

# **SECURE FILE MANAGEMENT SOFTWARE**

## **MAIN PROJECT REPORT**

*Submitted by*

**MARIYA ANTO**

**MARY ANTONY C**

**MERIN JOSEPH**

**ROSE MOHAN**

**SUMI JOSE**

*in partial fulfillment for the award of the degree  
of*

**BACHELOR OF TECHNOLOGY (B.TECH)**

in

**COMPUTER SCIENCE & ENGINEERING**

of

**UNIVERSITY OF CALICUT**

Under the guidance of

**Ms. Sujana K A**



JUNE & 2011

**Department of Computer Science & Engineering**  
**JYOTHI ENGINEERING COLLEGE, CHERUTHURUTHY**  
THRISSUR 679 531

# **SECURE FILE MANAGEMENT SOFTWARE**

## **MAIN PROJECT REPORT**

*Submitted by*

**MARIYA ANTO**

**MARY ANTONY C**

**MERIN JOSEPH**

**ROSE MOHAN**

**SUMI JOSE**

*in partial fulfillment for the award of the degree  
of*

**BACHELOR OF TECHNOLOGY (B.TECH)**

in

**COMPUTER SCIENCE & ENGINEERING**

of

**UNIVERSITY OF CALICUT**

Under the guidance of

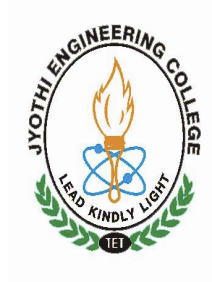
**Ms. Sujana K A**



JUNE & 2011

**Department of Computer Science & Engineering**  
**JYOTHI ENGINEERING COLLEGE, CHERUTHURUTHY**  
THRISSUR 679 531

**Department of Computer Science & Engineering**  
**JYOTHI ENGINEERING COLLEGE, CHERUTHURUTHY**  
**THRISSUR 679 531**



JUNE & 2011

**BONAFIDE CERTIFICATE**

Certified that this project report “ **SECURE FILE MANAGEMENT SOFTWARE** ” being submitted in partial fulfillment of the requirements for the award of degree of **Bachelor of Technology** of **University of Calicut** is the bonafide work of “ **MARIYA ANTO, MARY ANTONY C, MERIN JOSEPH , ROSE MOHAN, SUMI JOSE** ”, who carried out the project work under our supervision.

Prof. Muralee Krishnan C  
**HOD**  
Dept. of CSE

Ms. Sujana K A  
**PROJECT GUIDE**  
Asst. Professor  
Dept. of CSE

# CONTENTS

|  |            |
|--|------------|
| <b>Acknowledgement</b>                           | <b>iii</b> |
| <b>Abstract</b>                                  | <b>iv</b>  |
| <b>List of Figures</b>                           | <b>v</b>   |
| <b>List of Tables</b>                            | <b>vi</b>  |
| <b>List of Abbreviations</b>                     | <b>vii</b> |
| <b>1 Introduction</b>                            | <b>1</b>   |
| 1.1 Overview . . . . .                           | 1          |
| 1.1.1 Administrator Module . . . . .             | 2          |
| 1.1.2 Outsource Manager Module . . . . .         | 2          |
| 1.1.3 User Module . . . . .                      | 2          |
| 1.2 Motivation and Technical Relevance . . . . . | 3          |
| 1.3 Progress of project . . . . .                | 4          |
| 1.4 Member roles and responsibilities . . . . .  | 5          |
| 1.5 Layout . . . . .                             | 6          |
| <b>2 Literature Survey</b>                       | <b>7</b>   |
| 2.1 Documentation . . . . .                      | 7          |
| 2.1.1 Papers & websites . . . . .                | 7          |
| <b>3 Proposed System</b>                         | <b>9</b>   |
| 3.1 Process Model . . . . .                      | 9          |
| <b>4 System Requirements Specification</b>       | <b>11</b>  |
| 4.1 Software Requirements . . . . .              | 11         |
| 4.1.1 Descriptive list . . . . .                 | 11         |
| 4.2 Hardware Requirements . . . . .              | 12         |
| <b>5 Design &amp; Analysis</b>                   | <b>13</b>  |
| 5.1 System Analysis . . . . .                    | 13         |
| 5.1.1 Module breakup . . . . .                   | 13         |
| 5.1.2 Member effort . . . . .                    | 14         |

|          |                                  |           |
|----------|----------------------------------|-----------|
| 5.2      | System Design . . . . .          | 14        |
| 5.2.1    | Architecture . . . . .           | 14        |
| 5.2.2    | Data Flow Diagrams . . . . .     | 16        |
| <b>6</b> | <b>Implementation</b>            | <b>22</b> |
| 6.1      | Screen shots . . . . .           | 22        |
| 6.2      | Pseudo codes . . . . .           | 26        |
| 6.3      | Limitations . . . . .            | 35        |
| <b>7</b> | <b>Testing &amp; Maintenance</b> | <b>36</b> |
| 7.1      | Tests . . . . .                  | 36        |
| 7.1.1    | Unit Testing . . . . .           | 36        |
| 7.1.2    | System Testing . . . . .         | 37        |
| 7.2      | Maintenance . . . . .            | 38        |
| <b>8</b> | <b>Conclusion</b>                | <b>39</b> |
| 8.1      | Introduction . . . . .           | 39        |
| 8.2      | Future work . . . . .            | 40        |
|          | <b>References</b>                | <b>41</b> |

## ACKNOWLEDGEMENT

We take this opportunity to express our heartfelt gratitude to all respected personalities who had guided, inspired and helped us in the successful completion of this project.

First and foremost, we express our thanks to **The Lord Almighty** for guiding us in this endeavour and making it a success.

We are thankful to our Principal **Dr. U Lazar John** and the Management for providing us with excellent lab and infrastructure facilities.

Our sincere thanks to the Head of the Department of Computer Science & Engineering, Prof. **Muralee Krishnan C** for his valuable guidance and suggestions.

We would like to express our deepest gratitude to **Ms. Sujana A K** for her valuable contributions and guidance.

Last but not least, we thank all our teaching and non teaching staffs of Department of Computer Science & Engineering, and also our friends for their immense support and help in all the stages for the development of the project.

## **ABSTRACT**

Here we are designing on-line software for a company which provides outside contracts to other small companies. Main aim of the project is to provide security to the confidential files provided by the main company to the Outsourcing company for their work. Key to the project is to implement this security by providing complete control over all the documents provided by the company to the BPO. Project also provide security to data even though the records are accessed using internet. Main Company post file details to the site and interested companies can register through the site and apply for posted work. After scrutiny by main company, the new company if get sanctioned for the out job work, they can login to the site and add the team members for the work as well as their group names and each staff inside each group together with the IP address and Mac address of each staff. Only the staffs that are named in the list, can only login to the site and each time, the main company server check if the login is from the registered PC. If not, login will be expired. Thus all data will be securely handled by main company. Our project enhances secure transfer of files. This security is accomplished by the effective use of algorithms for the scanning the IP address and Mac addresses of each user. This project implements the safe and secure file transfer.

## List of Figures

|     |                                   |    |
|-----|-----------------------------------|----|
| 3.1 | The Spiral Software Process Model | 10 |
| 5.1 | Architecture                      | 15 |
| 5.2 | Level0 DFD                        | 16 |
| 5.3 | Level1 DFD Administrator          | 17 |
| 5.4 | Level1 Out Source Manager         | 18 |
| 5.5 | Administrator                     | 19 |
| 5.6 | Out Source Manager                | 20 |
| 5.7 | User                              | 21 |
| 6.1 | Login                             | 22 |
| 6.2 | Osm Registration                  | 23 |
| 6.3 | Admini File Permission            | 24 |
| 6.4 | OSM Edit File                     | 25 |
| 6.5 | OSM Change username password      | 25 |



## List of Tables

|     |                              |    |
|-----|------------------------------|----|
| 1.1 | Team Organization . . . . .  | 5  |
| 5.1 | Module Description . . . . . | 13 |
| 5.2 | Module Allocation . . . . .  | 14 |
| 7.1 | Unit test chart . . . . .    | 37 |

## List of Abbreviations

|             |  |
|-------------|--|
| <b>SFMS</b> | : <i>Secure File Management Software</i> |
| <b>SQL</b>  | : <i>Structured Query Language</i>       |
| <b>OSM</b>  | : <i>Outsource Manager</i>               |
| <b>FTP</b>  | : <i>File Transfer Protocol</i>          |
| <b>PC</b>   | : <i>Personal Computer</i>               |
| <b>IIS</b>  | : <i>Internet Information Server</i>     |
| <b>IE</b>   | : <i>Internet Explorer</i>               |
| <b>HTTP</b> | : <i>Hyper Text Transfer Protocol</i>    |
| <b>NIC</b>  | : <i>Network Interface Card</i>          |

(1)

---

## CHAPTER 1

### Introduction

#### 1.1 Overview

In our project we are developing software that can be used for secure management of confidential files by multi-national companies when they submit various jobs to BPO companies. Now a days information technology is so developed so that we can get every Information in one mouse click. In all existing system, all documents are handled manually. In this case there is a chance of getting documents corrupted. Owners of the file will lose complete control over the documents that they give to the companies for their various works. The objective of this project is to provide highly, efficient, secure, data transmission through the network so that information will not be given to wrong hands. The user will have complete control over all the documents. In the present project, the developed software acts as a middle ware between the user and the file server. The software enables the administrator an easy way to set permission for each user in the company for any specific file. Through better authentication and authorization, unauthorized attacks are minimized The main advantages of the proposed system are,

1. Allows data sharing.
2. Avoids unauthorized access on data.
3. User can obtain the privilege through permission setting.
4. Better authentication and authorization is provided.
5. The system can be further enhanced.
6. No need of leased lines for the secure data transfer
7. Use of a public domain i.e. the internet for secure file transfer
8. Easy to use interface.

The system has been designed to send documents to users in different parts of the country. This System is very user friendly. In this software the input forms are designed in such a way to make easy convenience of user and error free and as simpler as possible. For example, detail

views are used whenever possible so that the user can easily verify the required details from it. In this system we use the following modules for inputting the data:

1. Administrator module
2. Outsource manager module
3. User module.
4. File management module

### **1.1.1 Administrator Module**

Administrator, handle the overall system. Administrator set permission for the access of data from the server. It allows only authorized user to access the software. Administrator with valid username and password login to the software. It is then checked with that of in the user master table in the database. If it is valid then only the user is allowed to access the software, otherwise not. When administrator login in to the system, then the administrator can add the project and edit the project as needed. After the detail of the company has been read by the administrator, administrator can accept the company for outsourcing its work. The accepted company gets the user name and password for login. Administrator performs file allocation for the project.

### **1.1.2 Outsource Manager Module**

Outsource manager will be registered by the administrator login with valid username and password. Outsource manager adds new users with username, password, ip address and mac address. Outsource manager handles various functions like add user, group, file allocation, upload files, request for changing the password, username, ip and mac address.

### **1.1.3 User Module**

User work on files and edit the details. User can request for changing the user name, password ip and mac address for ensuring security. User can upload/download/work on files. User can view news.

## **1.2 Motivation and Technical Relevance**

Too many businesses are trusting critical data and processes to home-grown, do-it-yourself, insecure, unreliable, and manual processes. When one company needs to move and process important information and protect it while doing so you need a solution that has been specifically designed to meet those important security needs. That's what true Secured File management software is all about: Managing files as they transit your organization and your partners, managing those files through post-transfer processes, and securing and tracking those files to help you meet both internal and external security requirements. The objective of this project is to provide highly, efficient, secure, data transmission through the network so that information will not be given to wrong hands. The user will have complete control over all the documents. In the present project, the developed software acts as a middle ware between the user and the file server. The software enables the administrator an easy way to set permission for each user in the company for any specific file. Through better authentication and authorization, unauthorized attacks are minimized.

File transfer is an essential and important activity in the day-to-day computing world. Security lapses during file transfer can invite leak important data to the external world. As a result, securing FTP is of primary importance.

Secured File management software allows you to create a custom-branded, password-protected area where you can exchange project files with clients easily, securely, and professionally. Whether you deal with files that are simply too large to transfer by email, need secure file transfer, or just need a collaboration space where project-related files can be posted, Secured File management software can provide a solution for you.

### 1.3 Progress of project

Our project aims at development of online software which makes the secure transferring of files possible. We decided to implement this concept in a cooperative environment where the security of files is an inevitable option. Our idea was to setup software where the main administrator can post different projects to be done and various companies which are registered to the site can take up those tasks which they think that they could perform. Companies can add users who are the actual persons that work on the projects which they have chosen.

We found out that the best platform to implement our project is using Visual Studio and it would be perfect if use it with C# as programming language. The reason why we selected it is that it provides quality throughout all phases of development process, improved visibility; it brings all project artifacts data and status together in one place. It helps in effective design of the forms and provides option to link database software, The SQL Server. It helps in storing different details about the projects, users etc..

Our first task was to analyse this issue and decide how to implement our concept in a cooperative world. We designed our project with different modules. Module 1 is the Administrator module. This module includes the tasks that could be performed by the administrator. This may include posting of different projects, accepting companies, editing the projects in which different files [1] can be uploaded [2] and deleted, granting permission to the files uploaded by the companies, publishing notifications to alert the companies. We then designed various forms for each requirement. Created database for accordingly and linked it with the forms.

The next module is the outsource manager module which is the company. Any company could register to the site for accepting project works, but their registration becomes valid only if the administrator accepts their registration request. As soon as the OSM registers it is provided with a user name and password. We implemented an algorithm which uses the characters of the name of the OSM, special characters and numbers randomly to generate the username and password. The security is incorporated here by getting the IP address and the Mac address of the OSM system and storing it in the database. Even though the OSM has the username and password it could log in to the system only if it uses the system with the stored IP address and Mac address. The OSM can check all the projects available and choose the one or more. With these ideas we designed various forms required and created database corresponding to them.

Our next module is the File management module in which the files uploaded by the

administrator are encrypted so that there is no chance for a third party to access those data. The decryption is implemented in such a way that only the corresponding OSM with the password could decrypt and download the file. The password to decrypt the files are sent to the OSM via Email provided that the Email-id is given during registration. The algorithm we used for encryption is the AES algorithm, which is strong and resistant against the brute force attack. [3]

In the final stage we linked the forms together and implemented our project using the IIS sever. We found out that the project is working successfully.

## 1.4 Member roles and responsibilities

Our team consists of five members. The members of our group and their corresponding responsibilities are given in the table.

**1.1: Team Organization**

| <b>Name</b>   | <b>Role</b> |
|---------------|-------------|
| Sumi Jose     | Leader      |
| Mariya Anto   | Designer    |
| Mary Antony C | Debugger    |
| Merin Joseph  | Programmer  |
| Rose Mohan    | Programmer  |

Our project is divided in to four modules. The Administrator module which is concerned with how the administrator is given complete control over the files by setting permissions, managing the groups and user etc. The second is the Outsource module which is comparatively small and used for registering the users. The third, is the File management module which is used for the file manipulations. Finally we have to implement the User module. Each module needs a designing phase and a programming phase. Designing phase includes the creation of forms and the programming phase we need to write the code for each task to perform. We have divided the project among our members as designing and programming.

## 1.5 Layout

Here we present the outline of the contents in this project report. This provides a brief idea about the project and its implementation.

Chapter 2 presents the relevant documents referenced during the initial survey of the project concept. We present various papers we have referred and also different websites which we have browsed.

Chapter 3 presents the process model based on which we have carried out the software development process. Stage wise description of the project development is given here.

Chapter 4 includes the hardware and software requirements for the project. Presents various tools we have used to develop our project. Descriptions regarding the elements are also given.

Chapter 5 gives an overview of the schedule of the project work. Include member work effort and module allocations to each member here as per her responsibility. The section also presents the general architecture of our project concept. Also the different Data Flow diagrams used in our project development is also given.

Chapter 6 presents the screen shots of our project. Pseudo codes of the significant portions of the project is also included. This section also gives the limitations of our project.

Chapter 7 includes the details of the unit tests and system tests carried out and also proposals for future maintenance.

The last chapter, Chapter 8 summarizes the work done in our project. Also various future implementation that could be included in our project is also given.



---

## CHAPTER 2

### Literature Survey

#### 2.1 Documentation

This section includes the the literature survey we have conducted.

##### 2.1.1 Papers & websites

Some of the existing systems in this field are:

1. CC FILE TRANSFER [4]

It is a web based file transfer software built for people that need transfer file PC to PC regularly. it features in ease of use, speed, reliability, and security. CC File Transfer eliminates the hassles of FTP and limitations of Email. This program is designed to help you transfer file quickly and easily from PC to PC. The simple interface makes transferring file as easy as dragging and dropping. Best of all, CC File Transfer can transfer file over the Internet or home network using TCP/IP. CC File Transfer even lets you perform Windows-to-Linux transfers. If you need file transfer, you need CC File Transfer. Here, the transferred data is not secured. A third party can know the contents by packet sniffing.

2. MICROSOFT FILE TRANSFER MANAGER 5.0 [5]

File Transfer Manager is a desktop client-side application whose purpose is to provide secure and guaranteed delivery when downloading or uploading large files over the internet. There is a suspend/resume capability that allows users to resume interrupted transfers, pause and resume a file transfer as needed, multiple file transfers can be queued and prioritized, up to four transfer packages can be transferred concurrently to overcome the impact of network latency, built-in error checking and recovery from corruptions introduced in transport.

We have referred the following ieee papers for the purpose:

3. Evaluation Of Project Management Software Packages And Their Usefulness On Small Short Duration Projects [6]

It is pointed out that, on relatively small (5 million dollars) and/or short (1 wk.to 12 mo.) projects simple bar charts and seat of the pants decision making are usually used instead of an integrated work breakdown structure, activity network, and resource allocation approach. Two actual projects from 1989 were used to evaluate four 1989 project management software packages.

4. Reducing Efforts on Software Project Management using Software Package Re-usability [7]

In Software Development Process Model, the re-usability of elements can help to reduce the efforts of the project management for developing systems in a very short period. This paper focuses on the consecutive tasks like Domain Analysis, Package Analysis; and System Analysis for re-usability to minimize the required technical efforts in development area.

## CHAPTER 3

# Proposed System

### 3.1 Process Model

We decided to use the spiral model [8] of the software production process since it supports risk analysis. Risks are potentially adverse circumstances that may impair the development process and quality of products. The main characteristic of this model is that it is cyclic.

In the Stage 1 the objectives of the project in terms of quality is identified. Its is here we decided that our project requires an Admin who handles the whole system and as the clients of the admin, companies are needed. Admin allocates projects to the companies according to their requests and also provides required files related to it. The company consists of several employees. Thus there was a need of another module which we named as the user module which handles the functions of an employee of a company. Thus we started the design phase of our project.

In stage 2 the risk areas are identified. Here we realized that a secure login process so that no one other than the intended user could enter the system. Thus we decided to set a login name and password for the company and the user. But this method alone does not provide much security. So we thought of different idea in which the IP address and the mac address of the users are also used for login purposes. Several difficulties occurred in this stage. But by implementing several alternative we found out a good method to work out this idea.

In stage 3 The addition of projects by the Admin and the companies are implemented. Again the risk here was the security of the files managed. Since a public severer is used for the purpose the chance for sniffing and hacking is more. Thus we decided to implement encryption of the files uploaded by the Admin. We searched for a suitable algorithm for this process and finally we implemented AES algorithm [9] which is a powerful encryption algorithm. The companies using the same key could decrypt the files and download them. Then the problem which arose was how the required keys are made available to the companies. The alternative was to email it to the company's email id. This was implemented successfully.

The stage 4 consists of reviewing the results traverse so far and planning for the next iteration of the spiral. As the final implementation we hosted our project in a Local Area Network and tested successfully.

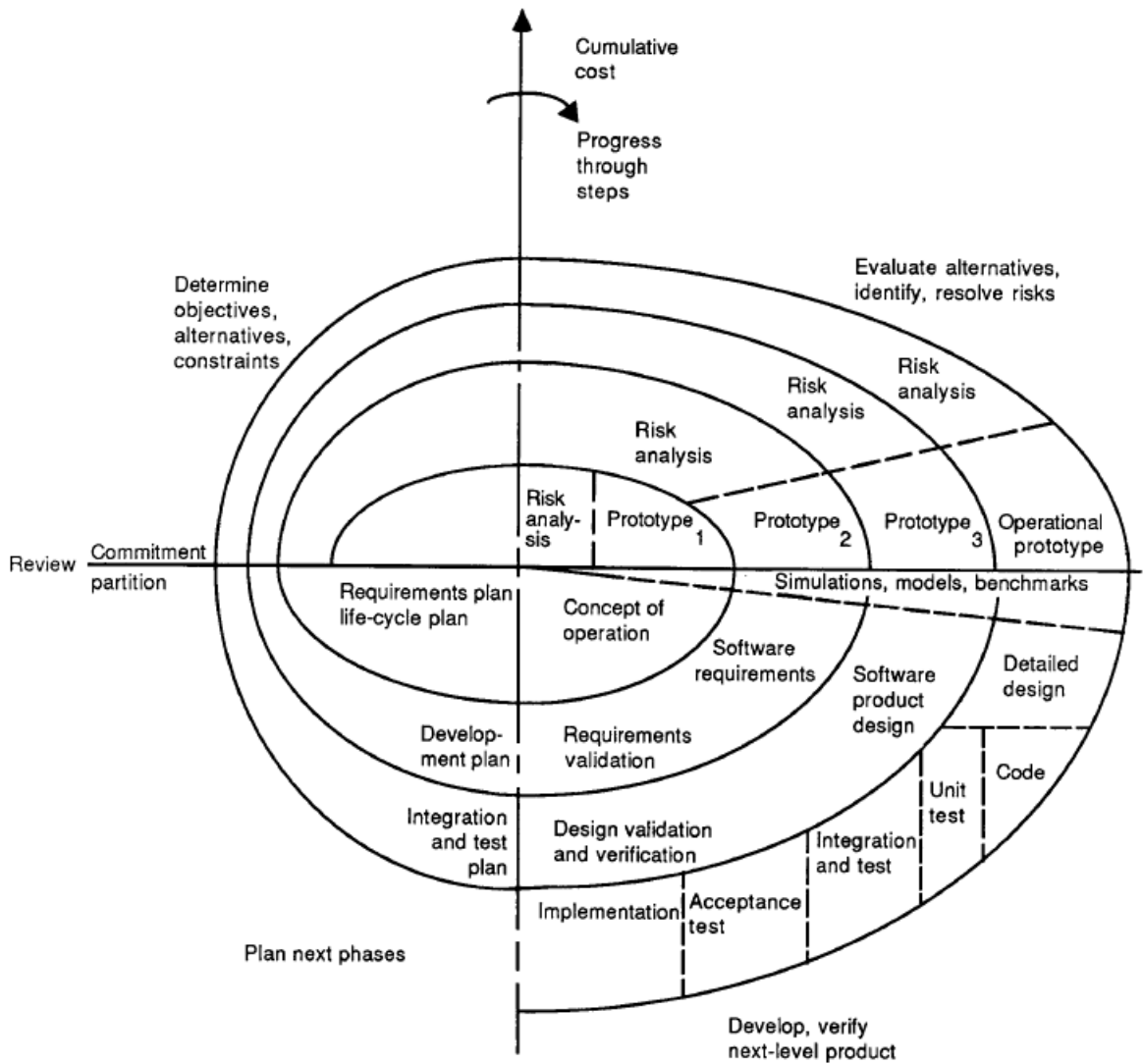


Fig. 3.1: The Spiral Software Process Model

---

## CHAPTER 4

### System Requirements Specification

#### 4.1 Software Requirements

1. Front End : ASP.NET
2. Back End : SQL Server 2005
3. Platform : Windows XP
4. Web server : IIS 5.0
5. Web client : IE 6.0
6. Internet standard : HTTP

##### 4.1.1 Descriptive list

**ASP.NET** ASP.NET is a programming framework built on the Common Language Runtime that can be used on a server to build powerful Web applications. Active Server Pages were released by Microsoft to enable the creation of dynamic pages based on user input and interaction with a website .ASP.NET improves upon the original Asp by providing code-behind .With Asp.NET and code behind, the code and the HTML can be separated.ASP.Net Web services Are XML based services that are exposed on the internet that can be accessed by other Web services and Web service clients.

**SQL Server 2005** SQL Server 2005, released in November 2005, is the successor to SQL Server 2000. It included native support for managing XML data, in addition to relational data.Microsoft SQL Server 2005 is comprehensive, integrated data management and analysis software that enables organizations to reliably manage mission-critical information and confidently run todays increasingly complex business applications. SQL Server 2005 allows companies to gain greater insight from their business information and achieve faster results for a competitive advantage. [10]

**IIS** Internet Information Server is a group of Internet servers (including a Web or Hypertext Transfer Protocol server and a File Transfer Protocol server) with additional capabilities for Microsoft's Windows NT and Windows 2000 Server operating systems. IIS is Microsoft's entry to compete in the Internet server market that is also addressed by Apache, Sun Microsystems, O'Reilly, and others. With IIS, Microsoft includes a set of programs for building and administering Web sites, a search engine, and support for writing Web-based applications that access database. [11]

## 4.2 Hardware Requirements

1. Processor : Pentium IV, 1.6 GHz
2. RAM : 512 MB DDR
3. Monitor : SVGA Samsung 591s
4. Keyboard : Samsung 105 Keys
5. Mouse : Logitech 3
6. CD-ROM : Samsung 52x
7. Hard Drive: 80GB HDD
8. NIC : Realtek

## CHAPTER 5

### Design & Analysis

#### 5.1 System Analysis

In this section we define different modules in our project and how the project work is been divided among the members.

##### 5.1.1 Module breakup

**5.1: Module Description**

| <b>Module</b>            | <b>Description</b>                  |
|--------------------------|-------------------------------------|
| Administrator module     | Handle the overall system.          |
| Outsource manager module | Handle various functions.           |
| User Module              | Work on files and edit the details. |
| File Management Module   | Login authentication and encryption |

Administrator, handle the overall system. Administrator set permission for the access of data from the server. It allows only authorized user to access the software. Administrator with valid username and password login to the software.

Outsource manager will be registered by the administrator login with valid username and password. Outsource manager adds new users with username, password, ip address and mac address. Outsource manager handles various functions like add user, group, file allocation and upload files. Change the password, username, IP and mac address.

User work on files and edit the details. User can change the user name, password IP and mac address for ensuring security. User can upload/download/work on files. User can view news.

File management module deals with the uploading of files by the different entities of the project. The files uploaded by the administrator requires more security hence it has to be encrypted. The encryption and decryption process is included in this module using the AES

### 5.1.2 Member effort

This section specifies the work done by the members of our team and the time required to complete each module in our project.

**5.2: Module Allocation**

| # | Task              | Estimated Effort | Start Date   | End Date     | Person            |
|---|-------------------|------------------|--------------|--------------|-------------------|
| 1 | Administrator     | (4man-hrs)       | (28/10/2010) | (30/11/2010) | Sumi,Merin,Mariya |
| 2 | Outsource manager | (3man-hrs)       | (1/12/2010)  | (3/3/2011)   | Mary,Rose         |
| 3 | User              | (3man-hrs)       | (2/1/2011)   | (12/4/2011)  | Rose,Merin        |
| 4 | File management)  | (2man-hrs)       | (12/11/2010) | (18/4/2011)  | All               |

## 5.2 System Design

### 5.2.1 Architecture

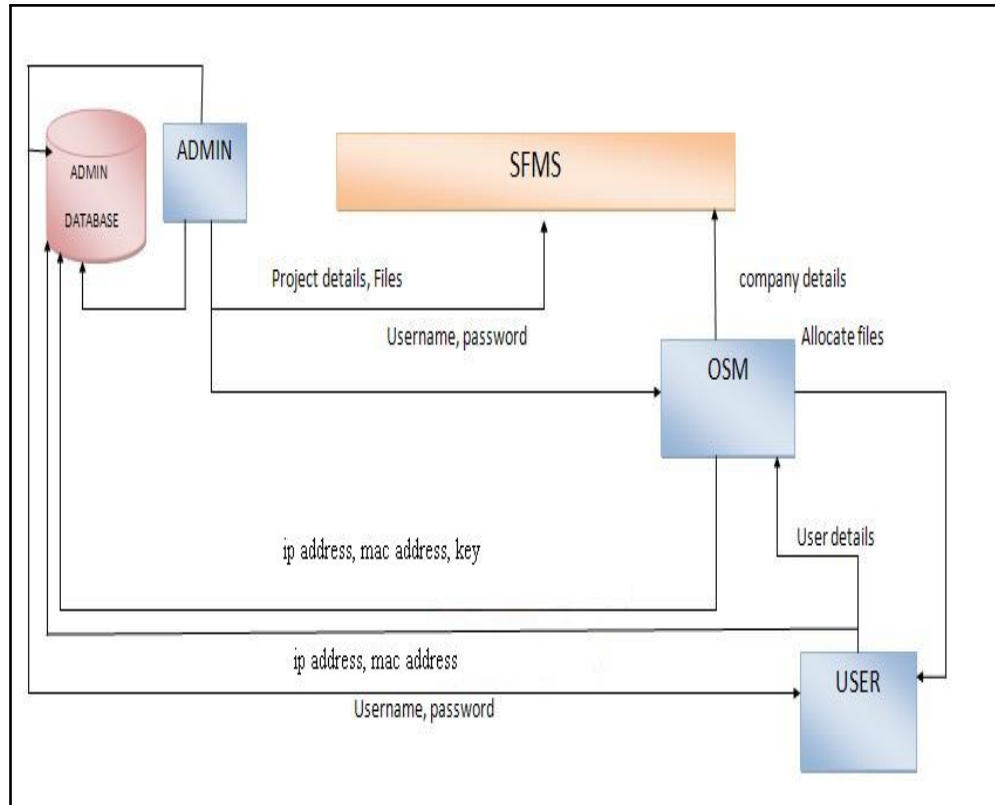
We have divided our system into three levels. The first level is level 0. It is represented by fig 5.2. The secure file management (SFMS) is accessed by the three modules: Administrator module, outsource manager module, and user module in different ways.

The second level is level 1. It is specified separately for administrator module, outsource module and user module. In the level 1 of administrator module, an admin accesses the software using a username which is divided by user itself. adminlogin is the database to store this user name and password. This is shown in fig 5.3. In the level of outsource module, the OSM registers into the site and searches for the desired task. The OSM then applies for the required task to admin. Admin user verifies the registration details and if it decides to accept, the company details are stored in the database add-company. Admin user then generates a username and password for the particular OSM. OSM will then use this to access the file provided by admin. In the level 1 of the user module, the OSM will register the user by making a request to admin, giving necessary details. After verification by admin, the user will be accepted or rejected. It will store the details of the user in the database user-request. Admin will generate the necessary username and password which is used to access items.

In the level 2 of the Admin module, we have implemented the following. The add-project will add the details of new projects in the database addnewproject. When we are editing the project, the corresponding project is retrieved from the file allocate table. The uploaded files are stored in the file allocate table and the file permissions are set in the file path table. To add news



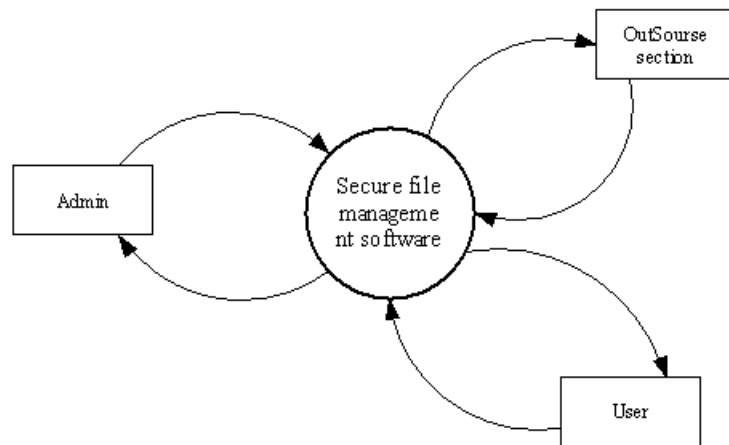
we are using the table addnews. User management module manages the users by storing the details of the users, accepting and rejecting users, adding group users etc.



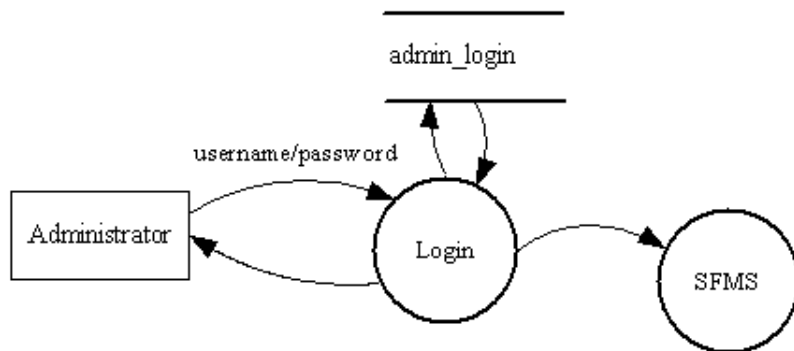
**Fig. 5.1: Architecture**

## 5.2.2 Data Flow Diagrams

In this section we present the data flow diagrams associated with our project. The description of the figures are given in the architecture section.



**Fig. 5.2: Level0 DFD**



**Fig. 5.3: Level1 DFD Administrator**

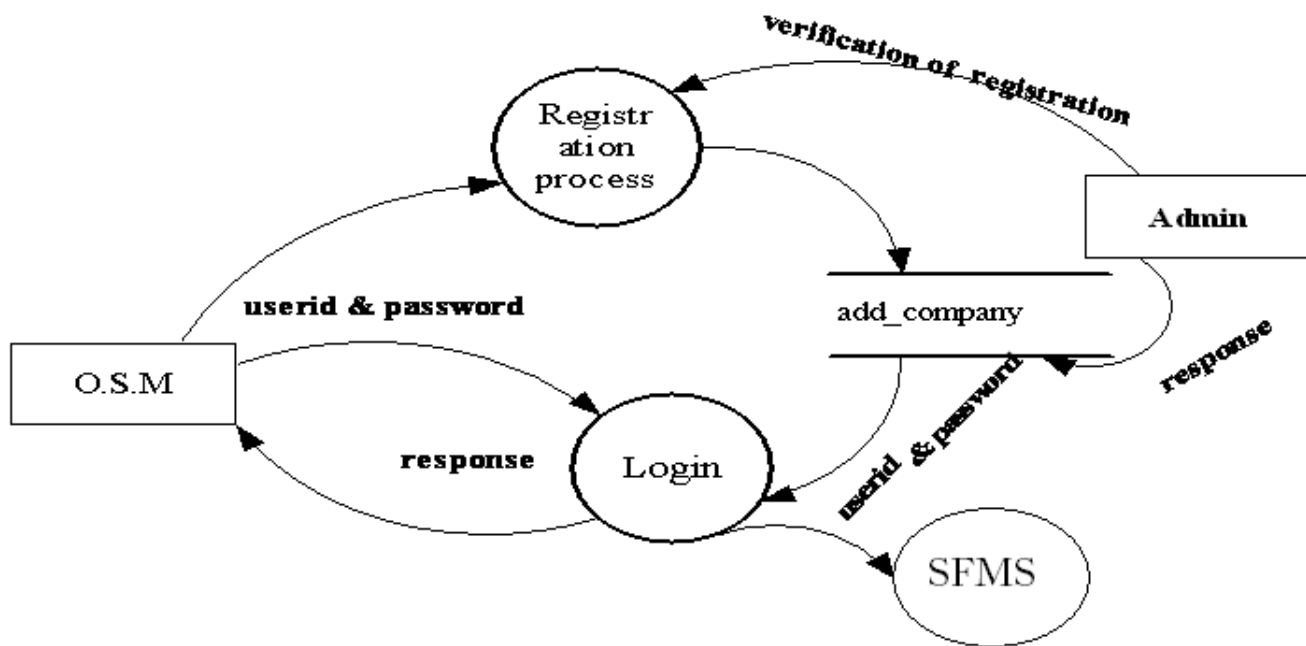


Fig. 5.4: Level1 Out Source Manager

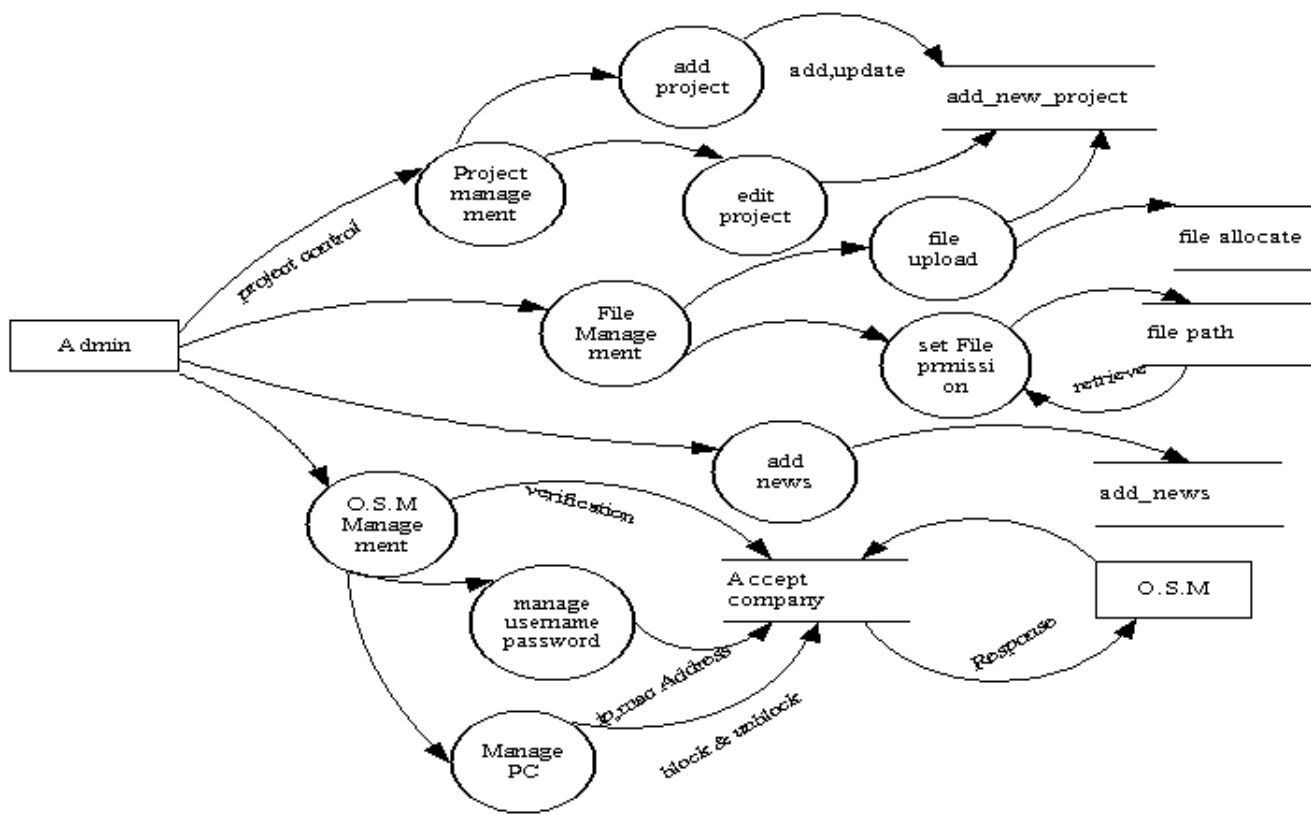


Fig. 5.5: Administrator

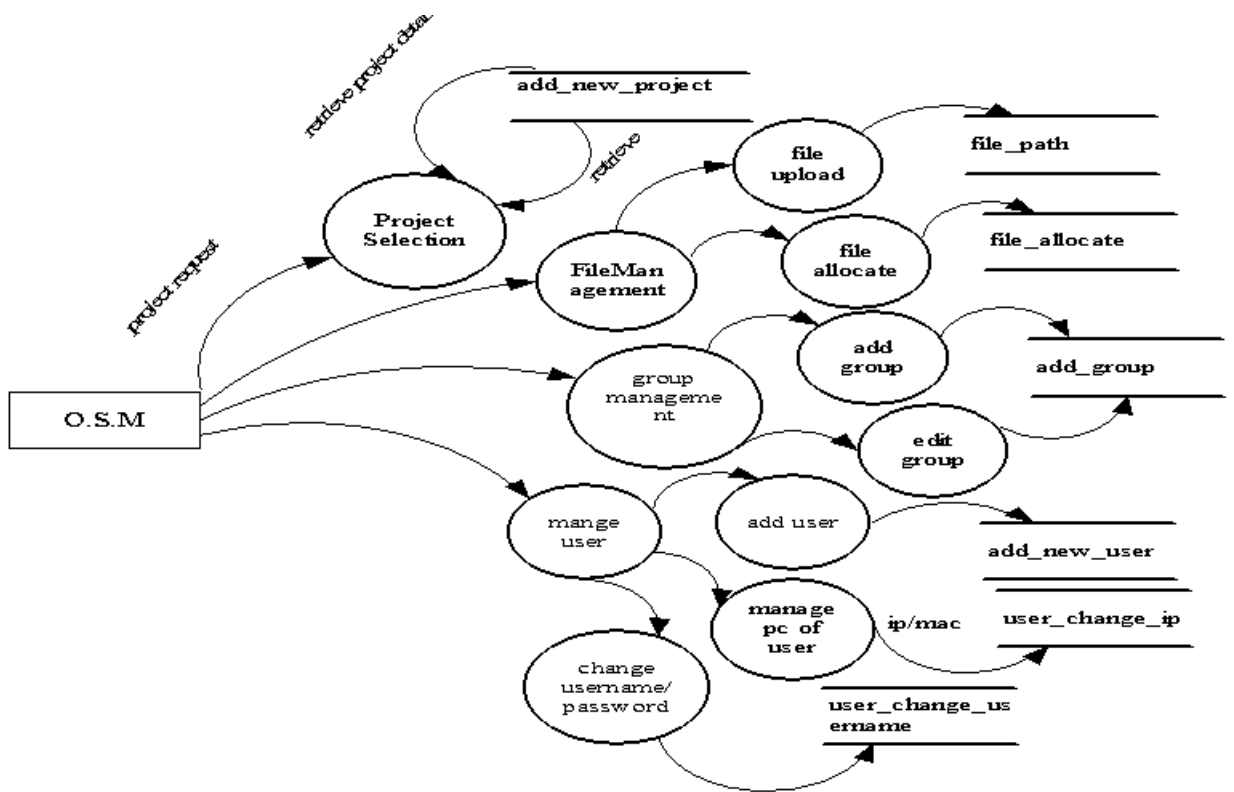


Fig. 5.6: Out Source Manager

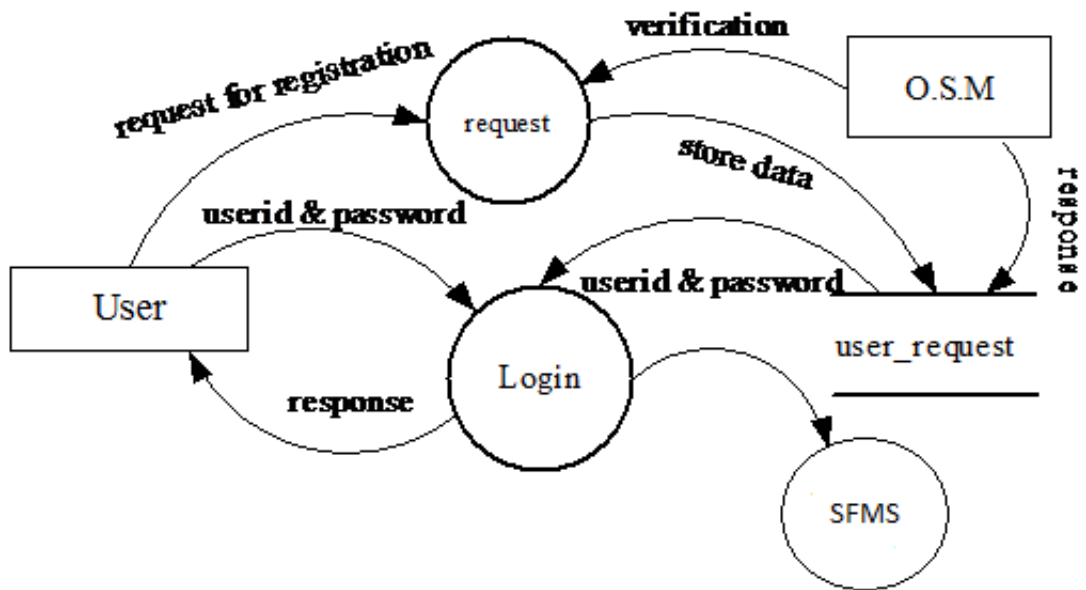


Fig. 5.7: User

# CHAPTER 6

## Implementation

### 6.1 Screen shots

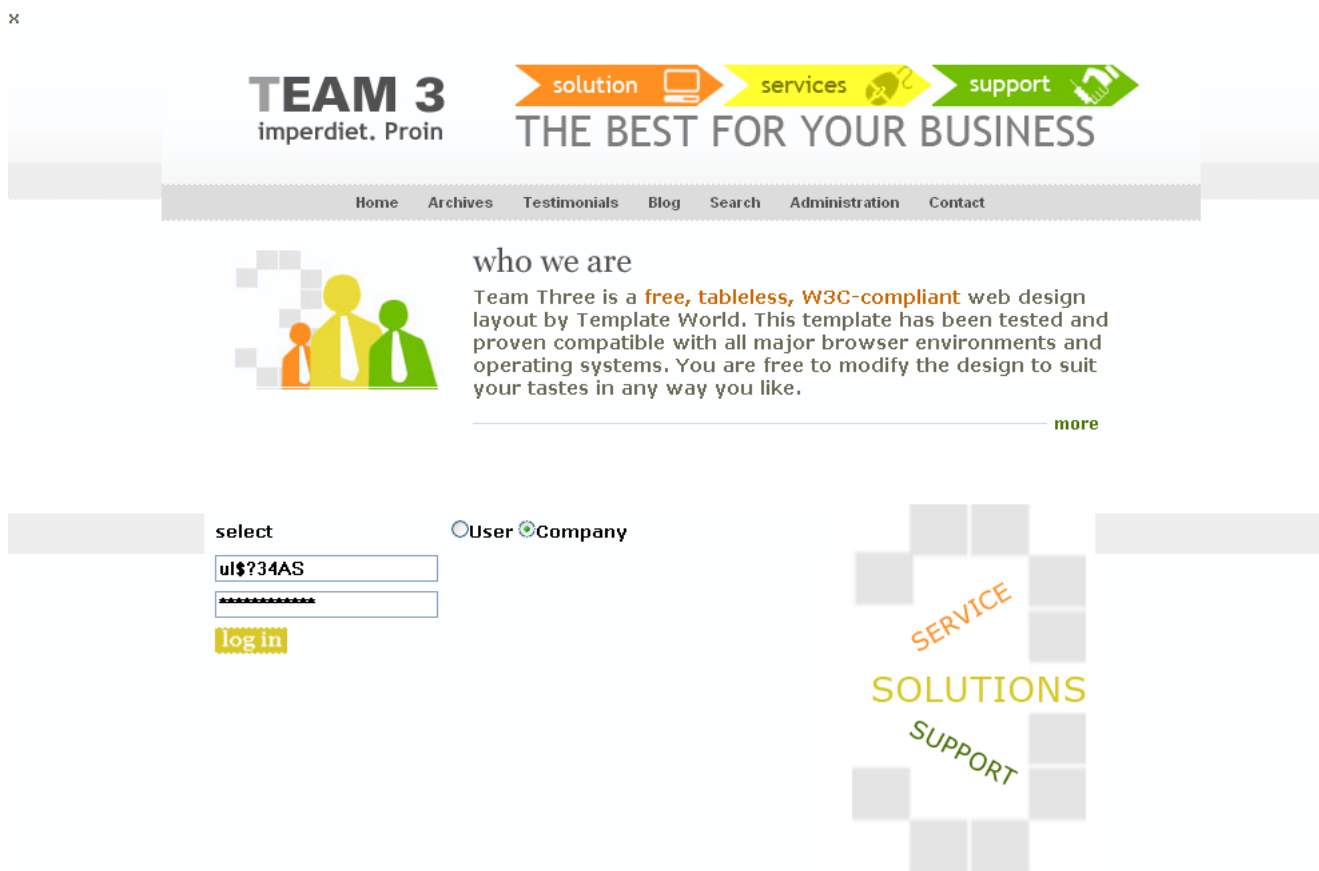



Fig. 6.1: Login






solution
services
support

## THE BEST FOR YOUR BUSINESS

---

[Home](#)   [Archives](#)   [Testimonials](#)   [Blog](#)   [Search](#)   [Administration](#)   [Contact](#)



### who we are

Team Three is a **free, tableless, W3C-compliant** web design layout by Template World. This template has been tested and proven compatible with all major browser environments and operating systems. You are free to modify the design to suit your tastes in any way you like.

[more](#)

#### Registration of Outsource Manager

|   |                       |
|---|-----------------------|
| Company Name  | GOKUL & CO            |
| <a href="#">Check Availability</a>  | Name Available        |
| Licence No  | EAG4535               |
| Reg : No  | 587457                |
| Regtd Date  | 06/23/2003            |
| Place   | kerala                |
| Phone Number  | 0458227585            |
| Web Site  | www.gokul.com         |
| E - Mail Address  | emailtogokul@ymail.co |
| <input checked="" type="checkbox"/> I have read, understood and agree to these <a href="#">Terms and conditions</a>                                 |                       |
| <span style="background-color: green; color: white; padding: 5px 15px; border-radius: 5px; display: inline-block; cursor: pointer;">Register</span> |                       |

#### Your Login Details

|           |          |
|-----------|----------|
| User Name | go?@52CO |
| Password  | 13gO~@gO |









Fig. 6.2: Osm Registration


solution 
services 
support 

## THE BEST FOR YOUR BUSINESS

---

[Home](#) [Archives](#) [Testimonials](#) [Blog](#) [Search](#) [Administration](#) [Contact](#)




### who we are

Team Three is a **free, tableless, W3C-compliant** web design layout by Template World. This template has been tested and proven compatible with all major browser environments and operating systems. You are free to modify the design to suit your tastes in any way you like.

[more](#)

### current links

- [Add New Project](#)
- [Companies](#)
- [Edit Project](#)
- [File Permission](#)
- [Add/Edit News](#)
- [Signout](#)



Select Category :  Company  Users

Select Company :  v

#### Company Files..

|   |                                     |
|---|-------------------------------------|
| id  | 1                                   |
| url   | Admin//haii//Rescued document.txt   |
| owner   | Admin                               |
| project_name  | haii                                |
| name  | IBM                                 |
| permission  | <input checked="" type="checkbox"/> |
| <input type="button" value="Allow"/> <input type="button" value="Block"/> |                                     |

1 2 3

Fig. 6.3: Admini File Permission

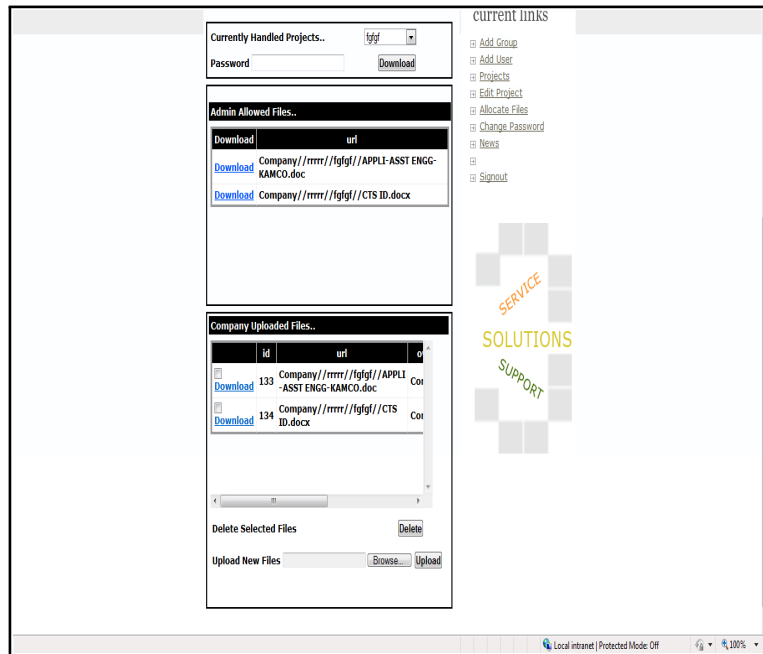


Fig. 6.4: OSM Edit File



Fig. 6.5: OSM Change username password

## 6.2 Pseudo codes

Here we include the codes used in implementing our project. The entire prototype is implemented using VC#. And the web pages are designed in ASP.NET.

Admin\_add\_New\_project

```
protected void lnk_check_Click(object sender , EventArgs e)
{
    if (txt_pjctname.Text != "")
    {
        flag = 0;
        getflag();
        if (flag == 1)
        {
            lbl_status.ForeColor = System.Drawing.Color.Red;
            lbl_status.Text = "Project_name_already_exist";
        }
        else
        {
            lbl_status.ForeColor = System.Drawing.Color.Green;
            lbl_status.Text = "Project_Available";
        }
    }
    else
    {
        lbl_status.Text = "Enter_Project_Name";
        lbl_status.ForeColor = System.Drawing.Color.Red;
    }
}
```

```
protected void btn_pjct_Click(object sender , EventArgs e)
```

```

{
    try
    {
        string url;
        byte [] keySalt = new byte [] { 0x49, 0x76, 0x76,
            0x61, 0x6e, 0x20, 0x4d, 0x65, 0x64, 0x76, 0x65,
            0x64, 0x65, 0x76 };
        HttpFileCollection hfc = Request.Files;
        string filepath = "";

        if (dt.Rows.Count > 0)
        {
            lbl_warng.Text = "Project_Already_Exists";
        }
        else
        {
            System.IO.Directory.CreateDirectory(Server.MapPath(".") +
                "//Admin//" + txt_pjctname.Text);
            for (int i = 0; i < hfc.Count; i++)
            {
                HttpPostedFile hpf = hfc[i];
                string [] a = hpf.FileName.Split('.');

                if ((hpf.ContentLength > 0) || (a[1] == "docx") ||
                    (a[1] == "txt"))
                {
                    hpf.SaveAs(MapPath("sample//" +
                        System.IO.Path.GetFileName(hpf.FileName)));
                    Encrypt(MapPath("~/sample/" +
                        System.IO.Path.GetFileName(hpf.FileName)),
                        MapPath("Admin//" + txt_pjctname.Text + "//" +
                        System.IO.Path.GetFileName(hpf.FileName)),
                        txt_key.Text, keySalt);
                    if (i == hfc.Count - 1)
                    {
                        filepath = filepath + "Admin//" + txt_pjctname.Text +
                            "//" + System.IO.Path.GetFileName(hpf.FileName);
                    }
                }
            }
        }
    }
}

```

```

    }
    else
    {
        filepath = filepath + "Admin//" + txt_pjctname.Text +
            "//" + System.IO.Path.GetFileName(hpf.FileName)
            + ",";
    }
    url = "Admin//" + txt_pjctname.Text + "//" +
        System.IO.Path.GetFileName(hpf.FileName);

    obj.execquery("insert_into_files_path_values('"
        + url + "','Admin','"
        + txt_pjctname.Text + "','Admin','True')");
    lbl_warng.Text = "Project_Successfully_Added";
    }
}
obj.execquery("insert_into_add_new_project_values
('" + txt_pjctname.Text + "','"
    + txt_tools.Text + "','" + txt_desc.Text + "','"
    + txt_web.Text + "','"
    + txt_bks.Text + "','" + filepath + "','"
    + txt_startdate.Text + "','"
    + txt_lastdate.Text + "','"
    + txt_amount.Text + "','Not_Set','"
    + txt_key.Text + "')");
    lbl_warng.Text = "Project_Successfully_Added";
    lbl_warng.ForeColor = System.Drawing.Color.Green;
    clear();
}
}
catch(Exception x)
{
    lbl_warng.Text = x.Message;
    lbl_warng.ForeColor = System.Drawing.Color.Red;
}
}

```

Admin\_Edit\_project

```
public static void Encrypt(string fileIn , string fileOut ,
    string pass , byte[] keySalt)
{
    PasswordDeriveBytes pdb = new PasswordDeriveBytes
    (pass , keySalt);
    Rijndael alg = Rijndael.Create();
    alg.Key = pdb.GetBytes(32);
    alg.IV = pdb.GetBytes(16);
    using (FileStream fsOut = new FileStream(fileOut ,
        FileMode.OpenOrCreate , FileAccess.Write))
    {
        using (FileStream fsIn = new FileStream(fileIn ,
            FileMode.Open , FileAccess.Read))
        {
            using (BinaryReader br = new BinaryReader(fsIn ,
                Encoding.Default))
            {
                using (CryptoStream cs = new CryptoStream(fsOut ,
                    alg.CreateEncryptor() , CryptoStreamMode.Write))
                {
                    int len = (int)br.BaseStream.Length;
                    cs.Write(br.ReadBytes(len) , 0 , len);
                    cs.FlushFinalBlock();
                }
            }
        }
    }
}
```

## OSM Registration

```

protected void imgbtn_reg_Click(object sender , ImageClickEventArgs
{
    try
    {
        DataTable dt = obj.getdatatable
("select * from add_company where comp_name ='" + txt_comp_name.Text + "'");

        if (dt.Rows.Count > 0)
        {
            lbl_warng.Text = "Company Already Exists";
        }
        else
        {
            if (CheckBox1.Checked == true)
            {
                string username = "";
                username = obj.getusername(txt_comp_name.Text);

                string password = "";
                password = obj.getpassword(txt_comp_name.Text ,
                    username);

                ip = getip();
                string mac = getmac();
                lbl_username.Text = username;
                lbl_passwd.Text = password;
                DateTime dat = Convert.ToDateTime
                (txt_date.Text.ToString());
                obj.execquery
                ("insert into add_company values ('" + txt_comp_name.Text + "',''"
                + txt_licence.Text + "',''" + txt_reg.Text + "',''"
                + dat.ToShortDateString() + "',''"
                + txt_place.Text + "',''"
                + txt_phn.Text + "',''" + txt_web.Text + "',''"
                + txt_email.Text + "',''" + lbl_username.Text + "'");
            }
        }
    }
}

```



```

        + lbl_passwd.Text + "',''" + ip + "',''" + mac
        + "','','False','0')");
        string msg = "Your_Username—>"
+ username + ";Password_is—>" + password;
        SendMail("sft.jec@gmail.com", txt_email.Text ,
        "Login_Information", msg);
    }
    else
    {
        lbl_warng.Text = "Accept_our_Terms_and_Conditions";
    }
}
}
catch (Exception x)
{
    lbl_warng.Text = x.Message;
    lbl_warng.ForeColor = System.Drawing.
    Color.Red;
}
}

public string getip()
{
    ip = Dns.GetHostByName(Dns.GetHostName()).
    AddressList[0].ToString();
    return ip;
}

public string getmac()
{
    try
    {
        string strip;

        DirectoryEntry DomainEntry = new DirectoryEntry
        ("WinNT://" + "workgroup" + "");
        DomainEntry.Children.SchemaFilter.Add("Computer");
    }
}

```

```
        foreach (DirectoryEntry machine in DomainEntry.Children)
        {
            string strMachineName = machine.Name;

            IPAddress IPAddress;
            try
            {
                IPAddress = getIPByName(machine.Name);
                strip = IPAddress.ToString();
            }
            catch
            {
                continue;
            }

            if (ip == strip)
            {
                strMACAddress = getMACAddress(IPAddress);
                //txtmac.Text = strMACAddress;
            }
        }
    }
    catch (Exception x)
    {
        lbl_warng.Text = x.Message;
    }
    return strMACAddress;
}
```

OSM Work project

```

protected void btn_download_Click(object sender , EventArgs e)
{
    GridView1.Visible = true;
    try
    {

        byte[] keySalt = new byte[] { 0x49, 0x76, 0x76,
            0x61, 0x6e,
            0x20, 0x4d, 0x65, 0x64, 0x76, 0x65, 0x64, 0x65, 0x76 };
        DataTable dt = obj.getdatatable
("select * from files_path where project_name=
'"+ ddl_projects.SelectedItem.Text +
"' and owner='Admin'");

        for (int i = 0; i < dt.Rows.Count; i++)
        {
            DataRow dr = dt1.NewRow();
            string encd_path = dt.Rows[i]["url"].ToString();
            string[] file_name = encd_path.Split('/');
            try
            {
                //Decryp (MapPath(encd_path),
                MapPath(encd_path), txt_passwd.Text, keySalt);
                Decryp(MapPath(encd_path), MapPath("Company//"+
                Session["comp_name"].ToString() + "//"
                + ddl_projects.Text + "//"+
                file_name[file_name.Length-1]),txt_passwd.Text,
                keySalt);
                dr["url"] += "Company//"+
                Session["comp_name"].ToString() + "//"
                + ddl_projects.Text + "//"
                + file_name[file_name.Length - 1];
            }
            catch (Exception x)
            {
                GridView1.Visible = false;
            }
        }
    }
}

```

```
        lbl_warng_passwd.Text = "Wrong_Password";
    }
    dt1.Rows.Add(dr);
}
GridView1.DataSource = dt1;
GridView1.DataBind();
GrdVw_company.DataBind();
}
}
catch (Exception x)
{
    lbl_warng_passwd.Text = x.Message;
}
}
```

### **6.3 Limitations**

SFMS is an online software which provides security to file transfer. Since it provides tight security, there are certain limitations. Our software does not provide any recovery option in the case of webserver crash.

For the security assurance, three trials are allowed for correcting the password, after that it is not possible to access through this. This software does not provide the forgot password option. The details of such problems will be then discussed through email.

Another limitation is that the multi admins are not allowed. The admin who owns the software, can only access it and will not allow other admin to use this. The buyer has to host the web site for their own purpose.

## CHAPTER 7

# Testing & Maintenance

### 7.1 Tests

Software testing is defined as the process by which one detects the defects in software. It is considered as the final opportunity for covert/rectify and us to detect that were in the software. Testing is the process which is done with explicit intention of finding errors that make the program fail. In short, system and quality assurance is a review of the software products and related documents for completion correctness, reliability and maintenance. The first step in system testing is to prepare a plan that will test all accepts of the system.

The common view of testing is that it is performed to pure that there are no errors in the program. But this is quite difficult in the analyst cannot pure that software is free from all sort of error. There for, the most useful practical approach is with the understanding the testing is that process of exiting program with the explicit indention of finding errors. That is to make the program fail. A successful test can be there for, one that find an error.

#### 7.1.1 Unit Testing

This is first level of testing in which different modules are tested against the specifications produced during the design of modules. Unit testing is done for the verification of the code produced during the coding phase and to test internal logic of the modules.

It refers to the verification of single program modules in an isolated environment. Unit testing focuses first on each module, to locate error after coding each page was tested and run individually. All unnecessarily coded statements were removed and it was ensured that all the functionality worked expected. Any logical errors found were corrected. All modular units of Secure File Management Software were tested and got approved.

Admin module: The admin module was first implemented and tested. Since it doesnt concern with ip address or mac address testing was easy. Each of the forms were tested separately and tested if it is working properly as a whole. The username and password was set default.

**7.1: Unit test chart**

| No | Unit Name                  | Test Status |
|----|----------------------------|-------------|
| 1  | Administrator Module 1     | Complete    |
| 2  | Outsource Manager Module   | Complete    |
| 3  | User Module                | Complete    |
| 4  | File Management Controller | Complete    |

Outsource manager module: To test the OSM module ip and mac address is needed. So first the testing was done by setting a default ip and mac of the system. Each of the forms were tested and finally the module was tested as a whole.

Later the admin and OSM module was tested together to see if it is working properly with the default settings

User module: To test the user module also ip and mac address is needed. So a default value of ip and mac was set and each of the forms were tested individually and the module as a whole was tested.

File Management module: Testing of this module included uploading and downloading of different types of file. We carried out different trials to find out whether the proper file management is taking place and also tested to whether media files could be handled.

### 7.1.2 System Testing

System testing is carried out as two phases in which the first one is the integration testing and second is the acceptance testing.

Integration testing is a systematic technique for constructing the program structures while conducted test to uncover errors associated with interfacing. Many tested modules are combined into subsystem, which is then tested. This testing is the verification of the interfaces among system parts. Integration address the issues associated with the dual problems of verification and program construction Administrative module interconnected by each module was tested to verify the realblitiy of the software and validity. Since each other module is strongly depended on the administrative module, the integration will reveal all errors in each module.

Bottom up integration is the traditional strategy used integrates the components of software system into functioning whole. This integration consists unit testing, followed by subsystem testing, followed by testing the entire system. A subsystem consists of several modules that can communicate with each other through well-defined interfaces. The primary concern here is compatibility of individual modules. Though each module works independently

they should work after linking them together.

After all the modules were designed they were combined. Whole system testing was done within a single system using default values. After that the system was connected to a LAN and actually obtained the IP and MAC addresses. Even then testing was done in a single system. The next step was to host the website in the local LAN and system was again tested. The testing showed variations from the expected result as many of the features were not supported by the IIS server. Appropriate modifications were made and the system was tested until it executed successfully.

Acceptance testing involves planning and execution of functional tests and in order to demonstrate that the implemented software satisfies its requirements. This software is feasible through that it can accommodate the future needs existing in the system. So maintains becomes easy using software.

## 7.2 Maintenance

Maintenance is the set of activities that are performed to modify the project after it is delivered. This phase mainly consist of correcting any remaining errors in the system, adapting the application to changes in the environment and improving, changing or adding features and qualities to our project.

Several steps have to be implemented in the maintenance of the system. Some of the suggestions include:

- It is better to timely update the password generation algorithm
- Changes in the code has to be made to support compaitability issue of IIS server
- It is better to timely update the username and password of the admin



## **CHAPTER 8**

### **Conclusion**

#### **8.1 Introduction**

This project deals with the security of data provided by a company to the BPO for various works. For giving complete control of the file to the owner various file permissions can be set by the owner. Each user will be provided a particular a username and password given by the administrator. With that username and password users can login to the system. Password is used for security. Each user has provided a particular a username and password given by the administrator. With that username and password users can login to the system. Administrator also has password to login to the system. It ensures security. Nobody can enter the system without correct username and password. Only registered user can access to the files. For providing more security, not only check the username, password but also the ip and mac address of the system. So the user with correct username, password and ip and mac address of users pc also correctly match with the value on the database. Any unauthorized access to the system can be blocked.

## 8.2 Future work

Our project is an outline of a secure project management software. It has a wide area of application in the cooperate world and it is very useful for employees to work even though they are not located at the same places. The major advantage of our project is the security it provides event hough it work on a public server.

- As the future implementation our project could be hosted in to the internet so that world wide clients could access the facilities provided.
- Implementing multiple admins makes the project a vast venture with a large number of clients whose requirements could be satisfied with in this project itself.
- Provisions to solve situations when the user forgets username and password could be implemented.
- Also options for instant messaging and voice over internet protocol (VOIP)can be included which makes the interaction among the different users who work on same project becomes possible.

## REFERENCES

- [1] [Online]. Available: <http://www.dotnetcurry.com/ShowArticle.aspx?ID=317>
- [2] [Online]. Available: <http://www.developerfusion.com/code/4505/add-a-javascript-popup-to-an-aspnet-button/>
- [3] [Online]. Available: <http://www.dotnetspark.com/Forum/1338-encryption-and-decryption.aspx>
- [4] [Online]. Available: <http://www.cc-file-transfer.com/large-file-transfer.com>
- [5] [Online]. Available: <http://microsoft-file-transfer-manager.software.informer.com/5.0/>
- [6] [Online]. Available: [http://ieeexplore.ieee.org/search/freesrchabstract.jsp?tp=&arnumber=183640&queryText3Devaluation+of+project+management26openedRefinements3D\\*26searchField3DSearch+All](http://ieeexplore.ieee.org/search/freesrchabstract.jsp?tp=&arnumber=183640&queryText3Devaluation+of+project+management26openedRefinements3D*26searchField3DSearch+All)
- [7] [Online]. Available: [http://ieeexplore.ieee.org/search/freesrchabstract.jsp?tp=&arnumber=4809260&queryText3Dreducing+efforts+on+software+project26openedRefinements3D\\*26searchField3DSearch+All](http://ieeexplore.ieee.org/search/freesrchabstract.jsp?tp=&arnumber=4809260&queryText3Dreducing+efforts+on+software+project26openedRefinements3D*26searchField3DSearch+All)
- [8] [Online]. Available: <http://searchsoftwarequality.techtarget.com/definition/spiralmodel>
- [9] [Online]. Available: <http://www.obviex.com/samples/Encryption.aspx>
- [10] [Online]. Available: <http://visualcsharp-tutorials.com/2010/12/sql-basics/>
- [11] [Online]. Available: <http://en.wikipedia.org/wiki/IIS>