



HOTEL MANAGEMENT SYSTEM

13MSE0266-SURYA PRAKASAM
13MSE0175-GAYATHRI

Project Report
Of
SWE503- Software Configuration Management PBL Course
Fall 2016-17

ABSTRACT:

This is the document of the Project proposal for developing a hotel management system. It consists of the current background of the hotel and problems having due to present system and how we are going to overcome those through our proposed system. After gathering requirements we have found that current hotels are either using a manual file based system for their processes or an online system without food booking system. They are keeping a huge amount of files to handle guest details, bills, inventory items. ..Etc. Search for a record in the file system is really harsh although they are not well secured there is a probability of getting damaged due to a fire or a tsunami or any other kind of a disaster. The project aimed to build a fully functional system in order to achieve the efficiency in the hotel management. In which can be used by customers to check the availability of rooms and reserve accommodation, order food and other facilities.

TABLE OF CONTENTS

1. INTRODUCTION

- 1.1 Existing and proposed system
- 1.2 Intuition to do this project

2. SYSTEM ARCHITECTURE

- 2.1 Modules
- 2.2 ER diagram
- 2.3 Use case diagram

3. SOFTWARE CONFIGURATION MANAGEMENT ACTIVITIES

- 3.1 Configuration identification
- 3.2 Configuration change control
- 3.3 Configuration status accounting
- 3.4 Auditing and review

4. SCM PLANNING

- 4.1 Introduction
- 4.4 Configuration baseline
- 4.5 Configuration management responsibility
- 4.6 Interface

5. SOFTWARE CONFIGURATION TOOL

- 5.1 TortoiseSVN -Introduction
- 5.2 Features
- 5.3 Repository Creation
- 5.4 Importing Hotel management and Checking out
- 5.4 Making changes
- 5.5 Sub Version management

6. TEST CASES AND RESULTS

6.1 Test case 1

6. CONCLUSION AND FUTURE ENHANCEMENT

7. APPENDICES

7.1 Source code

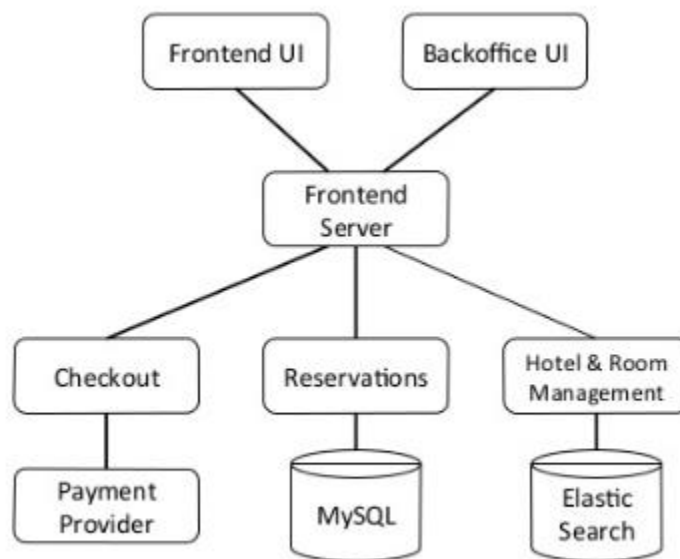
8. REFERENCES

1. INTRODUCTION

1.1 EXISTING AND PROPOSED SYSTEM

In the existing system, the person who wants to book a room has to visit the hotel for booking hotel rooms, and enquiry. The existing system is manual system. The hotel management has to keep records of rooms manually. To overcome the problems of manual system, online hotel management system is proposed. Although an online system exists, booking of food service doesn't exist. The Central objective of our Hotel Management system is to provide online facility for booking rooms and order foods.

2. SYSTEM ARCHITECTURE



2.1 MODULES

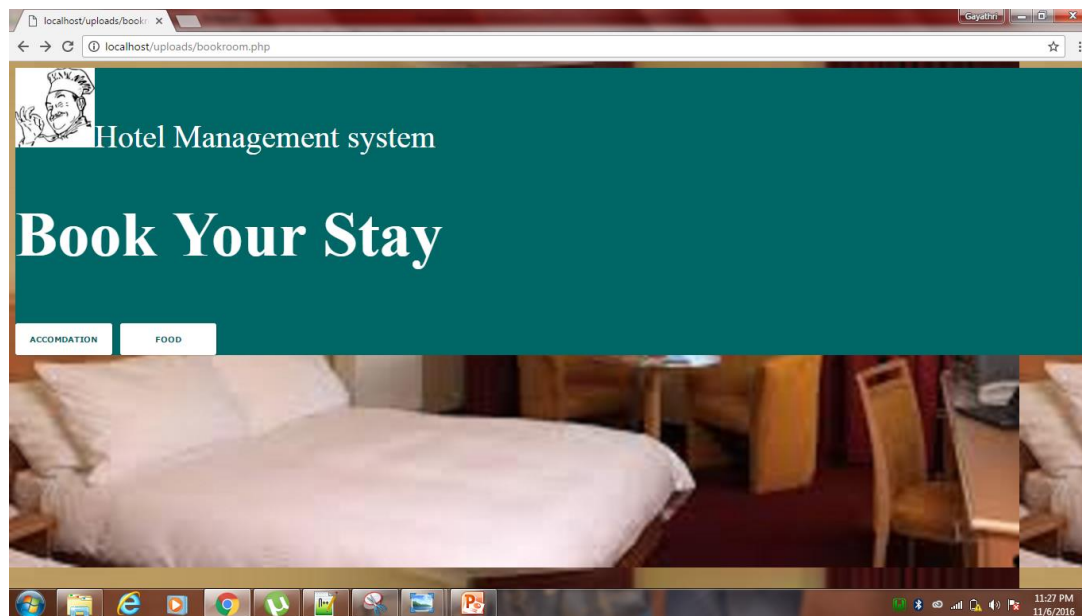


Figure 2-1: Home page

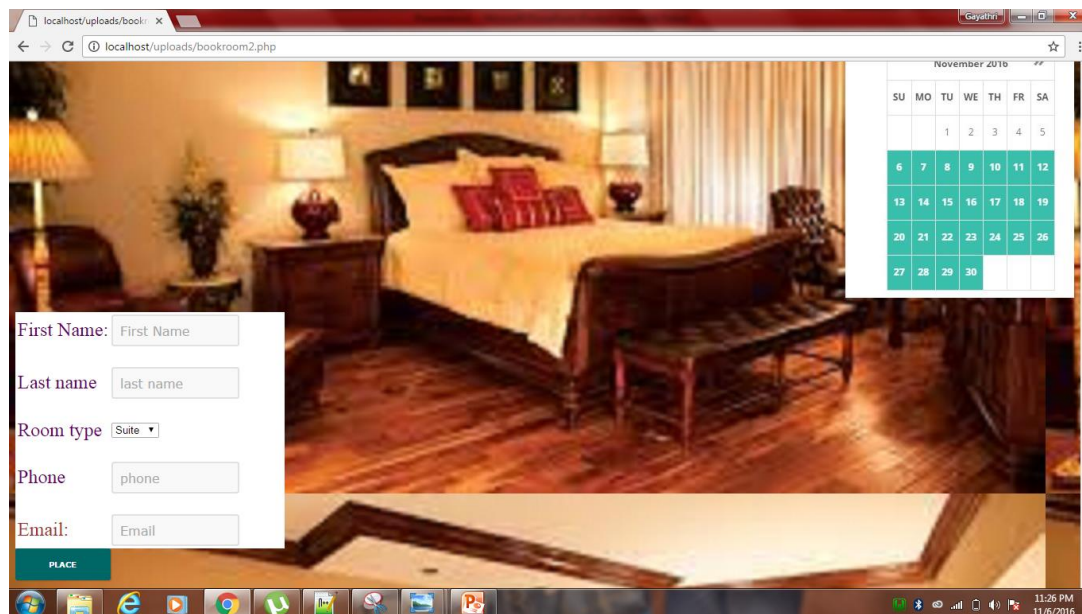


Figure 2-2: Checking room availability and booking

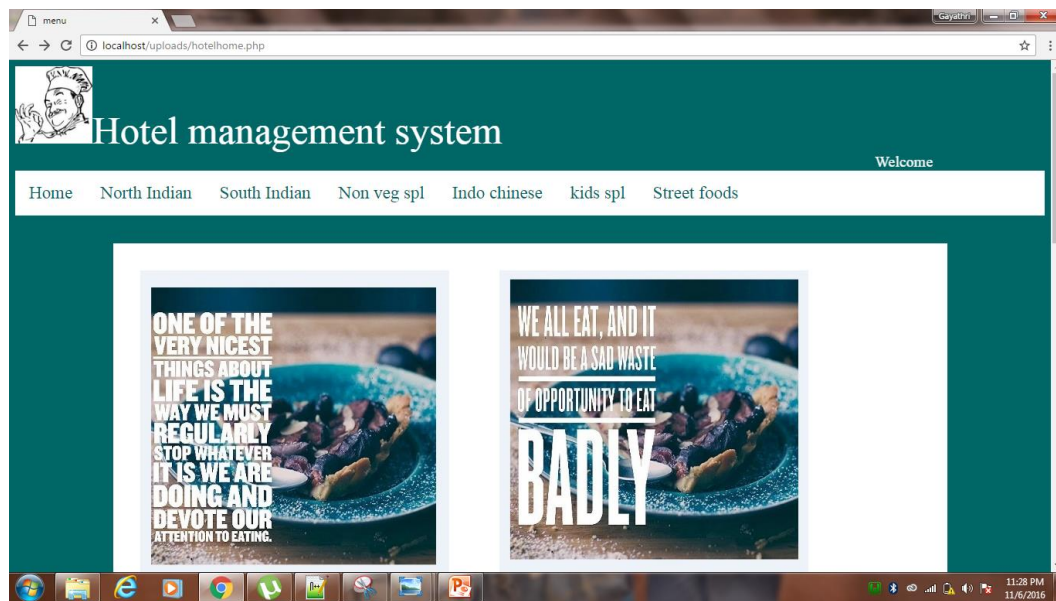


Figure 2-3: Ordering food home page

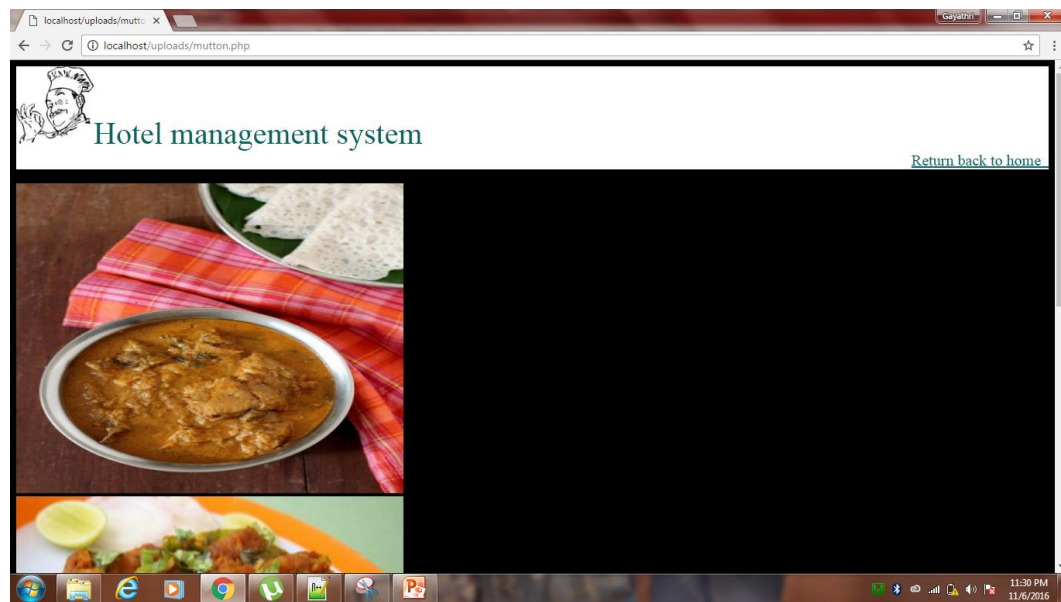


Figure 2-4: Budget Officer Registration

localhost/uploads/bookfood.php

Order your foods

[Return back to home](#)

Name:

Date and time:

menu list:

Email:

Figure 2-5: Food ordering Form

Untitled Document

localhost/feedback.php

Hotel mangement system

Username:

Review Star:

Comments:

Comments please:

Email:

© 2015 - SIMPLE THEME

Figure 2-6: Budget Officer home page

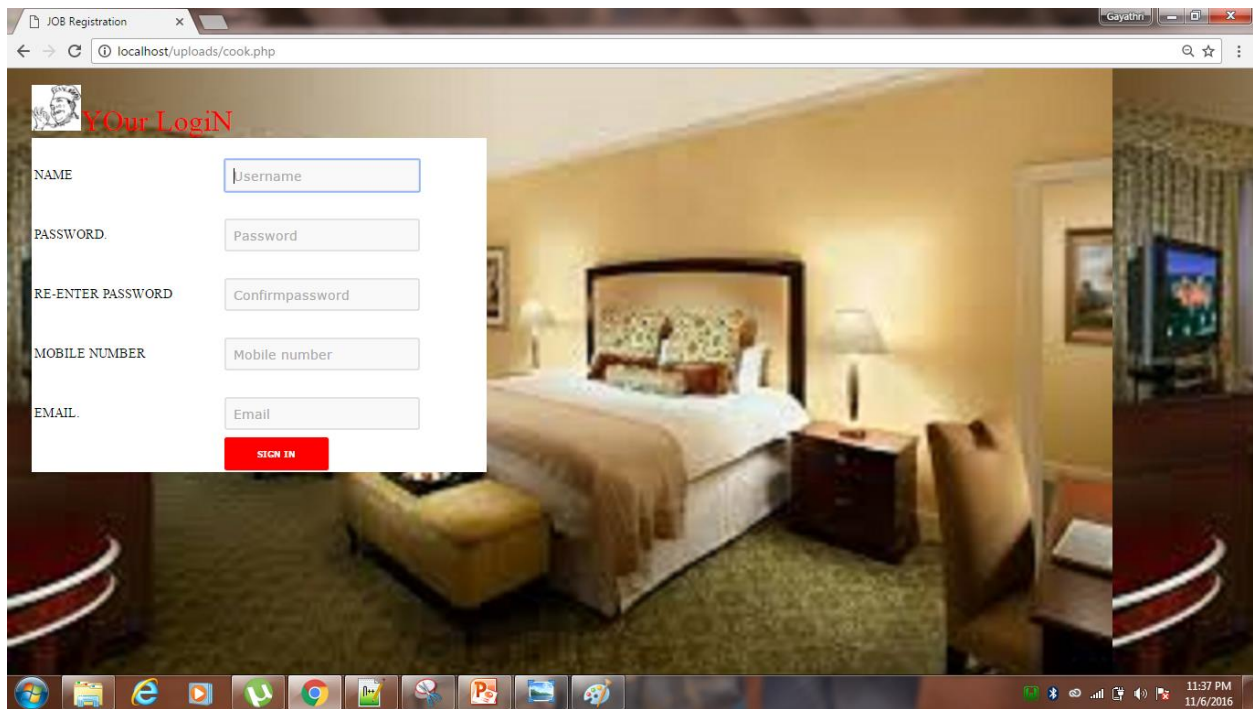
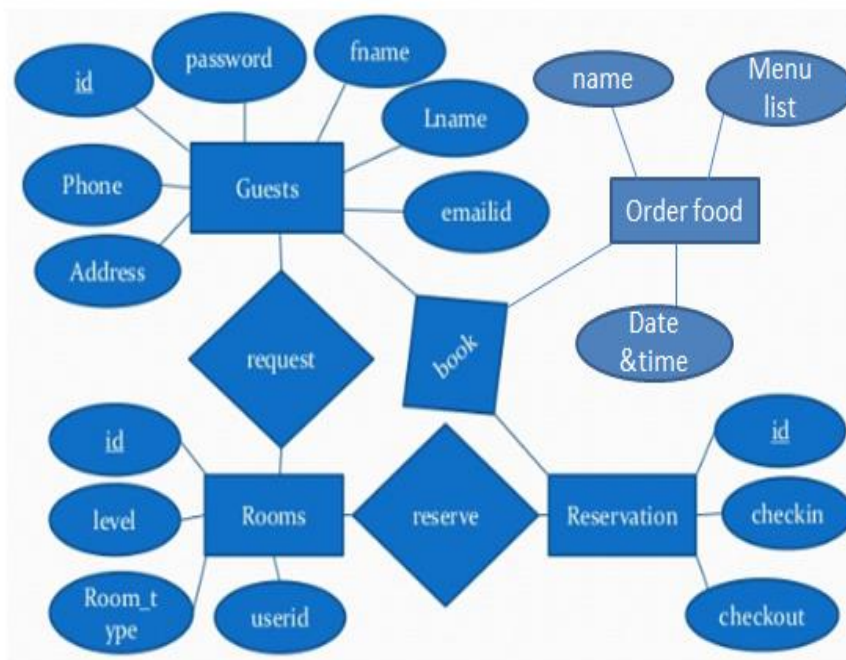


Figure 2-7: Distribution

2.2 ER DIAGRAM



USECASE DIAGRAM

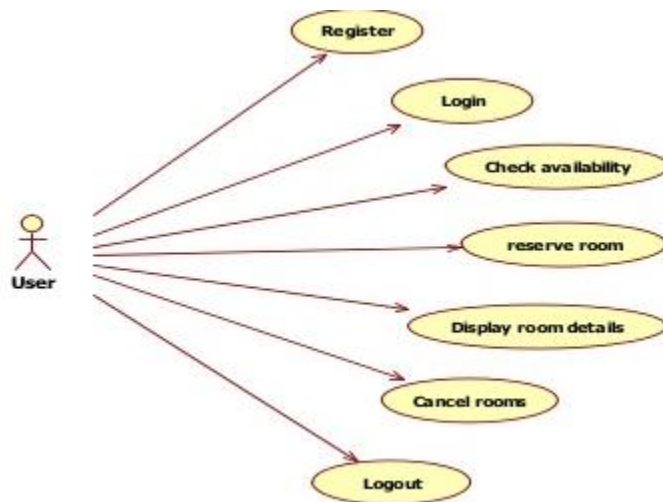


Figure 2-10: Department Officer

3. SOFTWARE CONFOGURATION MANAGEMENT ACTIVITIES

3.1 CONFIGURATION IDENTIFICATION

- Login (login as (using their job),username, password)
- Sign up page. (Name, email, password confirm password, address).
- Booking and ordering form contains:
 - i. First name
 - ii. Last name
 - iii. Username
 - iv. Password
 - v. Confirm password.
 - vi. Address
 - vii. Date and time
 - viii. Mobile number.
 - ix. City and state.
 - x. Email
 - xi. Room type
 - xii. Total amount
 - xiii. Database creation.

3.2 CONFIGURATION CHANGE CONTROL

- Puts you in a better position to be ready for change.
- Understanding that different characteristics of change need different approaches.
- Helps you implement a process to navigate the required change.
- Builds commitment to work together, to bring people with you.
- Can help with appreciating what is currently done well, so that the future is built on strengths which currently exist within the organization.
- Delivers change those results in real improvements.
- Compatibility.

3.3 CONFIGURATION STATUS ACCOUNTING

- Control changes to source code and documentation.
- Record the individual making the change.
- Record the date and reason for making the change.
- Provide version description to the unit level.
- Automatically assign new version numbers upon successful compilation of files.

3.4 AUDITING AND REVIEW

It generates the report whatever we are done in the previous activities.

4. SOFTWARE CONFIGURATION MANAGEMENT PLANNING

4.1 CHANGE CONTROL:

The activities for control of the product after formal approval of its **product configuration information**.

4.2 CONFIGURATION:

The interrelated functional and physical characteristics of a product defined in product configuration information.

4.3 CONFIGURATION BASELINE:

The approved product configuration information that establishes the characteristics of a product at a point in time that serves as reference for activities throughout the life cycle of the product.

4.4 CONFIGURATION MANAGEMENT RESPONSIBILITY:

- The organization should identify and describe responsibilities and authorities related to the implementation and verification of the configuration management process. The following should be considered:
- The complexity and nature of the product;
- The needs of the different product life cycle stages;
- The interfaces between activities directly involved in the configuration management process;
- The other relevant interested parties that may be involved, within and outside the organization.
- The identification of the responsible authority for verifying implementation activities.
- The identification of the dispositioning authority.

4.6 INTERFACE:

The definition of the interfaces is one of the most **important planning elements** for ensuring a smoothing operation. Every possible effort should be made to reach the common agreement regarding each organization's responsibility regarding the interface(s) and then document them in the subsection.

The basic types of interface should consider here are organization, phase, software and hardware.

ORGANIZATION: hotel management to people

PHASE: login, sign up, book accommodation and order food.

TOOLS USED: This project was build using the following applications:

- Notepad ++
- Wamp server
- Wamp MySql

- Linux /Windows OS
- TortoiseSVN.

TECHNOLOGIES USED:

- HTML.(Hypertext Markup Language)
- CSS(Cascading Style Sheet)
- Java script.
- PHP.(Hypertext Preprocessor)
- MySql

5. TOOL-TORTOISESVN

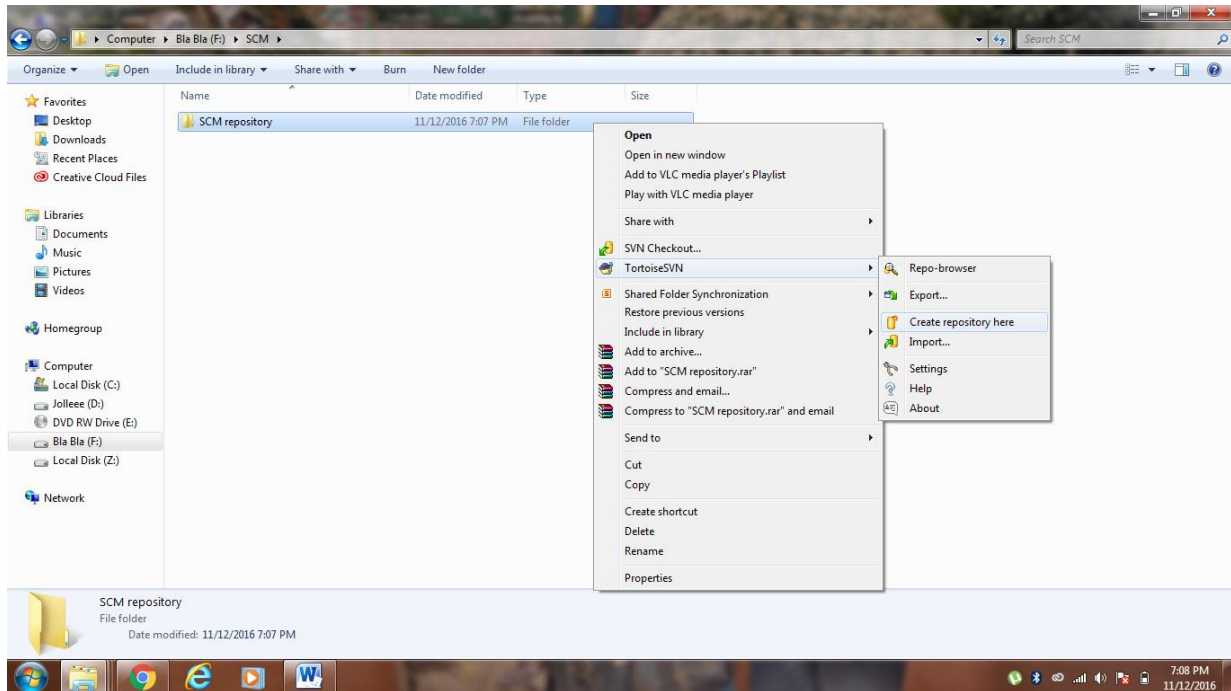
5.1 INTRODUCTION

TortoiseSVN is free software for software developers (programmers). It helps programmers manage different versions of the source code for their programs. Tortoise SVN is a Subversion client, implemented as a Microsoft Windows shell extension. It is released under the GNU General Public License

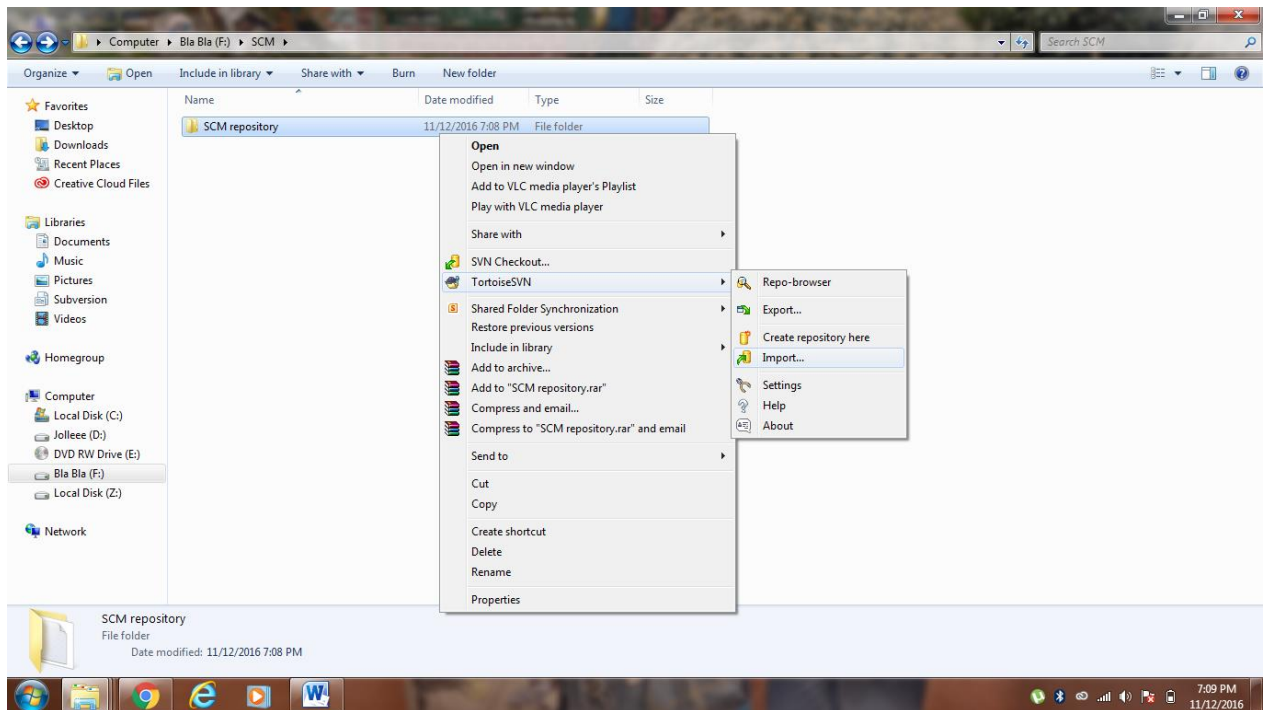
5.2 FEATURES

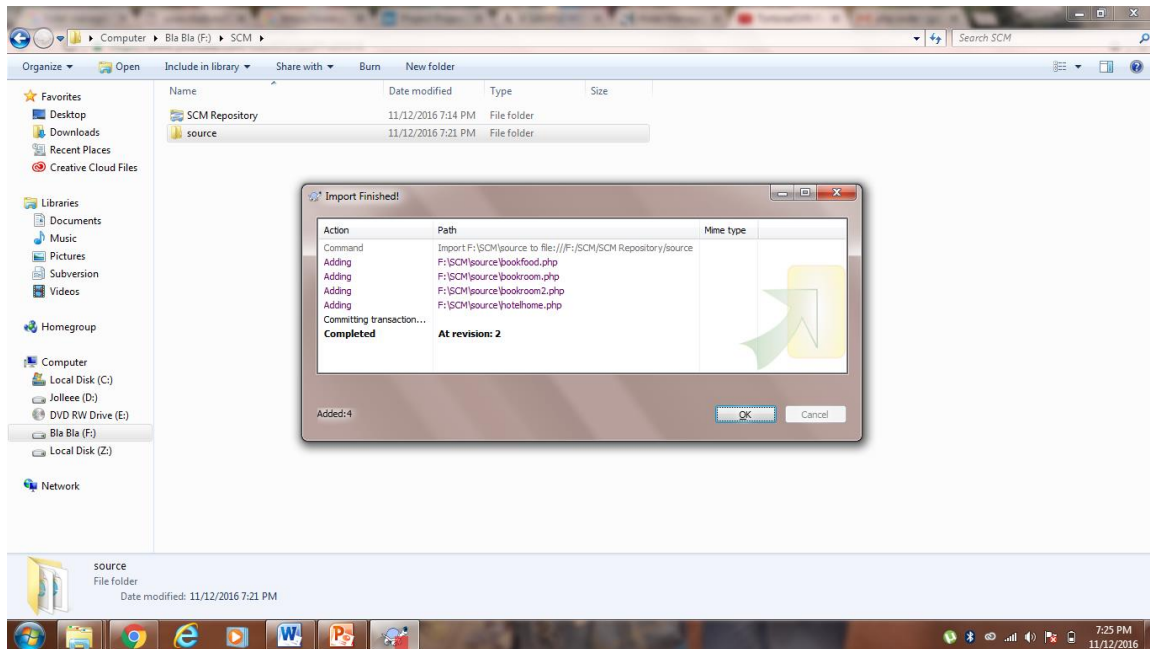
- Easy to use
- All Subversion protocols are supported
- Powerful commit dialog
- Integration with issue tracking systems
- Available in many languages
- TortoiseSVN is stable

5.3 CREATING REPOSITORY

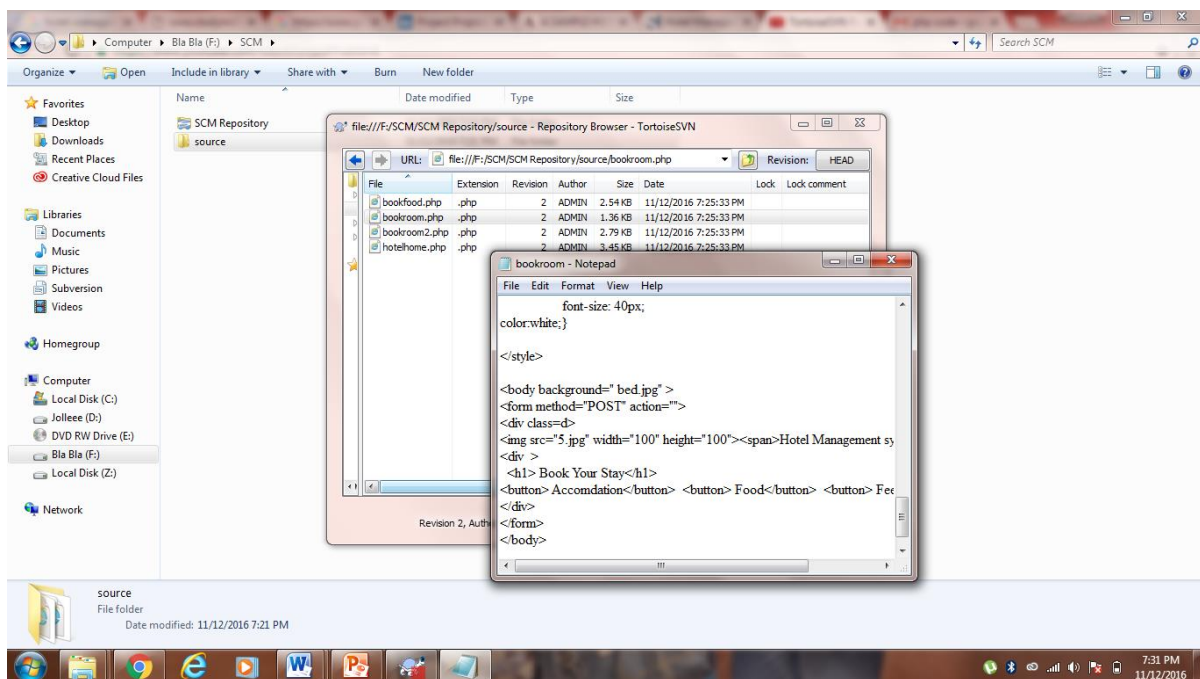


5.4 IMPORTING FILES

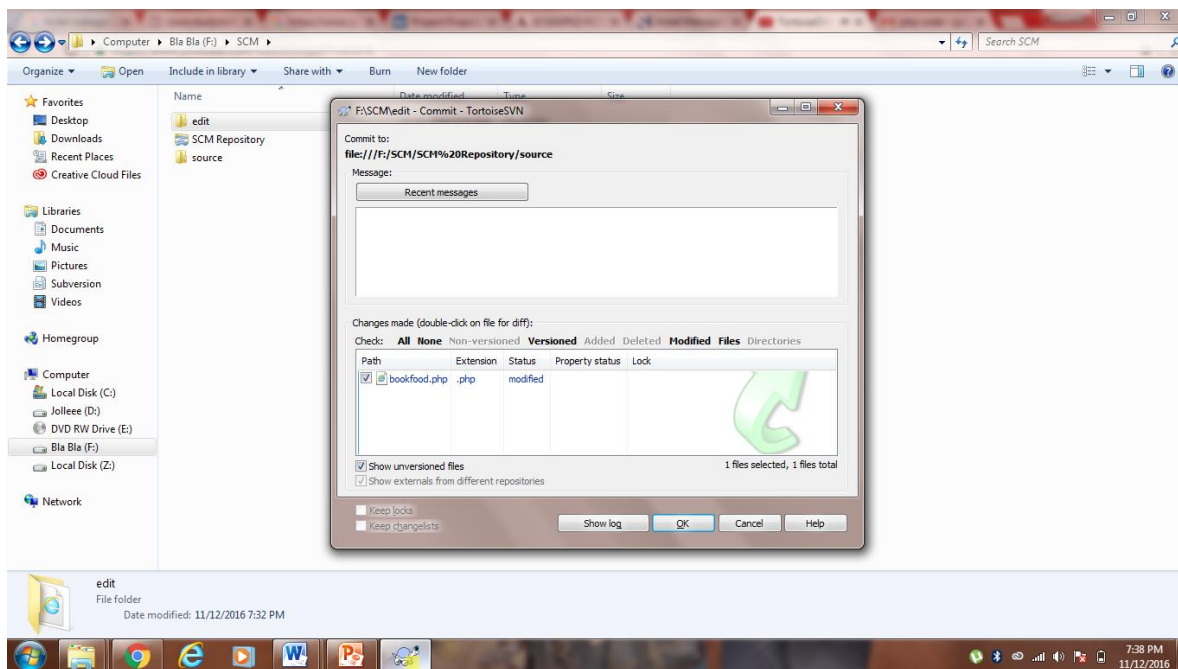
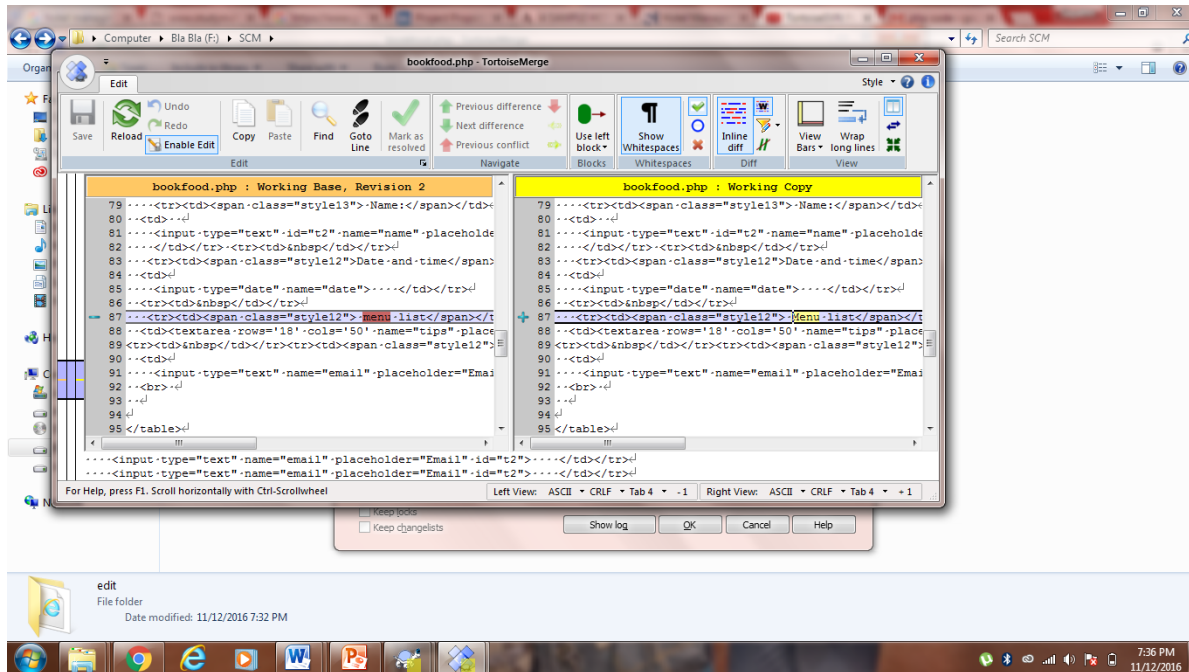




5.5 MAKING CHANGES

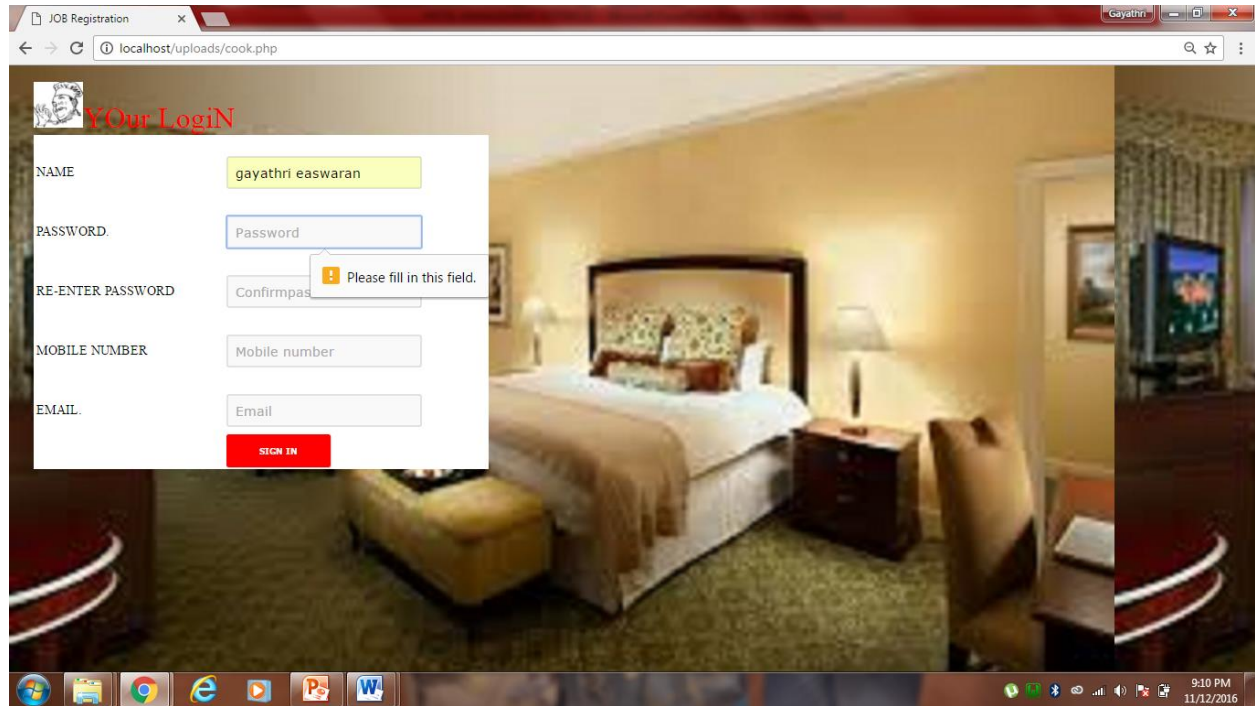


5.6 SUB VERSION



5. TEST CASES AND RESULTS

5.1 TEST CASE 1



6. CONCLUSION AND FUTURE ENHANCEMENT

The proposed system saves time of customers in quickly reserving rooms. The ability to book rooms anytime, from anywhere with Internet access. It provides the information about hotel facilities. Very secure and User friendly

For future development, we would enhance our project to overcome bugs and more user friendly.

7. APPENDICES

7.1 SOURCE CODE

7.1.1 home

```
<html>
<head>
<style type="text/css">
.style {
    Background-color:white ;
    font-size: 25px;
}
a {color: #006666;
font-size:19;}
p{color: white;
font-size:25;
}
.style11 {
    font-size: 30px;
    font-style: italic;
}
.style12 {
    color: #993333;
    font-size: 24px;
}
.style13 {
    color: #660066;
    font-size: 24px;
}
input
{
    width:75%;
    height:40px;
    border:solid #e1e1e1 1px;
    border-radius:3px;
```

```

padding-left:10px;
font-family:Verdana, Geneva, sans-serif;
font-size:16px;
background:#f9f9f9;
transition-duration:0.5s;
box-shadow: inset 0px 0px 1px rgba(0,0,0,0.4);
}
button
{
width:9%;
height:40px;
border:0px;
background:white;
background:-moz-linear-gradient(top, #595959 , #515151);
border-radius:3px;
box-shadow: 1px 1px 1px rgba(1,0,0,0.2);
color:#006666;
font-family:Verdana, Geneva, sans-serif;
font-size:10px;
font-weight:bolder;
text-transform:uppercase;
}
.style14 {
color:white;
font-size: 20px;
}
.d{
Background-color:#006666 ;
font-size: 40px;
color:white;}

</style>

<body background=" bed.jpg" >
<form method="POST" action="">
<div class=d>
<span>Hotel Management system </span>
<div >
<h1> Book Your Stay</h1>
<button> Accomdation</button> <button> Food</button> <button> Feedback</button>
</div>
</form>
</body>
</html>
7.1.2 Room.html

<html>
<head>

```

```
<style type="text/css">
```

```
.style {  
    Background-color:white ;  
    font-size: 25px;  
}  
a {color: #006666;  
font-size:19;}  
p{color: white;  
font-size:25;  
}  
.style11 {  
    font-size: 30px;  
    font-style: italic;  
}  
.style12 {  
    color: #993333;  
    font-size: 24px;  
}  
.style13 {  
    color: #660066;  
    font-size: 24px;  
}  
input  
{  
    width:75%;  
    height:40px;  
    border:solid #e1e1e1 1px;  
    border-radius:3px;  
    padding-left:10px;  
    font-family:Verdana, Geneva, sans-serif;  
    font-size:16px;  
    background:#f9f9f9;  
    transition-duration:0.5s;  
    box-shadow: inset 0px 0px 1px rgba(0,0,0,0.4);  
}  
button  
{  
    width:9%;  
    height:40px;  
    border:0px;  
    background:#006666;  
    background:-moz-linear-gradient(top, #595959 , #515151);  
    border-radius:3px;  
    box-shadow: 1px 1px 1px rgba(1,0,0,0.2);
```

```

        color:white;
        font-family:Verdana, Geneva, sans-serif;
        font-size:10px;
        font-weight:bolder;
        text-transform:uppercase;
    }
    .style14 {
        color:white;
        font-size: 20px;
    }
    .d{
        Background-color:#006666 ;
        font-size: 40px;
        color:white;}

</style>

<body background="bed1.jpg" >
<form method="POST" action="">
<div class=d>
<span>Hotel Management system</span>
<div align="right">
<a href="hotelhome.php" class="style14"target="_self" >Return back to
home&nbsp;&nbsp;&nbsp;</a><br></div><br></div><br>
<div >
<div align="right">
</div>
<table bgcolor="white">
<tr><td><span class="style13"> First Name:</span></td>
<td>
<input type="text" id="t2" name="fname" placeholder="First Name" value="">
</td></tr> <tr><td>&nbsp;</td></tr>
<tr><td><span class="style13">Last name</span></td>
<td>
<input type="text" placeholder="last name" name="date"> </td></tr>
<tr><td>&nbsp;</td></tr>
<tr><td><span class="style13">Room type</span></td>
<td>
<select> <option>Suite</option>
<option>VIP</option>
<option>Single</option> </select></td></tr>
<tr><td>&nbsp;</td></tr>
<tr><td><span class="style13"> Phone</span></td>
<td><input type="text" name="phone" placeholder="phone" required ></td></tr>
<tr><td>&nbsp;</td></tr><tr><td><span class="style12"> Email:</span></td>
<td>
<input type="text" name="email" placeholder="Email" id="t2"> </td></tr>
<br>

```

```

</table>
</div>
<button name="submit" type="submit">Place</button>
<br>
</form></body>
</html>
<?php
if(isset($_POST['submit']))
{

$un=$_POST['name'];
$p=$_POST['tipname'];

$cp=$_POST['tips'];
$em=$_POST['email'];

$dbname="tip";
$usr="root";
$l="localhost";
$pass="";
$conn = mysqli_connect($l,$usr,$pass,$dbname);

$query="INSERT INTO vtip(username,tipname,tip,email)VALUES('$un','$p','$cp','$em')";
if(mysqli_query($conn,$query))
{
header('Location: drop.html');
}

mysqli_close($conn);
}
?>

```

6.1.3 Food.html

```

<html>
<head>

<style type="text/css">

.style {
    Background-color:white ;
    font-size: 25px;
}
a {color: #006666;

```

```
font-size:19;}
p{color: white;
font-size:25;
}
.style11 {
    font-size: 30px;
    font-style: italic;
}
.style12 {
    color: #993333;
    font-size: 24px;
}
.style13 {
    color: #660066;
    font-size: 24px;
}
input
{
    width:75%;
    height:40px;
    border:solid #e1e1e1 1px;
    border-radius:3px;
    padding-left:10px;
    font-family:Verdana, Geneva, sans-serif;
    font-size:16px;
    background:#f9f9f9;
    transition-duration:0.5s;
    box-shadow: inset 0px 0px 1px rgba(0,0,0,0.4);
}
button
{
    width:9%;
    height:40px;
    border:0px;
    background:#006666;
    background:-moz-linear-gradient(top, #595959 , #515151);
    border-radius:3px;
    box-shadow: 1px 1px 1px rgba(1,0,0,0.2);
    color:white;
    font-family:Verdana, Geneva, sans-serif;
    font-size:10px;
    font-weight:bolder;
    text-transform:uppercase;
}
.style14 {
    color:white;
    font-size: 20px;
}
.d{
```

```
$em=$_POST['email'];
```



```
$dbname="tip";
$usr="root";
$l="localhost";
$pass="";
$conn = mysqli_connect($l,$usr,$pass,$dbname);

$query="INSERT INTO vtip(username,tipname,tip,email)VALUES('$un','$p','$cp','$em')";
if(mysqli_query($conn,$query))
{
header('Location: drop.html');
}

mysqli_close($conn);
}
?>
```

8. REFERENCES

1. <http://www.w3schools.com/php/>
2. <http://www.tutorialspoint.com/php/>
3. <http://www.w3.org/Style/Examples/011/firstcss.en.html>
4. <http://www.slideshare.net/Jaya0006/ppt-for-online-hotel-management5>.