DEBRE MARKOS UNIVERSITY BURIE CUMPAS

DEPARTMENT OF COMPUTER SCIENCE

PROJECT ON *WEB-BASED CUSTOMER SERVICE MANAGEMENT AND BILLING SYSTEM FOR BURIETOWN WATER SUPPLY SERVICE*

BY:

1. *Ayehush Kebede*..... *TER/007/07*
2. *Abaynesh Berie*.....*TER/037/08*
3. *Melkamu Deguale*..... *TER/062/08*
4. *Shegaw Yibeltal* *TER/070/08*

Adviser: Kerebih .G (MSc)

BURIE, ETHIOPIA

JUNE, 2019

Table of Contents

List of Tables

List of Figures

Acronyms

BWSSO……. Burie Water Supply Service Office

HDD……Hard Disc Drive

RAM……Random Access Memory

CPU……Central Processing Unit

PC……Personal Computer

CD…….Compact Disc

DVD…….Digital Versatile Disc

UML…….Unified Modeling Language

DBMS……Database Management System

MYSQL…….My Structured Query Language

PHP……Hypertext Pre Processor

CSS……Cascading Style Sheet

HTML……Hypertext Markup Language

OOSAD……..Object Oriented System Analysis and Design

UC……..Use case

DB……Database

HTTP……..Hypertext Transfer Protocol

CHAPTER ONE:

1.1. INTRODUCTION

Water supply management system is powerful, flexible easy to use and designed to deliver real conceivable benefits to office. we would help to solve the problems by replacing the way of desktop application system of Burie water supply services office into web based computerized system. BWSSO has many activities, such as customer's registration, calculating bill based on their customer information, since every activity performed in single computers so, we try to reduce these problems and enable the office system to have fast service to there by designing web based service management system for them.

1.2. Background

Background of the organization

Burie water supply service office is a water supply organization which is found in Burie Town. The organization is established in 1971E.C. At that time the office had seven employees, currently the organization have 52 employees and around 4500 customers which register to use the service. This office done many activities like, customer registration, Bill process calculating, and meter number registration and viewing reports for them, taking customers maintenance order and respond it.

Background of the project

Until 2001E.C the organization follows manual based office system which means, they follow traditional way of giving service for their customer. At 2001 E.C the office develop its own desktop application system. BWSSO use a desktop application to perform any activity, therefore in case of its desktop application the organization was challenged with problems of performance, so customer can't get quick access. To solve such problems we are going to develop web based water billing and customer management system for BWSSO.

1.3. Statement of the problem

There is already a computer aided system Burie water supply service office which is a standalone desktop application. In the system every activity is performed on a single desktop. Burie water supply service office has many activities such as registration of customer, bill calculate, meter reading, bill print, register materials. But,

- ✓ Since it is desktop application there is a performance problem on the organization, customer wastes their time and money.
- ✓ Since the system is installing in a single computer customers do not get quick access, materials do not distribute fairly.
- ✓ Another problem of Burie water supply service office is if a customer had no enough balance on his hand he/she needs to go home and get more many because there is no way of checking mechanism before going to office to pay the water billing.

Therefore, such problem is present currently from this we try to change the desktop application system in to web based system using today's technology. Then by using web based system every activities of the existing system will provide a quick access.

1.4. Objective of the project

1.4.1. General objectives

The general objective of the project is to develop web based customer management and billing system for Burie water supply service office.

1.4.2. Specific objectives

The specific objective of our propose systems are listed below:-

- ✓ To give fast service for customers
- ✓ To design database for BWSSO
- ✓ To generate bill and send to the customer.
- ✓ To receive maintenance service request from the customer.
- ✓ To generate appropriate report automatically
- ✓ To provides timely information for their employees and customers.

1.5. Scope of the project

Scope of the project for web based system is applicable for web based customer management and biweb application for customer are:-

- ❖ Customer registration and retrieval
- ❖ Registration of the maintain material
- ❖ Registration of meter reading
- ❖ Registration of the payment
- ❖ Generate report
- ❖ Bill calculation and printing

Even though customers can check how much they will pay in each month from home they cannot process online payments, because the way of processing online payment is not include in our scope.

1.6. Significance of the project

Our proposed system provides the following significance:

- Enhance employee morale of the organization by providing quality service
- Improve the confidence of the system user
- The system will save the customer's time and cost when they want to access service from the organization, the customers uses the system safely and gives comfort for them.
  - It increases performance of the organization since every activity can be done in the fraction of second without any difficulty.
  - The system makes the office more efficient and profitable in a short period of time.
  - Reduce man power of the organization.

1.7. Methodology

1.7.1. Systems Requirement

For the new system to be efficient there are a requirement for specific hardware and software.

Hardware Requirement

Computer: almost all tasks of our project are performed on computer.it includes:-

- Server-Intel(R) Core(R)i5 CPU @ 2.7GHz with minimum of 2 GB RAM and 1TB of HDD(PC)
- Client- Intel(R) Core(R)i3 CPU N2830 @ 2.16GHz with minimum of 4GB RAM and 80 GB HDD.

Flash: required for data movement to store and transfer data from one pc to other pc.

CD/DVD: necessary for the movement of relevant data and for backup and recovery mechanism.

Software Requirements

- Browsers: since our system is web based, it is very necessary requarment.it includes Internet Explorer, Mozilla Firefox and Google chrome.
- Notepad++: to edit programs
- Adobe Photoshop (CS6): for editing images.
- EdrawMax9.3: for designing UML diagrams
-
- Operating system:

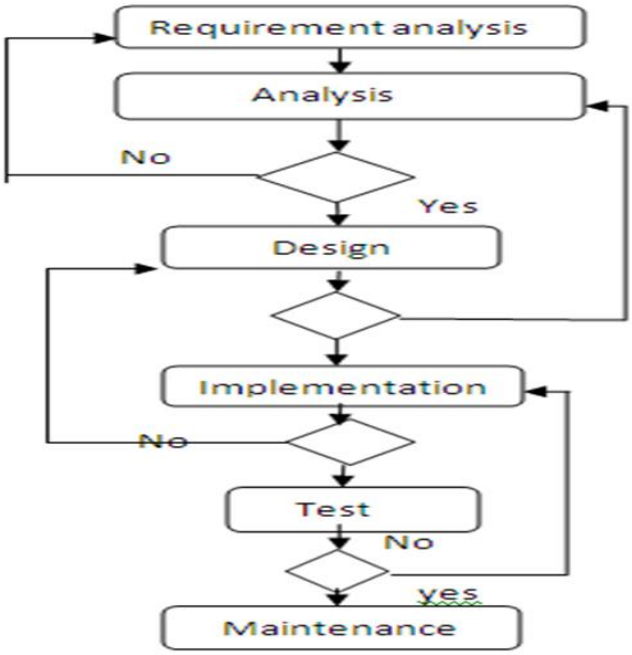
MS-windows 7, 8, 10: will be used for the system since it is readily available in laboratories

, 2008 server will be used for the system.

Database management software (DBMS): is the mandatory one for the new system. To implement the database easily, (MySQL) is recommended.

Software Process

The team will use the iterative method for the development of the project because of its suitability to develop a good and efficient system since it makes as backward and forward for solving error.



Programming language

- There are several types of web programming language that are used for making a site more dynamic. But, for this project we are choosing PHP scripting language to design our database, CSS, java script in html. Because:
  - It's fast and easy
  - It's cross platform
  - It's free(open source)
  - Platform Independent
- We used HTML, CSS and JavaScript in client-side or front-end.
- We used PHP and MYSQL in a server-side or back-end.

1.7.2. Data Collection methodology

For conducting our proposed system we used the following data collection methodology.

- **Observation:-**use this method to get the right information about the organization and also understand how the existing system works.in this process we tried to investigate the information by making our selves participant in the process. Such as we observe customers buying cash, registering, bill calculation, customer complain.
- **Interview: -** interview for interviewing the manager and employees for recognizing the existing working procedure of the organization. So we were able be to gather more information about the organization by interviewing what the organization has the problem and related to the customers compliant.

CHAPTER TWO:

2. SYSTEM ANALYSIS

2.1. Overview of the existing system

The existing system is practice at BWSSO and the system uses desktop application. New customer registers to the system first on paper by giving full information to get water service from system. Later on the customers register on the computer. Billing system is the other activity of the system in time of customer payment date, the bill printer print reading paper then gives to the reader. The reader read each customer water consumption and record on paper. The customer pay to bill seller and get recite for their water consumption payment. Customers not see their payment by any mines until they are come in the system to pay for their consumption. This leads customer exposed for unexpected expense because customer not know the exact amount of payment for their consumption. Therefore, to pay exact money they must return to their house and come again with exact money

Users of the existing system

Customer: Is the actor who is act into the system to get service.

- ✓ Order or request maintenance service.
- ✓ View own report.

Manager: Is an employee, who works on customer service office, which have responsible for view any report (comment), approve employee, such as Technician, Bill officer, Meter reader.

Meter reader: is a person who can read the monthly water consumption manually and send the corresponding value to the bill officer.

Technician: An employee who controls mainly the overall maintenance service of customers, specifically she/he have the following activities.

- ➢ Can View service report that requested by the customer then he/she can provided maintenance/installation service for the customer manually.
- ➢ Approve any requested activity and new customer.

- Generate service delivery report.

Bill officer: -is an employee, who works on customer service office, who has the following responsibilities.

- Register new customer.
- Generate bill report

2.2. System requirement Specification

2.2.1. Functional Requirement

Functional requirement is concerned with actual performance of the system that is going to be developed. Also it does describe the functionality or service provided by the new system. It also describes the interactions between the system and the user.

The new system is expected to provide the following functionality.

User: - The system should allow the users to login with their username and password.

The system should allow the users to logout after using the system.

Customer:-The system should allow the customer to order the maintenance service, and view the report.

Administrator: - The system should allow the administrator to create account, to update account, to delete account for the user except new customers.

Bill officer:-The system should allow the bill officer to register new customer, generate bill and register monthly water consumption.

Technician:-

- The system should allow the technician view maintenance request.
- The system should allow the technician Approve any request activity and Ap new customer.
- The system should allow the technician Generate service delivery report.

Manager: -

- The system should allow the Manager View report.
- The system should allow the Manager approve the accounts for employee, such as Technician, Bill officer, Meter reader.

2.2.2. Non-Functional Requirement

Non-functional requirements are the ones that relate with the reliability of the system. These requirements indirectly affect the performance of the system. They are used to build quality of the system. The following are the non-functional requirements for the new system.it is also known as Technical requirement.

- ✓ **Technical requirement:-**
- ❖ Security:- In order to make the system safe from unauthorized users the system will use a log in account to differentiate authorized users from unauthorized users of the system.
  We will use also session to restrict users from accessing page without their privilege so we will give session time that it will expire after the time passes.
- ❖ Error handling:-The system shale handles errors by giving error-message.
- ❖ Portability:-The system must be designed for platform independent and the system supports every operating system
- ❖ Availability:-the system shall have available at 24 hours for the customer service. The system shall have high availability.
- Accessibility:-Since the system is easily accessed it is easily used everywhere in which internet connection is available.
- ❖ Performance: -The system should have a quick response time in 5ns for any request made. It is expected that the software would perform functionally all the requirements that are specified by the organization.
- ❖ Accessibility:-Since the system is easily accessed it is easily used everywhere in which internet connection is available
- ❖ Compatibility and Portability: the system shall run in any computer system, regardless of the operating system and light weighted. Thus, having this in mind, our system is portable at all, because we have implemented, the project using an object oriented language.

The technical requirement of the system:

- The interface of the system will be user friendly easy to use.
- The interface will display error message if it detects invalid input
- The system will deny unauthorized accesses to the system domain
- The system will provide help for the user.
- Training the users to access the system.

2.2.3. Business Rule

A business rule is effectively an operating principle or policy the software must satisfy. It often relevant to access control issues, business calculations, or operating policies and principles of the organization. Therefore, our new system has the following business rules.

- The bill officer must be Registers water consumption value, then Generate bill report to the customer.
- Meter reader reads the water consumption value and send to the bill officer.
- If customer registered in the organization then, it must be approved by the Technician in order to gate any service from the organization.
- The customer shall notify the authority as soon as he/she is aware that the meter is broken.
- If customer orders the services and the technicians provide services for them.
- Apply payment based on the following rules in the table:-

➢

level	Consumption range	Consumption	rate	Customer name
1	0-5	5	6.75	residential
2	5.1-10	5	9.50	residential
3	10.1-25	15	11.40	residential
4	25.1-40	15	14.40	residential
5	Above 40.1	15	16.75	residential
1	0-5	5	10.75	private
2	5.1-10	5	12.65	private
3	10.1-25	15	16.15	private
4	25.1-40	15	19.25	private
5	Above 40.1	15	24.40	private
1	0-5	5	9.90	government
2	5.1-10	5	13.15	government
3	10.1-25	15	15.25	government
4	25.1-40	15	17.40	government
5	Above 40.1	15	20.60	government

2.2.4. Change cases

- ✓ Likely future changes (update) to either the system, in terms of its capabilities and properties are computable with the new version.
- ✓ The system will promote related international rules and regulations.

level	Consumption range	Consumption	rate	Customer name
1	0-5	5	4.5	residential
2	5.1-10	5	6.5	residential
3	10.1-25	15	8.2	residential
4	25.1-40	15	10.1	residential
5	Above 40.1	15	12.3	residential
1	0-5	5	7.2	private
2	5.1-10	5	8.6	private
3	10.1-25	15	11.15	private
4	25.1-40	15	15.26	private
5	Above 40.1	15	18.4	private

1	0-5	5	6.3	government
2	5.1-10	5	9.14	government
3	10.1-25	15	12.7	government
4	25.1-40	15	14.4	government
5	Above 40.1	15	18.60	government

2.3. System requirement analysis

Burie water supply service office system brings into play an Object Oriented System Analysis and Design (OOSAD) to model the system and describe the data. The reason why we chose an object oriented approach is because of the analysis is made on the classes and interaction among them in order to meet the functional requirements.

2.3.1. Actor and use case Identification

Actor Identification

Actors are parties that outside the system that have direct interaction with the system.

Actors are defined in a use case diagram as a stick figure and represent external factors that will provide interaction with the system.

We have five actors that interact with BWSSO are listed below:-

- **Customer**: Is the actor who is act into the system to get service.
  - Order or request maintenance service.
  - View his own report.
- **Administrator**: Is the actor that has the privilege of managing the overall activities of the user account such as.
  - Create account for all actors except new customer.
  - Delete account.
  - Update account.

- **Manager**: Is an employee, who works on customer service office, which have responsible for view any report(comment),approve the accounts for employee ,such as Technician, Bill officer, Meter reader .
- **Technician:** An employee who controls mainly the overall maintenance service of customers, specifically she/he have the following activities.
  - Can View service report that request by the customer then he/she can provided maintenance/installation service for the customer.
  - Approve any request activity and Approve new customer.
  - Generate service delivery report.

- **Bill officer**: -is an employee, who works on customer service office, which have the following responsibilities.
  - Register new customer.
  - Generate bill report
  - Register water consumption value

Use case Identification.

- Order service such as maintenance or installation
- View report
- Create account
- Delete account
- Update account
- Approve new customer
- Generate Bill
- Login
- Logout
- Generate water consumption value
- Receive Maintenance

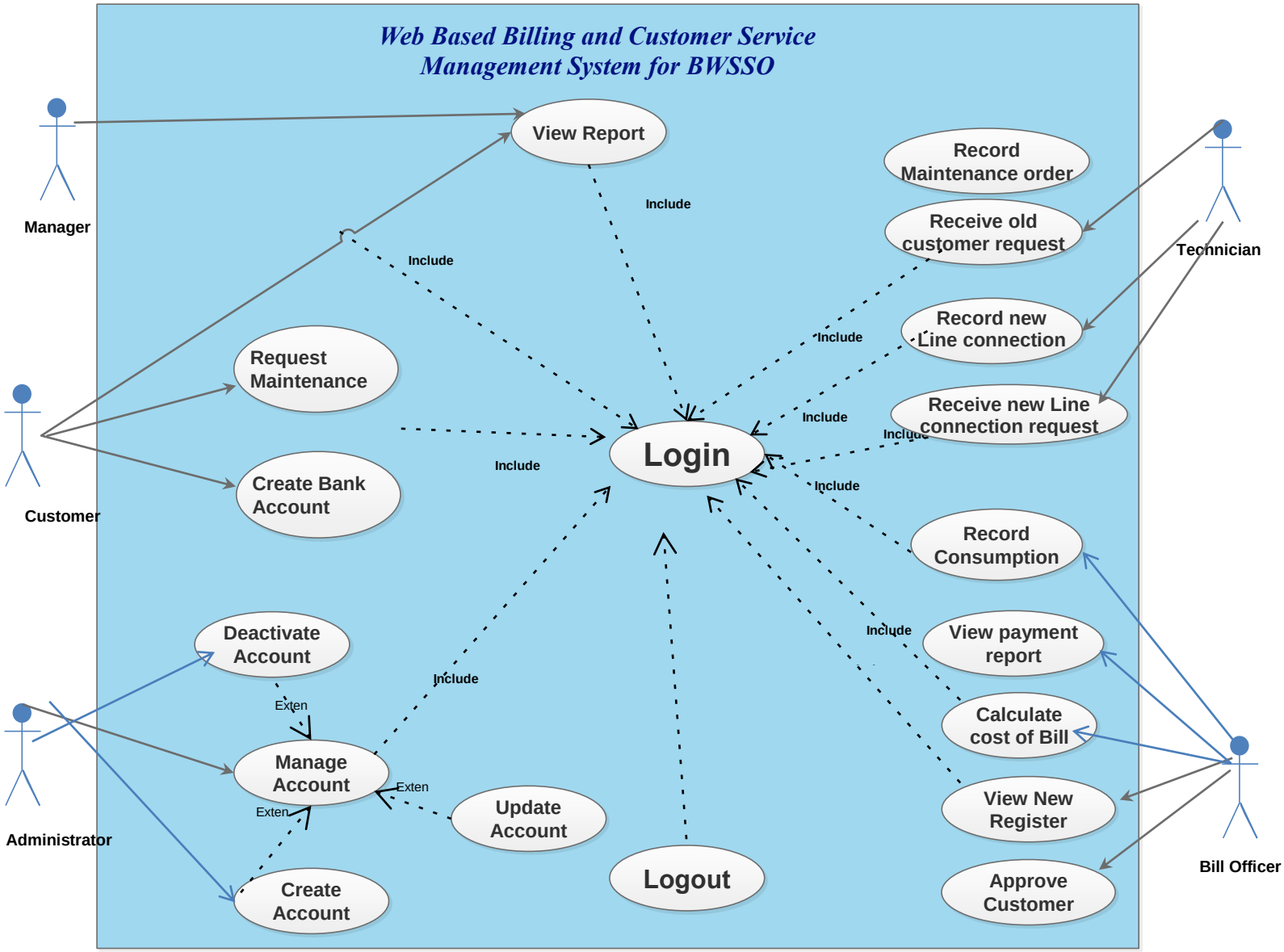


Figure 2. 1 Use case Diagram

Use case Description

Table 2. 1 Customer registration

Use case name	Registration of Customer	
Use case number	UC1	
Description	It allows Customer To Register The DB, a customers Requests Registration To Get New Water Connection From BWSSO.	
Actor	Bill Officer	
Pre-condition	The customer Should have contact with the Bill Officer to get the new water connection of BWSSO.	
Basic course of action (Flow of event)	User Action	System Response
	1 When the customer wants to register to the DB. 2.The Bill Officer Open the HomePage and click Apply Register link. 4.Enter the correct and all necessary information's of the customer.	3.The system displays the Customer register form page. 5.System Validate it. 6.Initiate the system to send to the organization. 7.The use case ends.
Post-condition	The customers can join to the organization and get the online service.	
Exception Handling	4.If The filled Customer information is invalid. 5.1. The system display error message. 5.2.Goto 3 to display Form Page	

Table 2. 2 Customer request service

Use case name	Request maintenance service	
Use case number	UC2	
Description	The customer enables to select the technical link and fill the request.	
Actor	Customer	
Pre-condition	Must have user name and password	
Basic course of action (Flow of event):	User Action	System Response
	1 The customers want to fill the request by selecting technical link. 3.The customer fills the inputs his/her required information. 5.The customer click save button.	2.The system displays the technical form page. 4.The system displays the customer's request information. 6.The use case ends.
Post-condition	Maintenance order record to the system	
Exception Handling	3.if The customer fill the incorrect request 4.1The system displays error message. 4.2 Go to2to select technical form page.	

Table 2. 3 Customer view report

Use case name	View report	
Use case number	UC3	
Description	The customer used to view report.	
Actor	Customer	
Pre-condition	The data should be submitted in BWSSO database	
Basic course of action or Flow of event	user	System
	1.The customer selects view report link. 3.The customer press alternative view icon.	2. The system displays the view report page. 4.The system displays their data to customer from BWSSO database. 5. The use case ends
Post-condition	The report is viewed by the customer	

Table 2. 4 Manager View report

Use case name	View report	
Use case number	UC4	
Description	The manager used to view report	
Actor	Manager	
Pre-condition	The data should be submitted in BWSSO database.	
Basic course of action Flow of event	user	system
	1. The manager selects view report link. 3. The manager press alternative view icon.	2. The system displays the view report page. 4. The system displays their data to manager from BWSSO database. 5. The use case ends.
Post-condition	The report is viewed by the manager.	

Table 2. 5 Administrator to delete account

Use case name	Delete account	
Use case number	UC5	
Description	It Allows administrator to delete user account	
Actor	Administrator	
Pre-condition	To delete the user account must be registered in the database	
Basic course of action (Flow of event)	User Action	System Response
	1. The administrator wants to delete account. 3. The administrator press on delete button. information.(alternative 4)	2. The system displays the delete account page. 4.The system validates the information.(alternative 4) 5. The account is deleted from the system.
Post-condition	3 .If the selected account is invalid. 4.1 The system displays error message. 4,2 Go to2to select the delete account again	

Table 2. 6 Administrator create account

Use case name	Create account	
Use case number	UC6	
Description	Used to create account for users	
Actor	Administrator	
Pre-condition	The user should be member of the BWSSO organization	
Basic course of Action	User Action	System Response
	1.The administrator selects create account link. 3.The administrator fills the required information and submits it.	2.The system displays create account page. 4. The system validates the information(Alternate 4). 5.The system registers the users into the system. 6.The use case ends.
Post-condition	The account is successfully created	
Post-condition	3. if administrator Invalid information entry. 4.1 The system displays error message 4.2 Go to2to select the create account again	

Table 2. 7 Bill officer to calculate cost of bill

Usecase name	Calculate cost of bill	
Use case number	UC7	
Description	The system calculates the cost of bill	
Actor	Bill officer	
Pre-condition	The bill officer should get the current water meter reading value of the customer.	
Post- condition	The customer bill will be calculated	
Basic course of action (Flow of event):	User Action	System Response
	1.Bill officer open the homepage. 2. Bill officer enter user name and password 4.Bill officer enter current water meter reading of the customer and other important information.	3.System validates it. 5.Initiate the system to calculatethe cost(Alternate 6.End use case
Post-condition	4.If Bill officer enter Invalid information 5.1 The system displays error message 5.2 Go to2to validate	

Table 2. 8 Use case for login

Use case name	Login	
Use case number	UC8	
Actor	Bill officer, technician supervisor, customer, administrator, meter reader, manager	
Description	A member login to BWSSO uses their appropriate user name and password.	
Precondition	Must have valid username and password	
Basic course of action	User	system
	1.Open home page 3. Enter username and password. 5. User login to the system.	2.From home page there is login form with sign in button 4.System validates the address (Alternate 4). 6.End use case.
Alternative action	3.If the user name and password are not correct 4.1 The system displays error message 4.2 Go to2to select the sign in button again	
Post condition	The employees enter to the System	

2.3.2. Sequence diagram

A sequence diagram shows an interaction arranged in time sequence. In particular, it shows the instances participating in the interaction by their "lifelines" and the stimuli that they arranged in time sequence. It does not show the associations among the objects.

Sequence diagrams are used to model the logic of usage scenarios or the description of the potential way the system used. Sequence diagrams are a great way to validate and flesh out the logic of use case scenarios and to document design of the system.

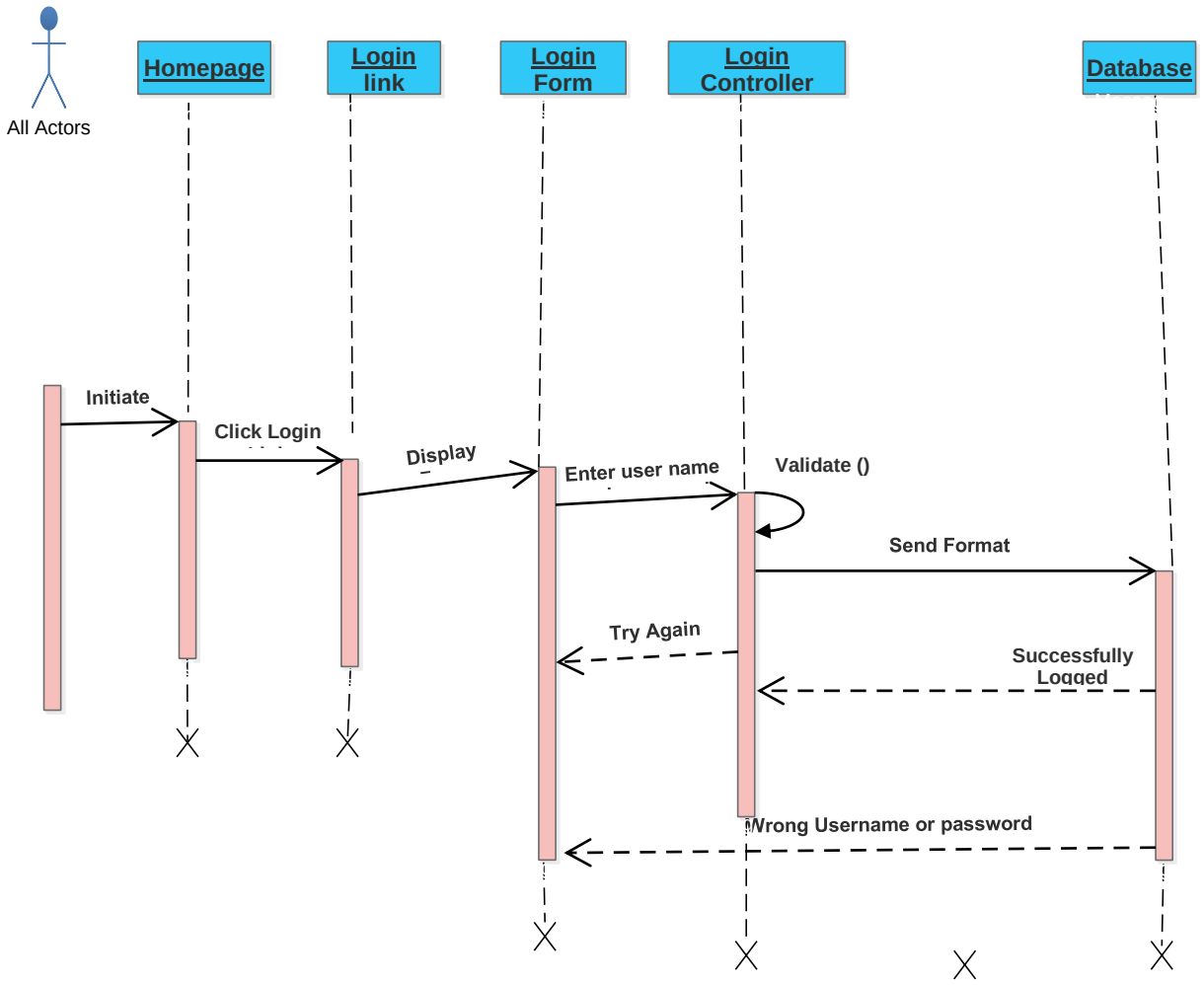


Figure 2. 2 UML sequence diagram for login

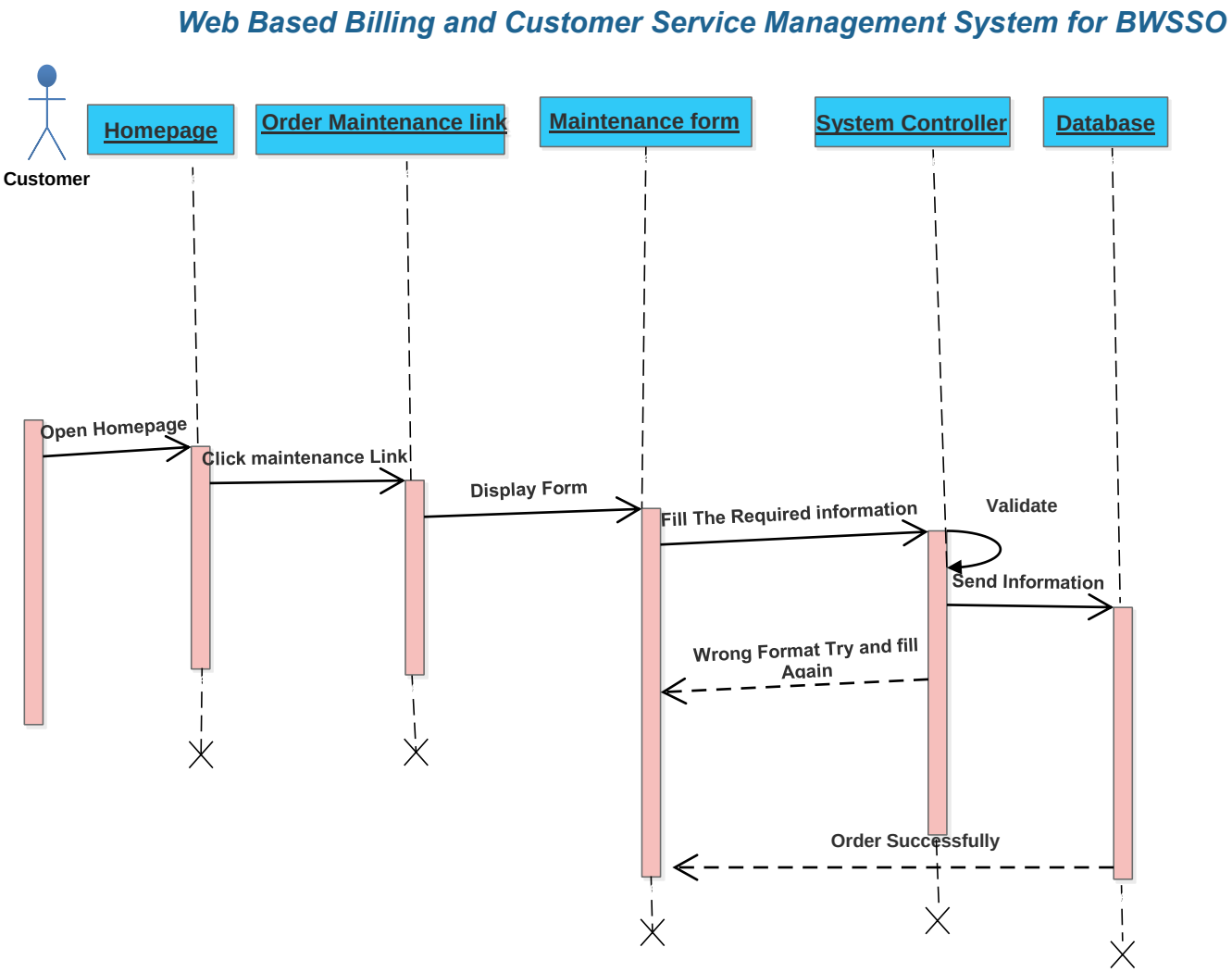


Figure 2. 3 UML sequence diagram for order maintenance

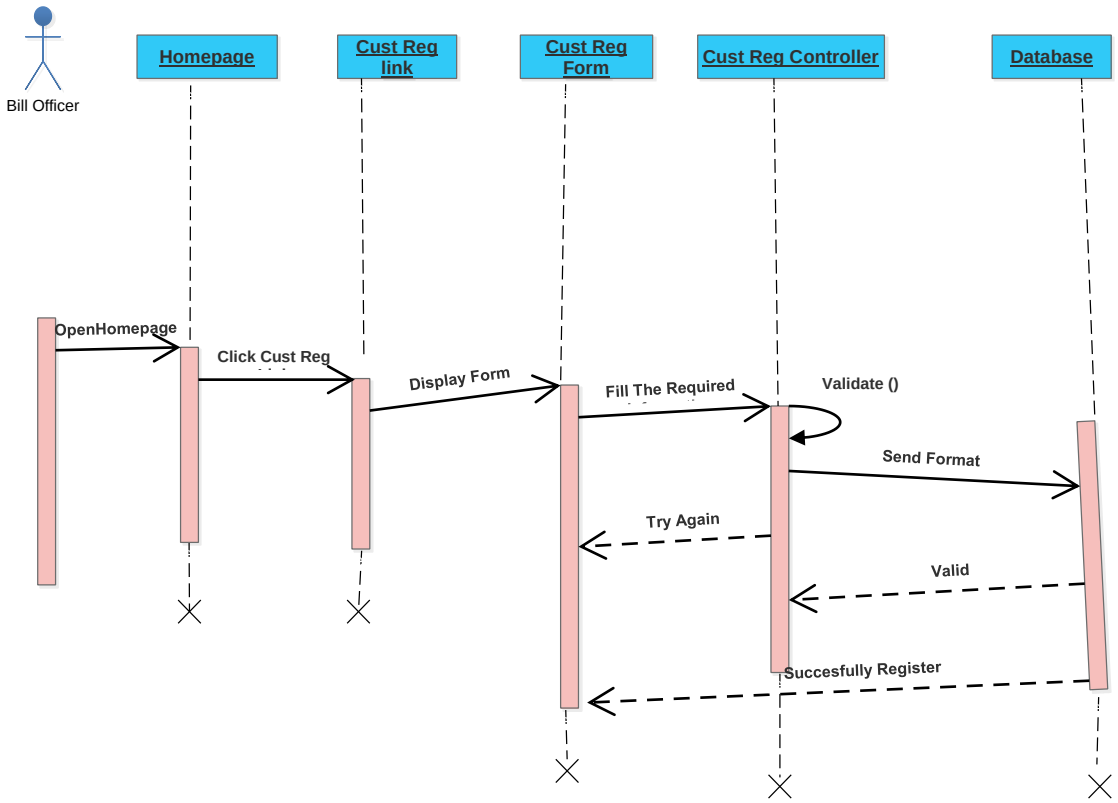


Figure 2. 4 UML sequence diagram for registration

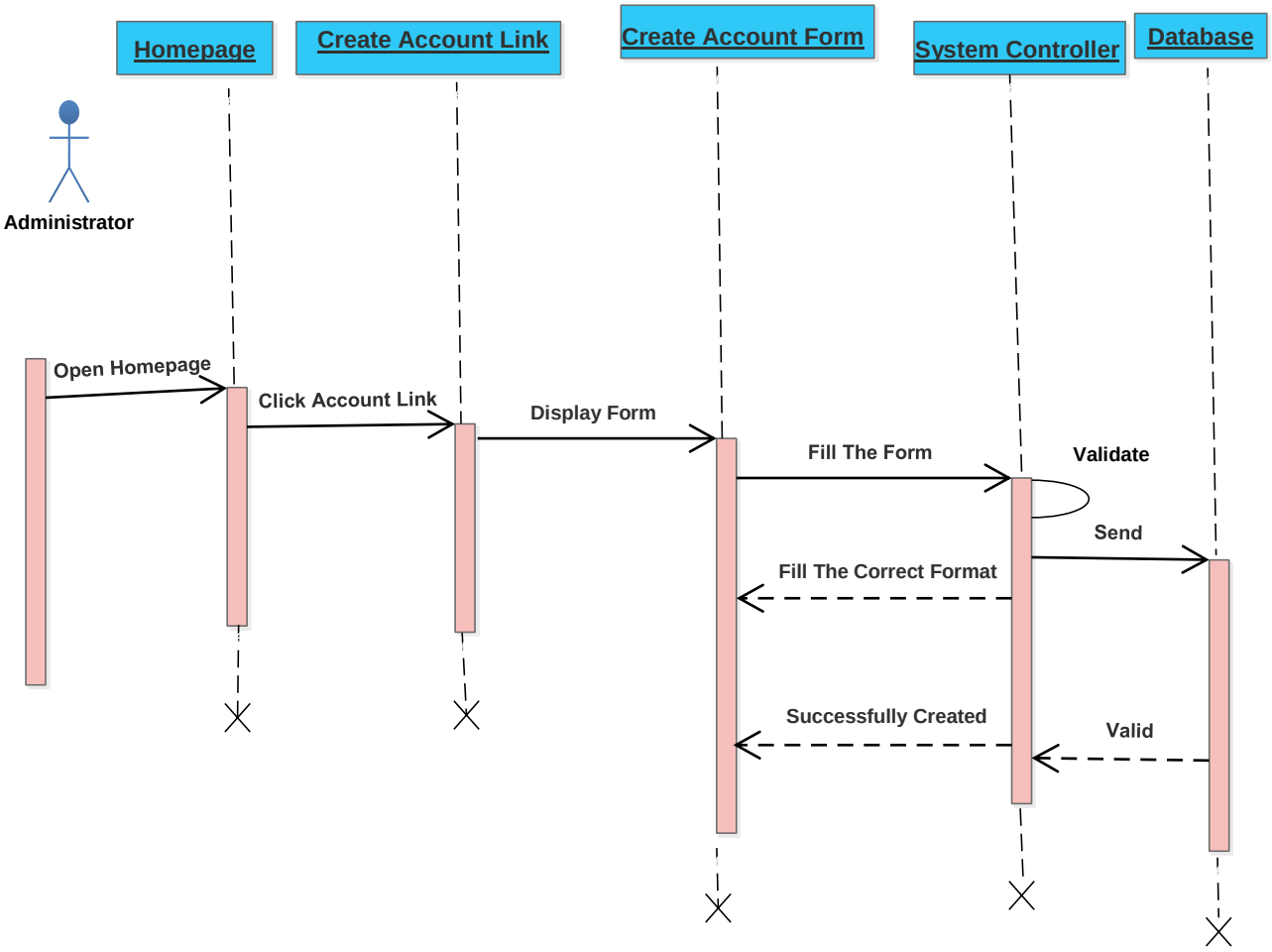


Figure 2. 5 UML sequence diagram for crate account

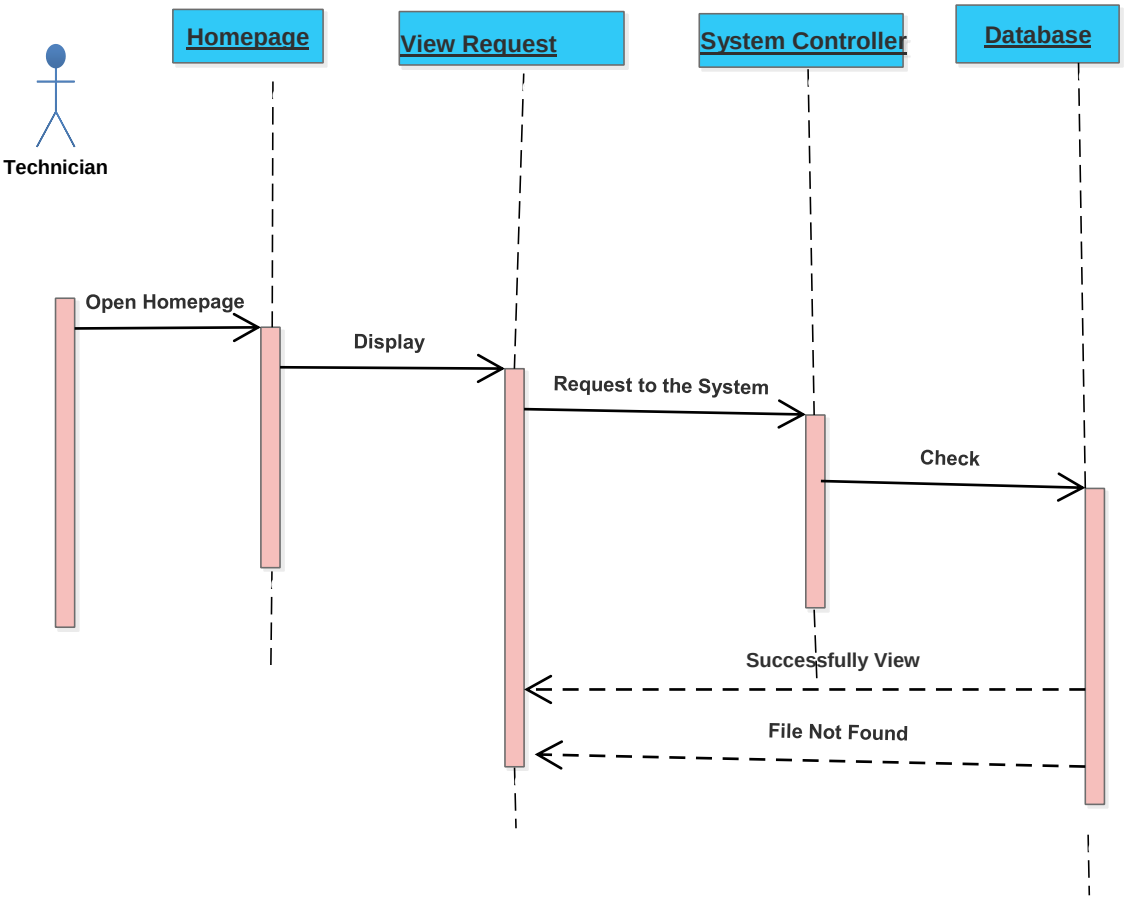


Figure 2. 6 UML sequence diagram for view order or request

2.3.3. Activity diagram

An activity diagram is a variation of a state machine in which the states represent the performance of actions or sub activities and the transitions are triggered by the completion of the actions or sub activities. It represents a state machine of a procedure itself. Activity diagrams model is a high level business or processes or transitions between states of a class. In this activity diagram we tried to document the flow of logic for the major business processes of this project.The activity diagram below shows the users activity performed on the customer management system for BWSSO.

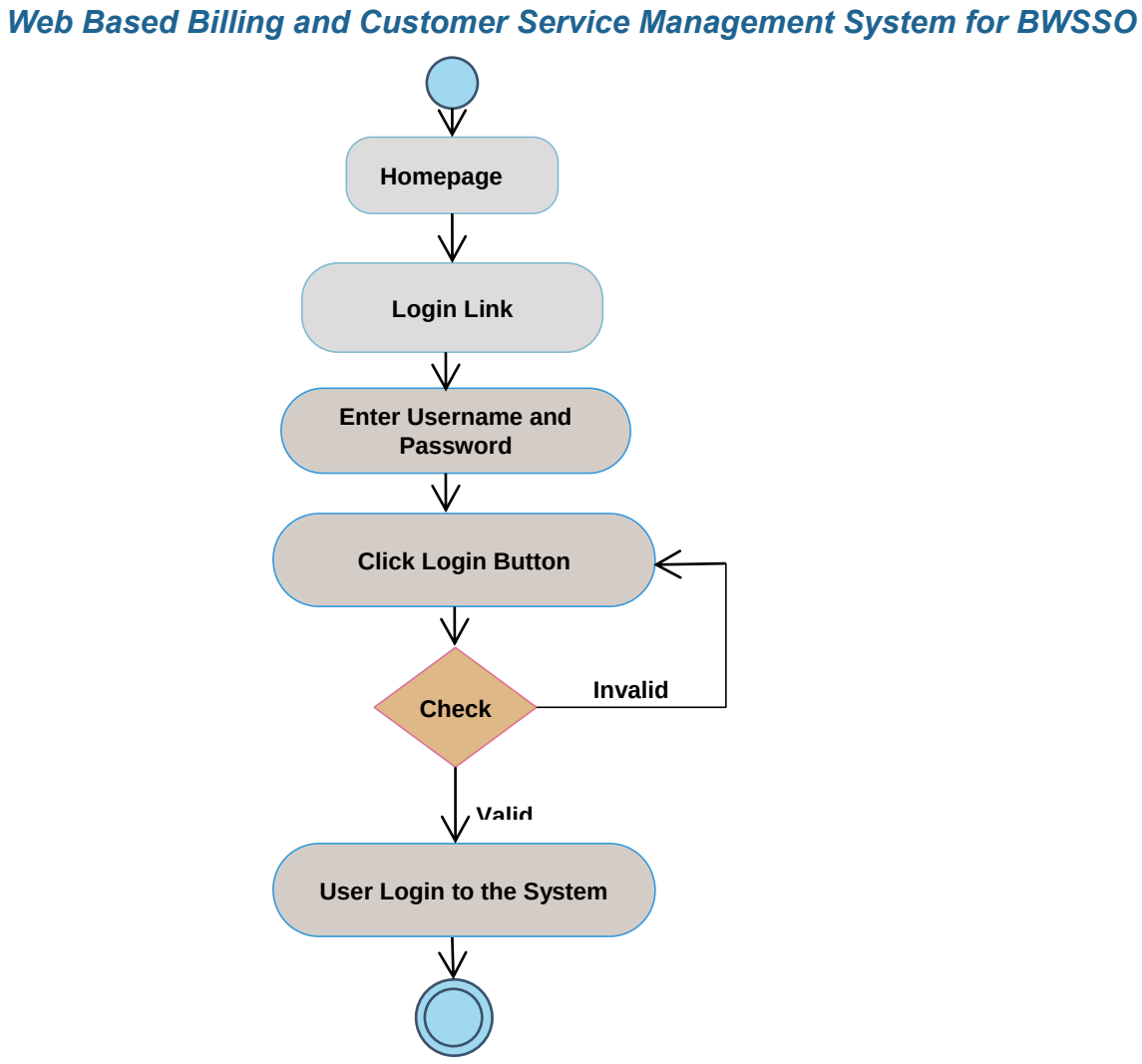


Figure 2. 7 activity diagram for login

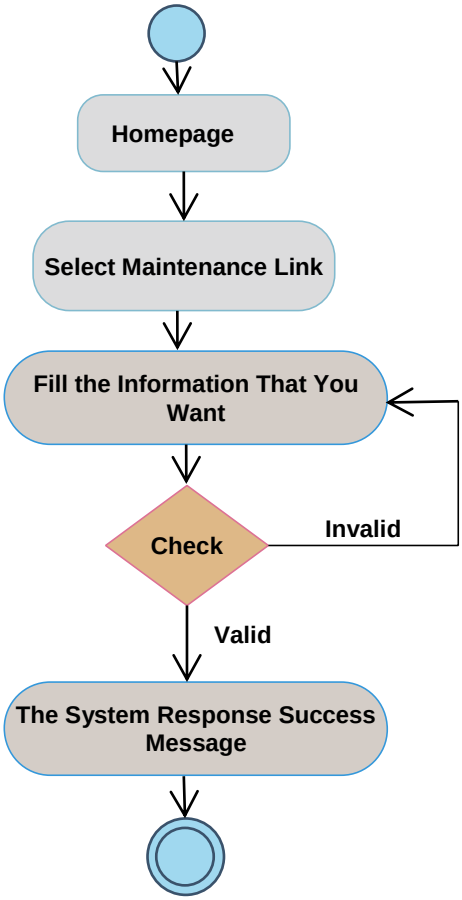


Figure 2. 8 activity diagram for order maintenance

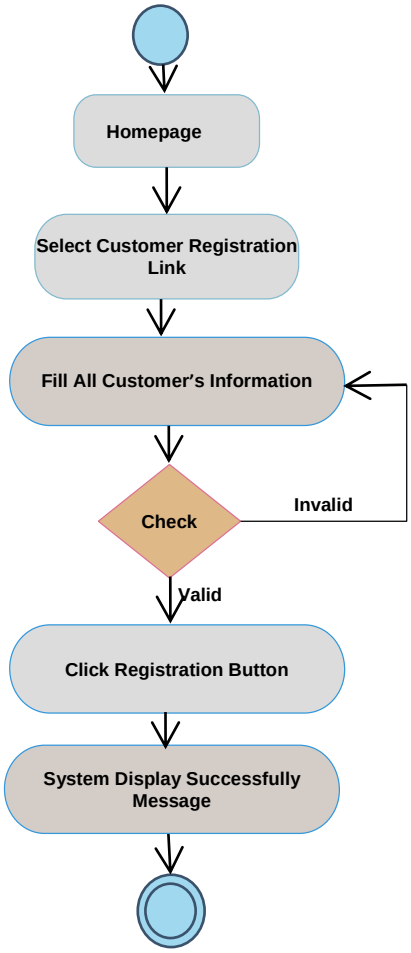


Figure 2. 9 activity diagram for generate report

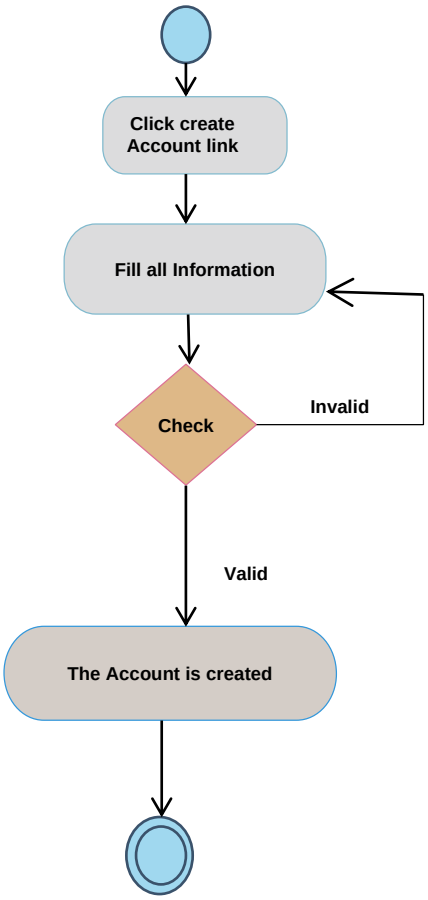


Figure 2. 10 activity diagram for create account

2.3.4. Analysis of class diagram

It represents the static view of an application. Class diagram is not only used for visualizing, describing and documenting different aspects of a system but also for constructing executable code of the software application. The class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The **class diagram shows a collection of classes, interfaces, associations, collaborations and constraints.**

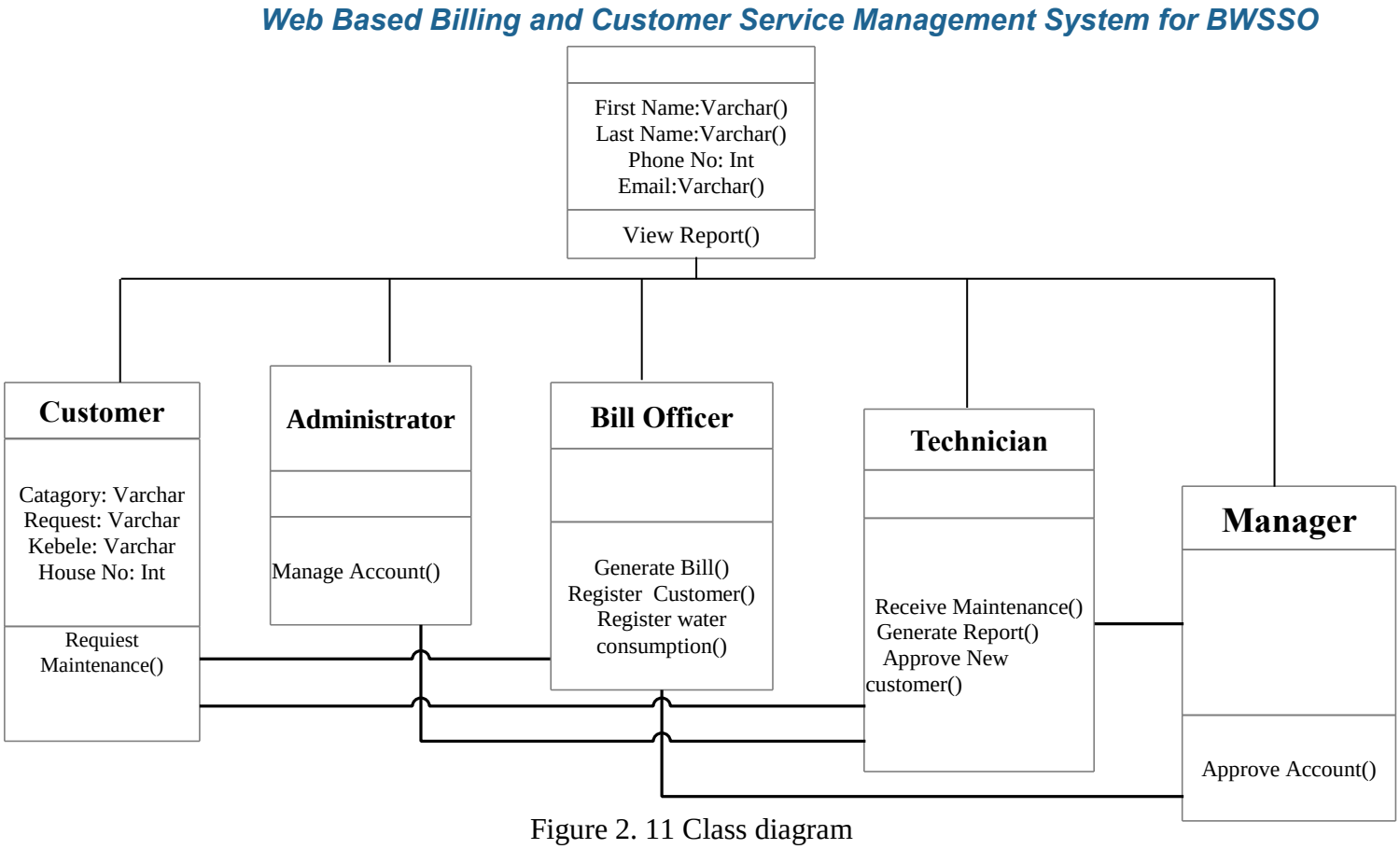


Figure 2. 11 Class diagram

CHAPTER THREE

3. SYSTEM DESIGN

INTRODUCTION

System design is the transformation of the analysis model into a system design model. In the analysis phase the team describes the system completely from the actor point of view and serves as the basis of communication between the client and the developers. But the analysis does not contain information about the internal structure of the system and its hardware configuration. In general how the system should be realized, so in system design phase the team describes the proposed system architecture, current software architecture and design goals.

3.1. Design class diagram

Class modeling used to describe the structure of this system. Class diagram provide an over view of target system by describing the object and classes inside the system and the relationship between them. Diagram describes our system in terms of objects, attributes, operations and relationships.

It provides a wide variety of usages; from modeling the domain specific data structure to detailed design of the target system .Class

Purpose:

A UML class diagram is not only used to describe the object and information structures in an application, but also show the communication with its users.

In a UML class diagram, classes represent an abstraction of entities with common characteristics.

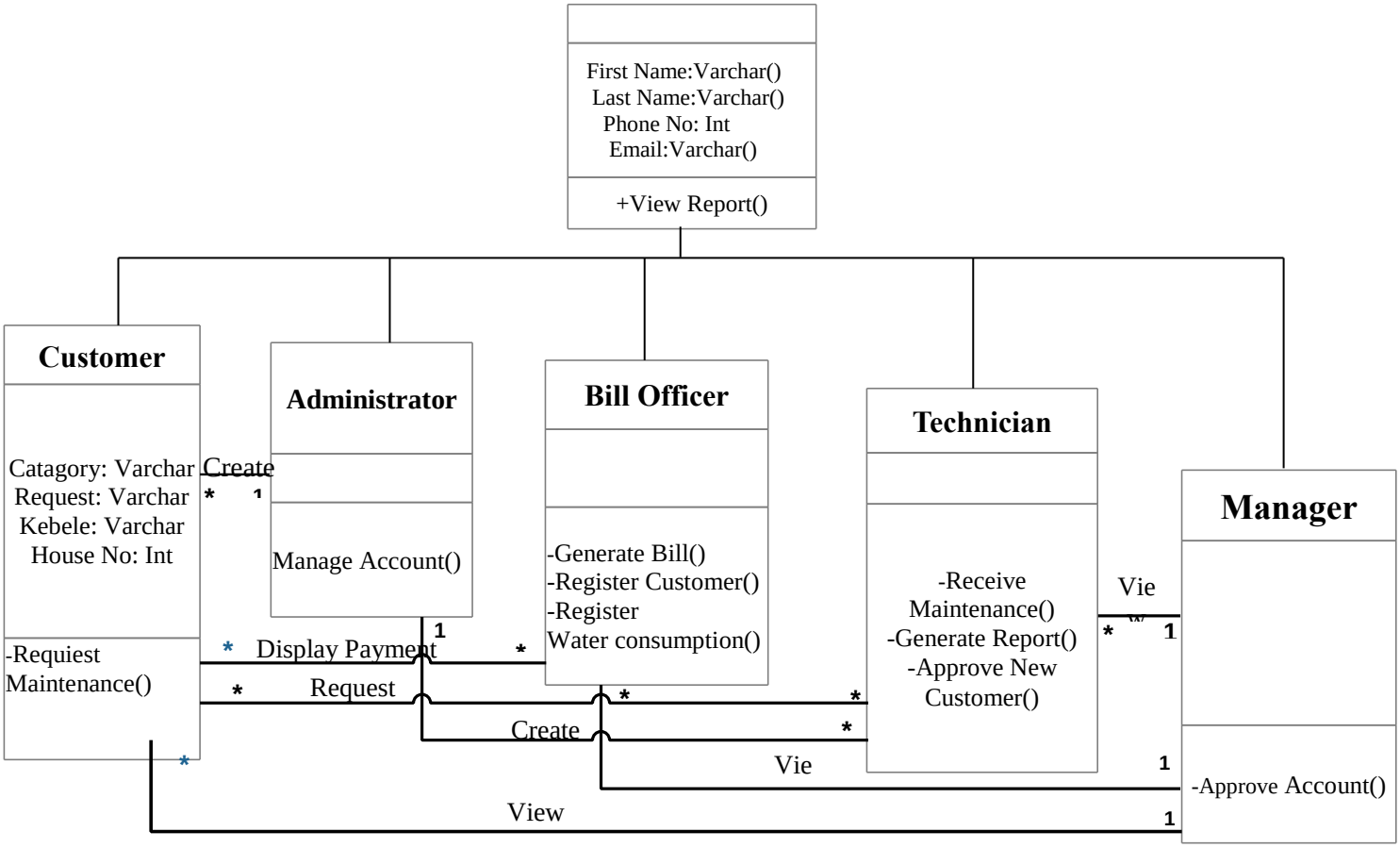


Figure 3. 1 Class diagram

Design Class diagram descriptions

Class Name	Attribute/ Operation	Description
User	User Name	Represent name of user
	First Name	Represent first name of user
	Last Name	Represent second name of user
	Phone no.	Represent phone no of user
	View Report()	User View Report
	Email	Represent email of user

Table 3. 1 Class Description of User

Class Name	Attribute/ Operation	Description
Administrator	Manage account()	

Table 3□2 Class Description of Administrator

Class Name	Attribute/ Operation	Description
Manager	Aprove Account()	Aprove the account of Employe

Table 3. 2 Class Description of Manager

Class Name	Attribute/ Operation	Description
Customer	Request	Customer requests to get service
	kebela	Customer's address
	House no	Identify the house of customer
	Request Maintenance()	Customer requests to maintenance

Table 3. 3 Class Description of Customer

Class Name	Attribute/ Operation	Description
Bill officer	Generate Bill()	Generate bill for customer
	Register water consumption()	Register water consumption value
	Register customer()	Register new customer

Table 3. 4 Class Description of Bill officer

Class Name	Attribute/ Operation	Description
Technical	Receive maintenance()	Technical Receive maintenance for request of customer
	Approve new customer()	Approve customer
	Generate report()	Generate report to the organization

Table 3. 5 Class Description of Technical

3.2 User interface design

The proposed system user interface design is the process that focuses on how information is provided to and accepts from users. Thus the user interface design is a technique which is designed in our new system for defined the manner in which users and system exchange information easily. Interface provides a user to perform the activity the system easily and effectively .in these system users will communicate with the system through the following interface.

Home page: All users firstly can get the home page. This page consists of Home, customer registration, About Us, user, Contact Us page .The user can see this page and select their needs and continues to next session.

Figure 3. 2 User interface for login

Registration:-

This registration page help for bill officers to register the customer by fill the necessary or relevant information's in the given form

Figure 3. 3 User interface for customer registration

3.3 System Archtecture(Deployment Diagram)

Deployment diagram is used to show the hard ware of the system, the software that is installed in the hard ware and also the middleware that used to connect the machines to one and another. It also shows how the software and the hard ware component work together.

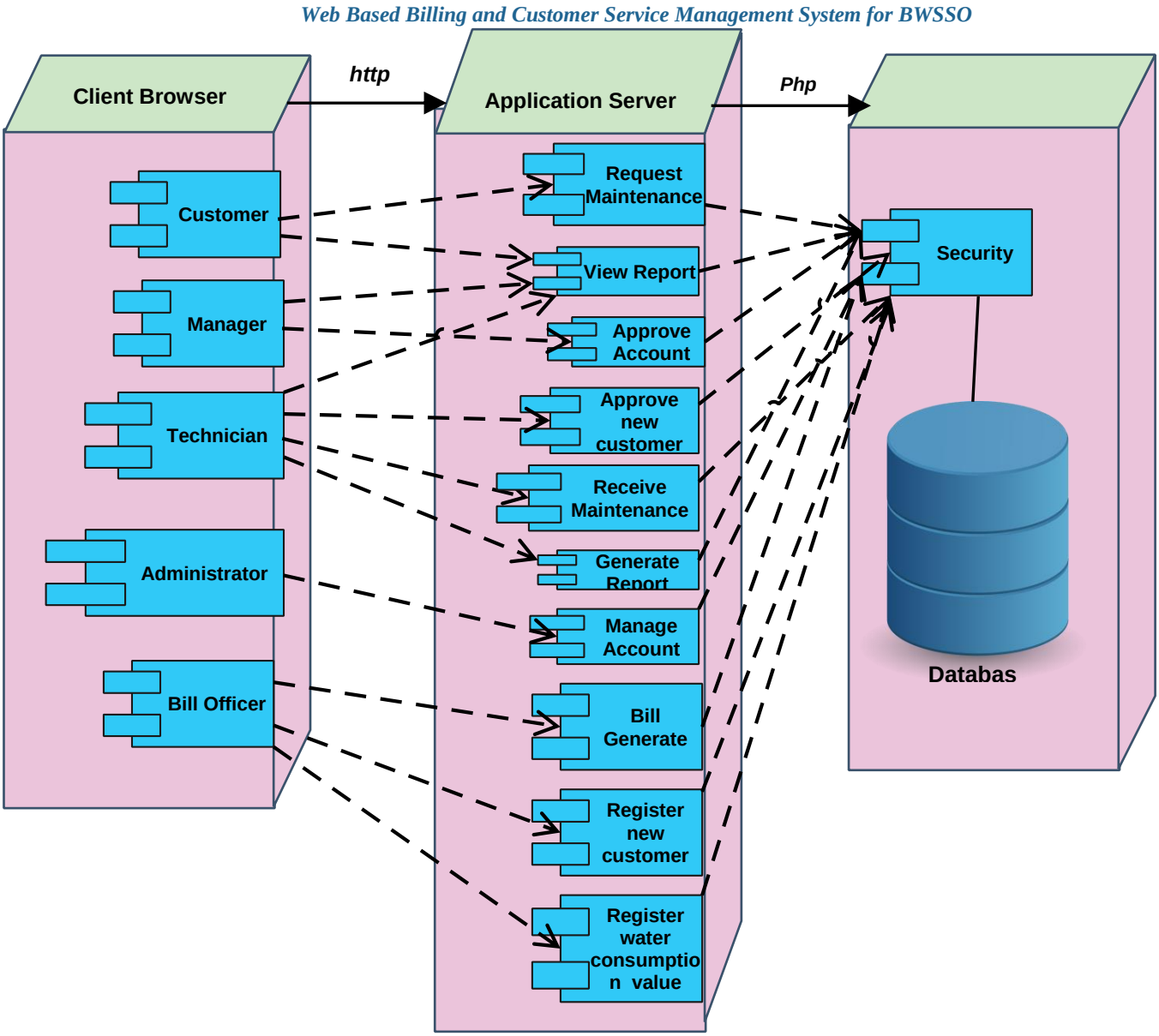


Figure 3. 4 Deployment diagram

CHAPTER FOUR

4. IMPLEMENTATIONS

4.1 Implementation Overview

In this chapter the team focuses on the implementation part, implementation concerned with the type of material (Hardware and Software required), objectives of implementation and code samples of the system.

4.1.1 Overview of programing languages used

Our system is implemented in PHP and MYSQL in a server-side or back-end programming languages. And HTML, CSS and JavaScript in client-side or front-end

We use HTML and CSS for prepare user interface.

- We use java script for form validation
- We use MYSQL database for storage of data. Because

4.1.2 Objectives of implementation

The objective of systems implementation phase is to convert the final physical system specifications into working and reliable system, document the work that has been done, and provide help for current and future users of the system.

4.2 Algorithm used

Algorithm for registration of new customer

Function registration(firstname,lastname,customerid,kebele,housenumber,username,password**).**

Function validates Form (Form Name);

Algorithm for checking whether the **field is emptyor not**

function Empty(elem, helperMsg)

{

if (elem.value.length == 0)

{

Display error message “Please insert only letters for your first name”;

}

Return true

};

Algorithm for checking whether the **field takes only numbers or not**

{

If function isNumericExpression = /^[0-9]+$/;

{

return true;

}

else

{

Display error message “please insert valid house number”;

}

end of the function validate Form

Algorithm for checking whether the **field takes only Characters or not**

{

iffunction isAlphabetalphaExpression = /^[a-z or A-Z]+$/;

{

return true;

}

else

{

Display error message “please insert only letters”;

};

end of the function validateForm.

Algorithm for Authentication (login)

Function Authentication (username and password)

If (variables are valid)

Entered password=retrieved password and

Entered username=retrieved username

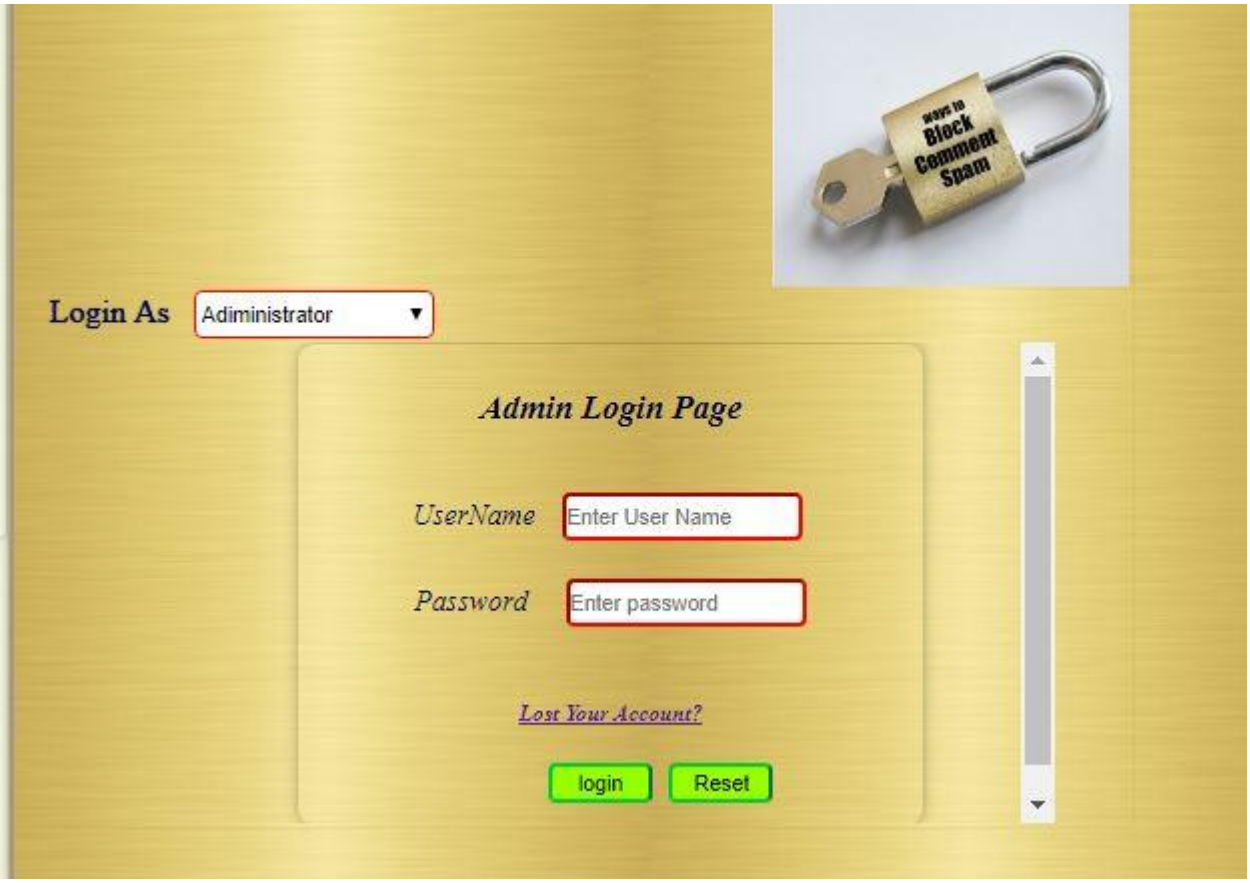
Open new page

Else

Display error “enter the correct username and password”

Return false.

end of the function authentication



4.3 Sample code

The admin login sample code

```
<html>

<head>

<script type='text/javascript'>

function formValidator() { // Make quick references to our fields

var user_name = document.getElementById('user_name');

var password = document.getElementById('password');

if (lengthRestriction(user_name, 4, 30))

  {

if (lengthRestrictionforpassword(password, 4, 15))

  {

return true;

  }

      }
```

```
return false;
}
function notEmpty(elem, helperMsg)
{
if (elem.value.length == 0)
{
alert(helperMsg);
 elem.focus(); // set the focus to this input
 return false;
}
return true;
}
function lengthRestriction(elem, min, max)
 {
var uInput = elem.value;
if (uInput.length >= min && uInput.length <= max)
{
return true;
}
else
{
alert("Please Enter Valid Username!");
 elem.focus(); return false;
}
}
function lengthRestrictionforpassword(elem, min, max)
 {
var uInput = elem.value;
if (uInput.length >= min && uInput.length <= max)
{
return true;
}
else
{
alert("Please Enter Valid Password!");
 elem.focus(); return false;
}
}
</script>
<style type="text/css">

form  {
background: -webkit-gradient(linear, bottom, left 175px, from(#CCCCCC), to(#EEEEEE));
background: -moz-linear-gradient(bottom, #CCCCCC, #EEEEEE 175px);
margin:auto;
position:relative;
width:370px;
height:250px;
font-family: ;
```

```
font-size: 14px;
font-style: italic;
line-height: 24px;
font-weight: ;
color: #09C;
text-decoration: none;
-webkit-border-radius: 10px;
-moz-border-radius: 10px;
border-radius: 10px;
padding:10px;
border: 1px solid #99779;
border: inset 0px solid #333;
-webkit-box-shadow: 0px 0px 8px rgba(0, 0, 0, 0.3);
-moz-box-shadow: 0px 0px 8px rgba(0, 0, 0, 0.3);
box-shadow: 0px 0px 8px rgba(0, 0, 0, 0.3);
}
.style1 {
   color:#000033;
        font-weight: bold;
}
.style2 {color: #0000FF}
body {
        background-image: url(Image/main.jpg);
}
.a input[type="submit"]{
width:65px;
height:25px;
border:1;
border-radius:5px;
border-bottom-style:solid;
border-color:#00CC33;
-webkit-border-radius:5px;
background-color:#99FF00;
}
.a input[type="reset"]{
width:65px;
height:25px;
border:1;
border-radius:5px;
border-bottom-style:solid;
border-color:#00CC33;
-webkit-border-radius:5px;
background-color:#99FF00;
}
.a input[type="text"]{
width:150px;
height:30px;
border:1;
border-bottom-style:solid;
border-color:red;
border-radius:5px;
```

```
-webkit-border-radius:5px;

}

.a input[type="password"]{

width:150px;

height:30px;

border:1;

border-bottom-style:solid;

border-color:red;

border-radius:5px;

-webkit-border-radius:5px;

}

</style>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"></head>

<body ><p align="center">

<table width="463" border="0" class="a">

<tr>

<td width="64" height="221"> </td>

<td width="389">

<form action="form1/adminlogincode.php" onsubmit='return formValidator()' method='POST' target="_top" id='form1'>

<h2 align='center' class="style1">Administrator Login Page</h2>

<div align="center"><span class="style2"></span>

<font color="#000033" size="+1">UserName</font>    <input type="text" name="user_name" id="user_name" placeholder="Enter User Name" id="username">

<br>

<br>

<span class="style2"></span>

<font color="#000033" size="+1">Password</font>    <input type="password" name="password" id="password" placeholder="Enter password"><br /><br />

                   <input type='submit' name="submit" value="login" />  

<input type='reset' value='Reset' />

</div>

</form></td>

</tr>

</table>

</p>

</body></html>
```

CHAPTER FIVE

5. TESTING

5.1 Unit testing

First we will tests each unit at each system. Each modules of the system can be tested check the working of each classes, methods and attributes of the system. For immediately maintain at which the problem is occurred.

Test case 1 :-for authentication of user login

Unit test=authentication of login user		
Assumption =login to the appropriate page		
Test data=username, password(empty,valid,invalid)		
Steps to be executed	Test data	Expected result
Empty username, password	Username=" ",password=" "	Please enter valid username and password
Valid username, invalid password	Username=userPassword=12	Please enter valid password
Valid password invalid username	Password=1234Username=us	Please enter valid username
Valid username and password	Username=user password=1234	Open new page

Test Case2 - customer registration

Unit to Test = Registration of Users			
Assumption= to get appropriate service from the organization			
Test Data= first name, last name ,Keble, house number, User name, password (invalid , Valid ,empty)			
Steps to be Executed		Test Data	Expected Results
Empty first name and all others filled and Click register button		Any valid data for the other fields	Please enter only letters for your first name
Invalid last name and valid input for other filled and Click register button		last name=abaynesh	Please enter Valid letters for last name
Enter valid house number ,empty all other filled and Click register button		House number= 09 Any valid data for the other fields	Please Enter the field
Enter All fields with valid input	All fields with valid data	Successfully Registered	

5.2 Integration testing

After we test each unit of the proposed system we will perform an integration test to check whether the system meets all the functional requirements. When a number of components are complete; it will test to ensure that they integrate well with each other, the operating system, and other components.

5.3 System testing

System test insures that the entire integrated software system meets requirements. It tests a configuration to insure known and predictable results. System testing is based on process description and flows, emphasizing pre-driven process links and integration points. In system testing is not about checking the individual parts of design, but about checking the system as a whole. In effect it is one giant component. System testing insures the features of functional and non-functional requirements and the specifications.

5.4 Acceptance testing (alpha testing, beta testing)

Acceptance testing

This testing is done by the customer (on-behalf) to ensure that the delivered product meets the requirements and works as the customer expected. It includes:-

Alpha:- As the team we test our implemented code before releasing to the market by using two phases.

- ✓ First phase that we follow as the developer we debug by using our wampserver and different browser.
- ✓ In the second phase we follow the methods QA staff for additional testing.

Beta:- We conducted some users to ensure whether or not accept our implemented system.

CHAPTER SIX:

6. CONCLUSION AND RECOMMENDATION

6.1 Conclusion

We have developed web based customer management system to enable carefully store all necessary customer information effectively. The system also helps users for the proper functioning of their operation in the considerable time and accuracy. the developed system have also been aimed to improve customers registration process, inserting customers consumption, calculating monthly fee process, displaying customers payment, and customer maintenance request within efficient and effective way. the proposed system enables to register a new customers and their requesting order through internet connection without going to the organization. Finally the team expects that the developed system will change the general customer management system and make it more reliable and efficient than the previous manual system.

6.2 Recommendation

Based on shortage of time and some other condition like resources we do not include some features to our project. The team wants to recommend those who want further to work on our project to include the features like employee salary detail information. The other features were commend to be added the penalty of customer consumption fee beyond the deadline.

The team recommend strongly that the system should be available at any time. Users of this website should have a knowledge and skill of computer usage and Internet access. Be able to use more acceptable and available system to internal system users.

Future enhancements

For the future as the services of the organization becomes increased, this system should be improved by adding functions and using better technologies. So, future works that the project team proposed are:

- ❖ Maintaining the system according to the services of the organization after applying and testing the acceptance of the project by the organization
- ❖ Connecting the system with other water supply organizations.