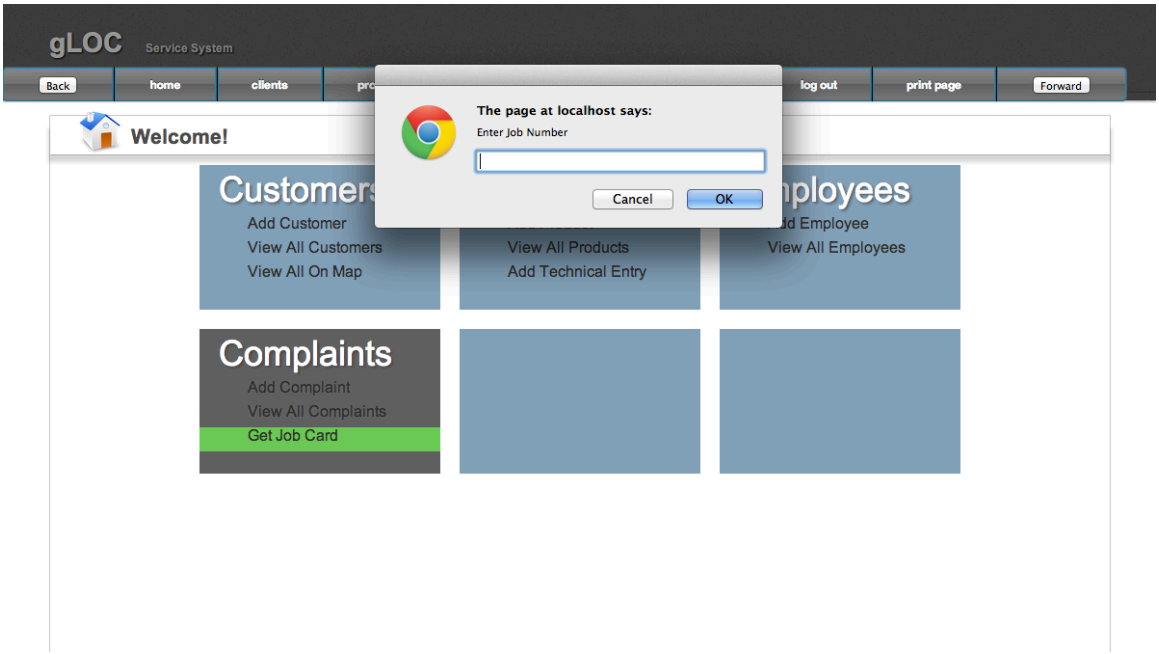

Figure 3.11 – Page to generate Job Card

3.6 Challenges

The major challenge faced in the implementation of this application had to do with Internet connectivity. Several pages of the application had to make calls to online Libraries, as the application made use of the Google Maps API. This challenge was tackled by saving most data generated into the database at first call, on the *Add Customer* page. Data will then be pulled using PHP. The only section that will have to be pulled from Google servers online will be the maps and directions.

Another challenge encountered is with the generation of report with Static Maps. This was a challenge because Google does not encourage the creation of reports with static maps generated through a local database. To solve this challenge, instead of requiring that the user generate a report in order to have customer details, the customer details page will be printable. Printable sections were defined to as to not include parts that will not be of use.

Lastly, organizing data on each page was also a challenge. Since the application was merged with an already existing application, every little change on the design would affect several other sections. Moreover, a template was used on the previous application. This made it demanding to customize every single page. I however, came through this by doing a lot of work on the CSS styles. Detailed attention was paid to styles used and additional styles added.

4. CHAPTER 4: TESTING AND RESULT

4.1 Compatibility Testing

Compatibility testing involved testing the application on different browsers. Although the application was designed and implemented using the Google Chrome browser on an Apple Operating System, it should not have performance difficulties working on other standard browsers.

While testing on the Firefox browser of an Apple Operating System, a few CSS distortions were noticed. Although this was expected, it did not affect the functionalities of the application.

4.2 Functional Testing

The application must be able to perform actions request by the user. All forms are able to accept inputs and transfer data to the database when an add action is requested by the user.

All maps available in the applications are able to perform geocoding functions and return accurate results based on specifications of the user.

Users were satisfied with functionalities offered by the application. General feedback received is that the functionalities provided were useful to the Ghanaian context. However, one main issue pinpointed by users is the fear that places on the map may not be fully accurate and the fact that Ghana did not have the best road name system.

4.3 Usability Testing

To test the usability of the application, random students were given the opportunity to try the features and give feedback. The general feedback

received is that the user-interface is tidy and it is easy to navigate the pages. Moreover, users appreciated the colors used. Colors were said to be easy on the eye.

5. CHAPTER 5: CONCLUSIONS AND RECOMMENDATIONS

In conclusion, this application will find success in solving the issues identified earlier. With the ability to print out the customer details page together with the complaint details, field technicians will be able to easily locate customer no matter how remote their area is. This will reduce frustration, the time spent making phone calls, and help to reduce communication costs.

Recommendations for future work include a mobile version of the application. Service personnel should be able to access the application on their mobile devices in order to view customer details. This will also reduce print costs. Another interesting feature that could be integrated into the application is an SMS feature. The application could have a feature that sends a link of the map showing customer location to devices of field technicians. Directions can also be considered. Better marker management is also recommended for future improvements to the application.

6. Bibliography

(n.d.).

Web Design Library. (2013). *Advantages of PHP Programming*. Retrieved April 2013, from WEB DESIGN LIBRARY: http://www.webdesign.org/web-programming/php/advantages-of-php-programming.21905.html

Eatock, J., Paul, R. J., & Serrano, A. *A Study of the Impact of Information Technology on Business Processes Using Discrete Event Simulation: A Reprise.* Brunel University, Department of Information Systems and Computing.

Drummond, W. J., & French, S. P. (2008). The Future of GIS in Planning: Converging Technologies and Diverging Interests. *Journal of the American Planning Association , 74* (2), 161-174.

FPDF Library. (2013). Retrieved April 2013, from FPDF: http://www.fpdf.org/

Gelinas, U. J., Sutton, S. G., & Fedorowicz, J. *Business Processes and Information Technology.*

GIS Lounge. (2013). *What is GIS?* Retrieved from GIS LOUNGE: http://www.gislounge.com/what-is-gis/

Google Inc. (2013, April 9). *Google Maps API*. Retrieved March 2013, from Google Developers: https://developers.google.com/maps/

Mandayam, G. *Geographic Information Systems: A planning tool for human service agencies.* Arizona State University, School of Social Work, College of Public Programs.

Miracle Service. (2013). *Routing and Mapping*. Retrieved April 2013, from Miracle Service...Service Management Made Easy: http://www.miracleservice.com/index.php/features/routing-mapping/

MSC. (n.d.). *After Sales Service / Customer Service.* Retrieved from Management Study Guide: http://managementstudyguide.com/after-sales-service.htm

ORACLE. (2008, March). *Integrating Oracle Application Express with Google Maps .* Retrieved March 2013, from Oracle: http://www.oracle.com/technetwork/developer-tools/apex/integrating-apex-with-google-maps-169532.pdf

ORACLE. (2001). *Oracle Maps.* Retrieved from Oracle Fusion Middleware: http://docs.oracle.com/cd/E14571_01/web.1111/e10145/vis_omaps.htm

ORACLE. (n.d.). *Top Reasons to Use MySQL.* Retrieved from mysql.com: http://www.mysql.com/why-mysql/topreasons.html