# A Minor Project Report on

File Management System

Submitted in Partial Fulfillment of the Requirements for

Degree of **Bachelor of Engineering in Information Technology**

under Pokhara University

Submitted by:

**Anit Chaudhary, 12402**

**Bipin Subedi, 12411**

**Suresh Pokhrel, 12448**

Under the Supervision of:

**MR.Niranjan Khakurel**

Date:

2016-04-16

 **Department of Information Technology**

**NEPAL COLLEGE OF**

**INFORMATION TECHNOLOGY**

Balkumari, Lalitpur, Nepal

# Abstract

*File Management System(FMS) is web-based application for the systematic handling of the files in the organization to organize the files recorded in database. FMS ensures the security violation with use of certified login. Simply, it is intended find intended files uploaded in the database just sitting by the chair. FMS introduces chat for the users within organization for chatting with the co-workers. File Dash board is for every users in the organization. Privilege for every user is set different to read, modify and delete.*

# Acknowlegdement

We would express our deepest appreciation to all those who provided us the possibility to complete this report.

         Furthermore, we would like to acknowledge with much appreciation the crucial role of the staffs of NCIT College and teachers who gave the opportunity to revive our skills and implement the theoretical knowledge in the real time projects.

           A special thanks to team mates, who helped in every steps and gave suggestion about the project. Lastly, many thanks go to the head of department of IT Mr. Madan Kadariya and our supervisior Mr.Niranjan Khakurel, who have invested his full effort in guiding the team. We have to appreciate the guidance the seniors especially in our project presentation that has improved our presentation skills thanks to their comment and advices.

Contents

[Abstract i](#_Toc448562919)

[Acknowlegdement ii](#_Toc448562920)

[List of Tables iv](#_Toc448562921)

[List Of Figures iv](#_Toc448562922)

[1. Introduction 1](#_Toc448562923)

[1.1. Problem Statement 2](#_Toc448562924)

[1.2. Signification of study 3](#_Toc448562925)

[1.3. Project objective 4](#_Toc448562926)

[1.4. Scope 5](#_Toc448562927)

[2. Project Development Cycle 6](#_Toc448562928)

[3. Architectural Skeleton 7](#_Toc448562929)

[Presentation tier: 7](#_Toc448562930)

[Application and Logical Tier: 7](#_Toc448562931)

[Data Tier: 7](#_Toc448562932)

[4. Requirements 8](#_Toc448562933)

[4.1. Functional Requirements 8](#_Toc448562934)

[4.2. Non Functional Requirements 8](#_Toc448562935)

[5. Teams and Tools 9](#_Toc448562936)

[5.1. Teams 9](#_Toc448562937)

[5.2. Tools 9](#_Toc448562938)

[6. System Design 10](#_Toc448562939)

[6.1. Use-Case Diagram 10](#_Toc448562940)

[Use case 1: login 10](#_Toc448562941)

[Use case 2: Upload 11](#_Toc448562942)

[Usecase 3: search 11](#_Toc448562943)

[Usecase 4:File Listing and Deleting 12](#_Toc448562944)

[Usecase 5: Employee Profile 12](#_Toc448562945)

[6.2. Sequence Diagram 13](#_Toc448562946)

[6.3. Activity Diagram 14](#_Toc448562947)

[6.4. Class Diagram 15](#_Toc448562948)

[7. Testing Result 16](#_Toc448562949)

[Unit Testing 16](#_Toc448562950)

[Integration testing 18](#_Toc448562951)

[8. Budget/Cost estimation 19](#_Toc448562952)

[9. Time Schedule 21](#_Toc448562953)

[9. Snapshots : 22](#_Toc448562954)

[Bibliography 25](#_Toc448562955)

# List of Tables

[Table 1: Functional Requirement 8](#_Toc448562956)

[Table 2: Non-Functional Requirement 8](#_Toc448562957)

[Table 3:Team Roles 9](#_Toc448562958)

[Table 4: Tools Requirement 9](#_Toc448562959)

# List Of Figures

[Figure 1: Waterfall Model 6](#_Toc448562960)

[Figure 2: Use Case Diagram 10](#_Toc448562961)

[Figure 3: Login Sequence Diagram 13](#_Toc448562962)

[Figure 4: General spring MVC Architecture 4 sequence diagram 13](#_Toc448562963)

[Figure 5: Activity Diagram 14](#_Toc448562964)

[Figure 6:Class Diagram 15](#_Toc448562965)

[Figure 7: DashBoard 22](#_Toc448562966)

[Figure 9 :file Upload 22](#_Toc448562967)

[Figure 10: login 23](#_Toc448562968)

[Figure 11: Employee Profile Upload 23](#_Toc448562969)

[Figure 12: Searching 24](#_Toc448562970)

# Introduction

**File management**, formerly known as data management, is the upgraded version of the previously made desktop-based application as our mini project. It is intended to control the storing and accessing of data from web-browser. In our application, we use real time folder system as the reference to the database storage of any organization. FMS is java based web-application that focuses its usage within the organization. FMS come up search of files from database, chat with the clients. File Management System must store path of file, preserve the data inside the folder, maintain the hierarchy. Nonetheless, FMS also assists in deletion and insertion of the database contents according to the privileges set by the application.

FMS, validated the multiple clients login with specific username and passwords from login frame.

# Problem Statement

Likely thousands of files increase in our organization, accessing these files is very tremendous waste of time for every time to search junk of files in shelf. Queuing Crowds of the people hour to hour in the offices has been common criteria to get work done. This made to come up with an idea that what if the file management is made digital rather handy works which will be beneficial for the client as well as employee. As we see files being lost, tore, misplace, theft and so on they should maintain the security in the storing of the files. Secure accessing of the database can provide the high security and provide effective management of the files.

# Signification of study

When the user practice good in such system in the computer , then they will find that working on our system with ease .However we use in our Business Organization ,the benefits of managing our files are numerous .

There are likely many files in our computers ,some may need each day some may need occationally so accessing these files is very important to navigate in our system. Since our system FMS is a just an example  project but is helpful in rid out of the burden in managing of files. This system is adding and deletion of the files added by the user with all the information of files with instance date . With such system,  highly confidencial file are secure and is efficient to time and memory.

# Project objective

Main objectives of our project is to give easy layout for effective navigation to the files within the organization. FMS provides dashboard to each and every user of the FMS with some privilege. Secure browsing with login and privilege. Protect the document and files in the computer so that external hazard will not affect the files.

# Scope

Every computer system needs to have some way to manage the files that it contains. The way that each system manages files is different and unique. The way to save information on the system should make life simpler on the user. Managing files and storage is the major part of file management system.

Uploading, deleting, downloading files by users makes the efficient use of file management system. The main scope if our web app be seen in our government offices where the documents of every citizen can be assign in the computerized way in the storage where any kinds of external hazards(eg. worn out due to improper handling, wind, water, fire) will not effect the documents. Providing document whenever necessary through computer less than a time is one of the main feature of file management system.

# Project Development Cycle

**Linear sequential method** is also called “Classic Life Cycle” or “Waterfall” model or “Software Life Cycle” suggests a systematic and sequential approach to software development that begins at the system level and progresses through analysis, design, coding, testing and support. The waterfall model derives its name due to the cascading effect from one phase. In this model each phase well defined starting and ending point, with identifiable deliveries to the next phase .

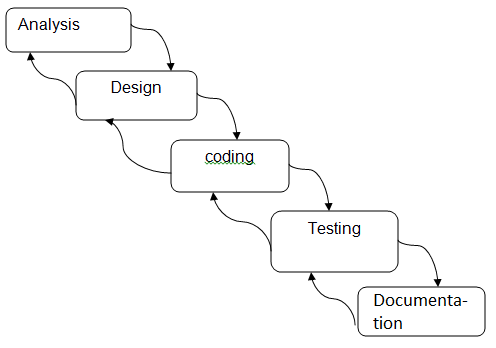


Figure 1: Waterfall Model

* + 1. **Discipline**
* Requirement
* Analysis and Design
* Implementation
* Testing
* Documentation

# Architectural Skeleton

We use 3-tier architecture for development of our project:

* Presentation Tier
* Logical Tier
* Data Tier

## Presentation tier:

The GUI interface makes the user friendly environment for the interaction of the system. The login and logout through the GUI interface restrict the unauthorized user

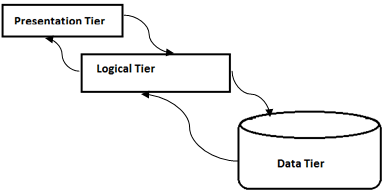
## Application and Logical Tier:

The logical manipulation of the data accessed by the user like while login, while searching of files is evaluated over here. It is the bridge connection of both Tier. MVC architecture is followed in this architecture.

.

## Data Tier:

The data or files that needed to stored or taken out are stored in the database. Our application/logical tier may request information from the data tier, it then processes this information and returns it to the presentation tier in response to the user request.



# Requirements

## Functional Requirements

|  |  |
| --- | --- |
| **Requirements** | **Priority** |
| Users enables with login environment | Needed |
| Users can a upload of files | Needed |
| Users can chat with each other | Needed |
| Database stores the user’s data files as well as predefined data sets | Needed |
| System shows message notification | Needed |
| System shows as per their online status | Optional |

Table 1: Functional Requirement

## Non Functional Requirements

|  |  |
| --- | --- |
| **Requirements** | **Priority** |
| System works on Internet Browsers | Essential |
| System must be user friendly | Essential |
| Runs on operating systems (windows,linux,and any other) | User’s desire |

Table 2: Non-Functional Requirement

# Teams and Tools

## Teams

Role of each individual team members are:

|  |  |  |
| --- | --- | --- |
| **Titles** | **Name** | **Roles** |
| Developer and SQA | Anit Chaudhary | Develops the application according to plan and specification produced. |
| Designing , Project Manager,Documenta-tion | Bipin Subedi | Designing ,Manage overall project ,indentify threads and risk occurring in entire system development |
| Testing | Suresh Pokhrel | Develops test plan, test cases and evaluate so as to check whether implementation satisfies the requirement perfectly or not. |

Table 3:Team Roles

## Tools

Tools Required for the development of our project are:

|  |  |
| --- | --- |
| **Tools** | **Purpose** |
| Browser(Mozilla firefox , internet explorer etc.) | Browsing |
| Eclipse | Coding |
| Project libre | Time schedule |
| My-Sql | Database design |
| Edraw Max 7.9 | Design |
| Visual paradigm | Design |

Table 4: Tools Requirement

# System Design

## Use-Case Diagram

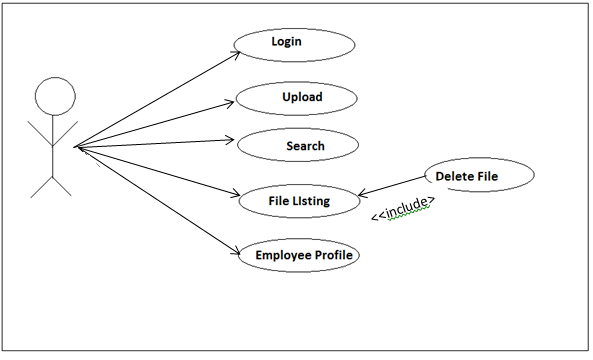


Figure 2: Use Case Diagram

### Use case 1: login

|  |  |
| --- | --- |
| **Name** | Login |
| **Actor** | Employee |
| **Precondition** | User is not login in |
| **Description** | 1. Employee types his/her username and password.  2. Employee click Login button.  3. User is Login.  If user is get incorrect password then login page again get display |

### Use case 2: Upload

|  |  |
| --- | --- |
| **Name** | Upload |
| **Actor** | Employee |
| **Precondition** | File is not uploading. |
| **Description** | 1. User Enter Each and every information of files in a form.  2. Employee browses the file for Upload.  3. Employee clicks the Submit button. |

### Usecase 3: search

|  |  |
| --- | --- |
| **Name** | Search |
| **Actor** | Employee |
| **Precondition** | Employee searches the file. |
| **Description** | 1. Employee types the file name for search.  2. he/she clicks the search button.  3. Searched file get displayed in dashboard. |

### Usecase 4:File Listing and Deleting

|  |  |
| --- | --- |
| **Name** | File Listing |
| **Actor** | Employee |
| **Precondition** | All files information on the database get displayed |
| **Include** | Delete files button |
| **Description** | 1. User Enter Each and every information of files in a form.  2. Employee browses the file for Upload.  3.employee Delete the files through the listed files |

### Usecase 5: Employee Profile

|  |  |
| --- | --- |
| **Name** | Employee Profile |
| **Actor** | Employee |
| **Precondition** | All employee information on the database get displayed |
| **Description** | 1. Employee Click the Profile Button. 2. Employee Profiles get Displayed. |

# Sequence Diagram

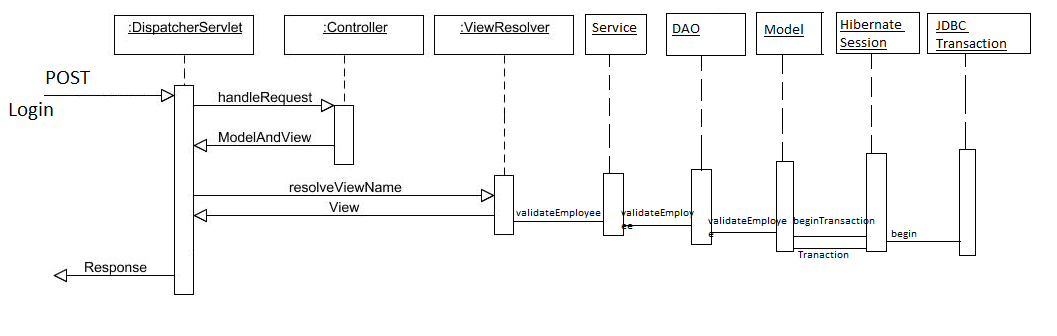


Figure 3: Login Sequence Diagram

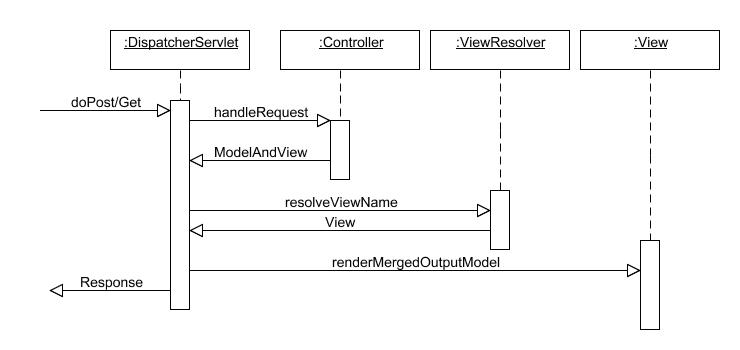


Figure 4: General spring MVC Architecture 4 sequence diagram

## Activity Diagram

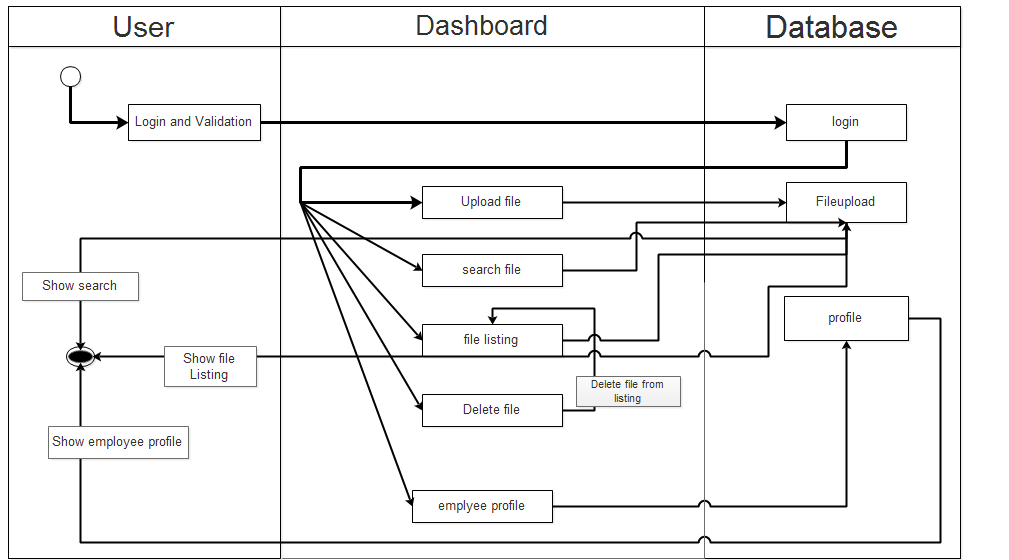


Figure 5: Activity Diagram

## Class Diagram

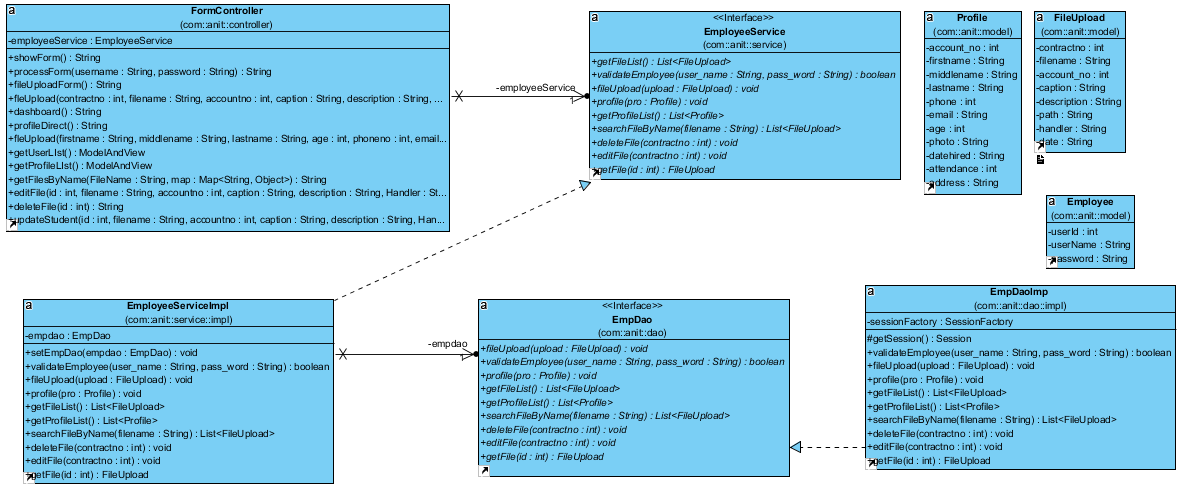


Figure 6:Class Diagram

# Testing Result

Testing should done for the well-functioning of the system. We tested each and every part of our System for the intent of the end user. Alpha Testing has been conducted in the controlled environment in the developer side in the presence of developer.

## Unit Testing

**For login**

|  |  |  |
| --- | --- | --- |
| User Name | password | Test Result |
| Anit | anit123 | OK |
| Bipin | Bipinhello | OK |

**For Employee entry**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A/c no. | First name | Middle Name | Last Name | Age | Phn no. | Email | Photo | Test result |
| 1 | Anit | Thary | Chaudhary | 20 | 9809892921 | [greyhatanit@gmail.com](mailto:greyhatanit@gmail.com) | C:\..... | OK |
| 2. | Bipin | Mitra | Subedi | 20 | 9860213404 | [Bpn.subedi@gmail.com](mailto:Bpn.subedi@gmail.com) | D:\..... | OK |

**For File upload**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| contract no. | File name | a/c no | Caption | Description | Path | Handler |
| 1 | Login.jpg | 3400 | Code | Coder | C:\........ | Anit |
| 2. | Emp.jpg | 4000 | Img | Image | D:\......... | Bipin |

|  |  |
| --- | --- |
| Date | Test Result |
| 2014-11-13 | OK |
| 2014-1-09-14 | OK |

**For Searching**

|  |  |
| --- | --- |
| Search | Login.jpg |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| contract no. | File name | a/c no | Caption | Description | Path | Handler | Date | Test  Result |
| 1 | Login.jpg | 3400 | Code | Coder | C:\........ | Anit | 2014-11-13 | Detected |

**For Delete**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A/c no. | First name | Middle Name | Last Name | Age | Phn no. | Email | Photo | Delete |
| 1 | Anit | Thary | Chaudhary | 20 | 9809892921 | [greyhatanit@gmail.com](mailto:greyhatanit@gmail.com) | C:\..... | Delete |
| 2. | Bipin | Mitra | Subedi | 20 | 9860213404 | [Bpn.subedi@gmail.com](mailto:Bpn.subedi@gmail.com) | D:\..... | Delete |

|  |  |
| --- | --- |
| Delete | Anit |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A/c no. | First name | Middle Name | Last Name | Age | Phn no. | Email | Photo | Delete |
| 2. | Bipin | Mitra | Subedi | 20 | 9860213404 | [Bpn.subedi@gmail.com](mailto:Bpn.subedi@gmail.com) | D:\..... | Delete |

|  |
| --- |
| Test Result |
| OK |
| OK |

## Integration testing

We perform integration testing unit testing, obviously we need to perform integration testing by integrating our each and every part of our system.

**Login**

|  |  |  |
| --- | --- | --- |
| User Name | password | Test Result |
| Anit | anit123 | OK |
| Bipin | Bipinhello | OK |

|  |
| --- |
| Dashboard |
| Upload |
| Search |
| File Listing |
| Delete |
| Employee profile |

|  |
| --- |
| Test Result |
| OK |
| OK |
| OK |
| OK |
| OK |

# Budget/Cost estimation

We use simple Counting of the lines of code for the budget estimation where we add

avg. productivity=10 kloc /person-per-month

Person per Month = 15000 per month

|  |  |  |  |
| --- | --- | --- | --- |
| Measurement Parameter | Count | Avg-value | FP count |
| No. of Inputs | 3 | 4 | 12 |
| No. of Outputs | 4 | 5 | 20 |
| No. of Logical Files | 3 | 3 | 9 |
| No. of User Inquiries | 3 | 10 | 30 |
| No. of External Interface | 1 | 7 | 7 |
|  | | Total Count | 78 |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Controller | FormController | 250 |
| Service | EmpService | 20 |
|  | EmpServiceImpl | 122 |
| Dao | EmpDao | 20 |
|  | EmpDaoImpl | 173 |
| Model | Employee | 45 |
|  | Profile | 131 |
|  | FileUpload | 84 |
| Views | Dashboard.jsp | 1452 |
|  | Edit.jsp | 32 |
|  | fileList.jsp | 1476 |
|  | fileUpload.jsp | 1478 |
|  | Profile.jsp | 1494 |
|  | profileList.jsp | 1487 |
|  | Submitform.jsp | 108 |
|  | updateUser.jsp  Index.jsp | 71  1 |
| Spring-dispatcher-servlet.xml |  | 60 |
| Web.xml |  | 20 |
|  | Total Estimated KLOC | 8534 |

Estimated KLOC= 8534

FP estimate=total count \*(o.65+0.01\*∑(FP))

= 78 \*1.17

= 91.16

≈ 92

Effort= FP/avg. productivity

= 92/10

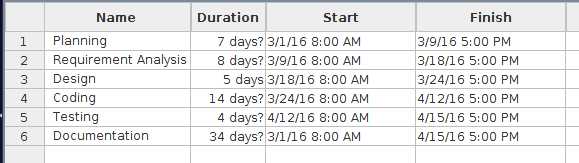
= 9.2

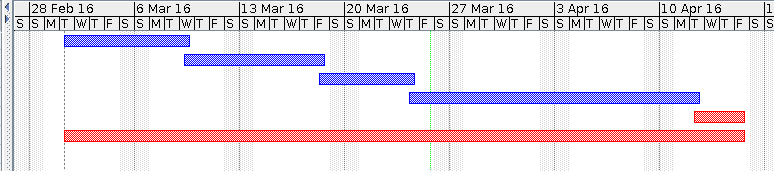
Total Project Cost= FP\* Labor rate / Productivity

= 92 \* 15000/10

= Rs. 1,38,000

# Time Schedule





# Snapshots :

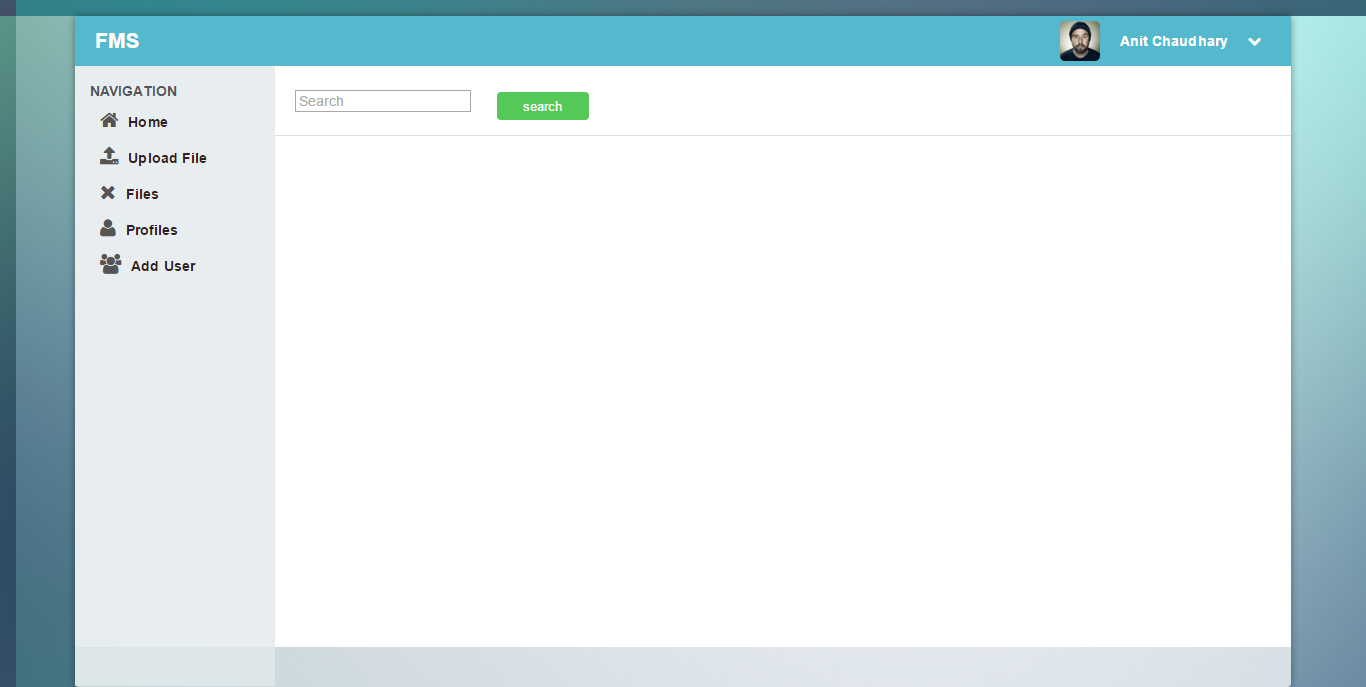


Figure 7: DashBoard

12

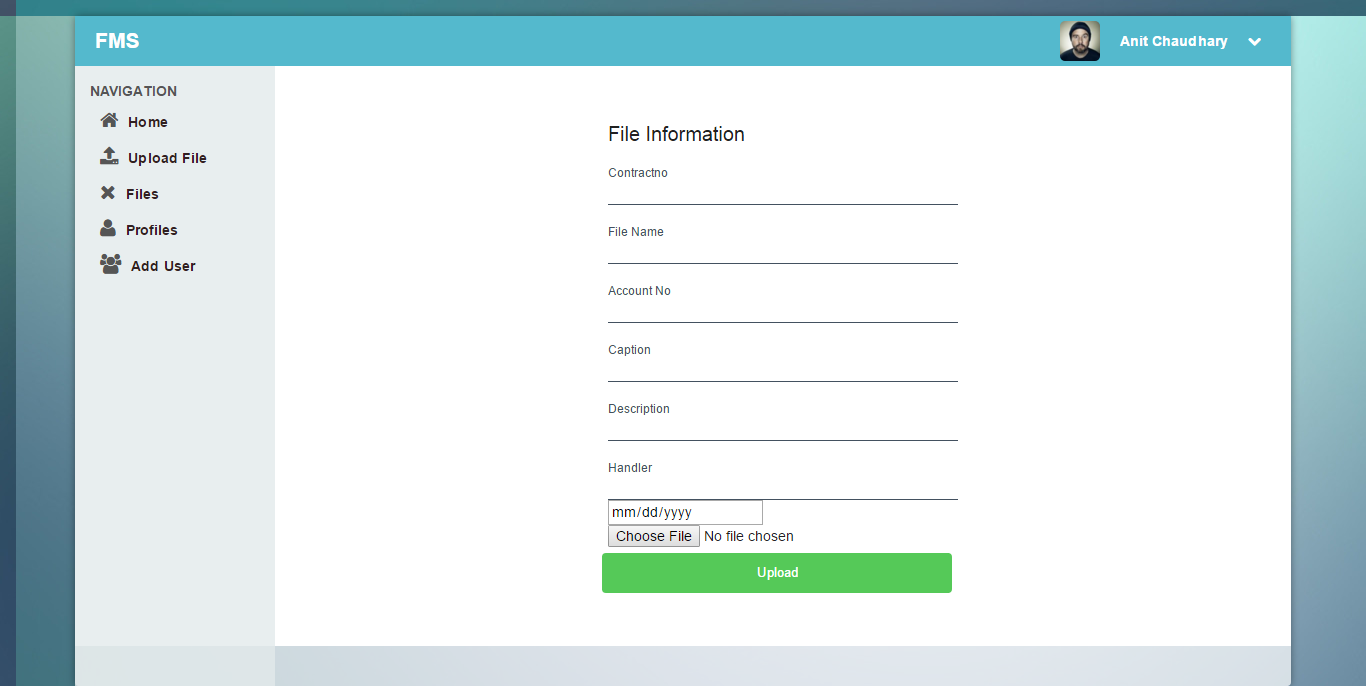


Figure 9 :file Upload

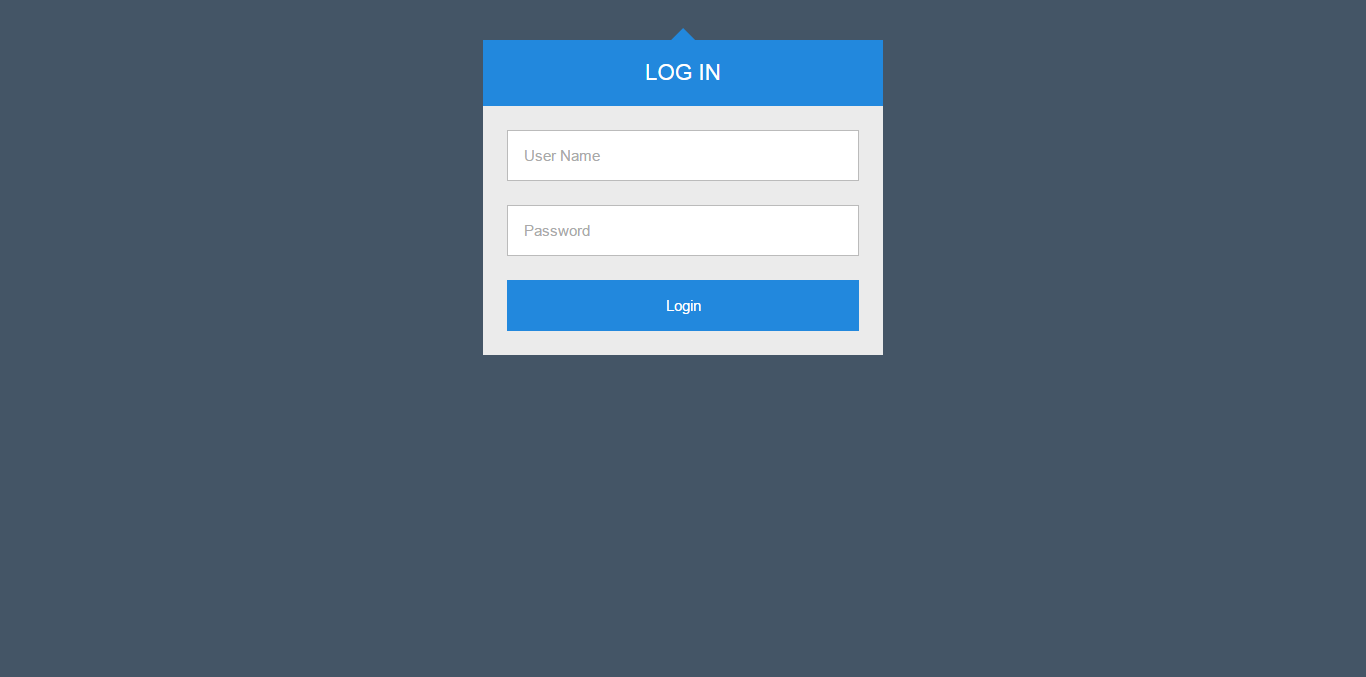


Figure 10: login

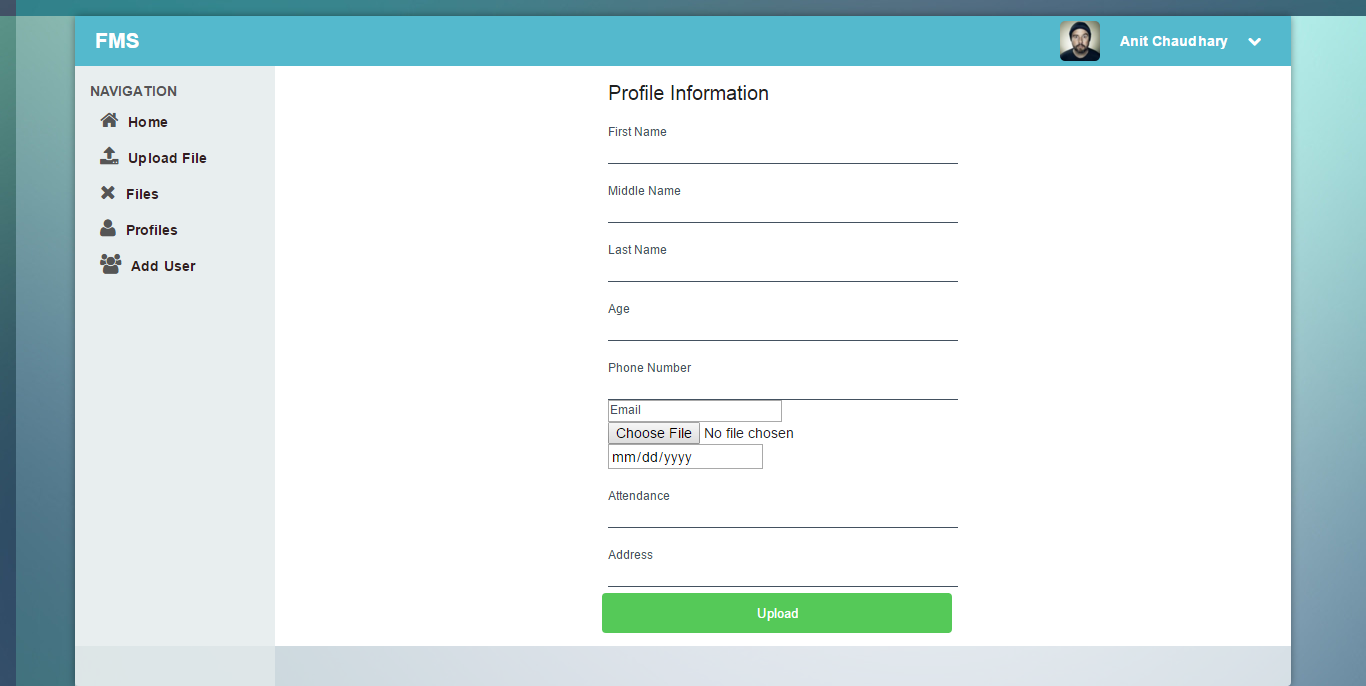


Figure 11: Employee Profile Upload

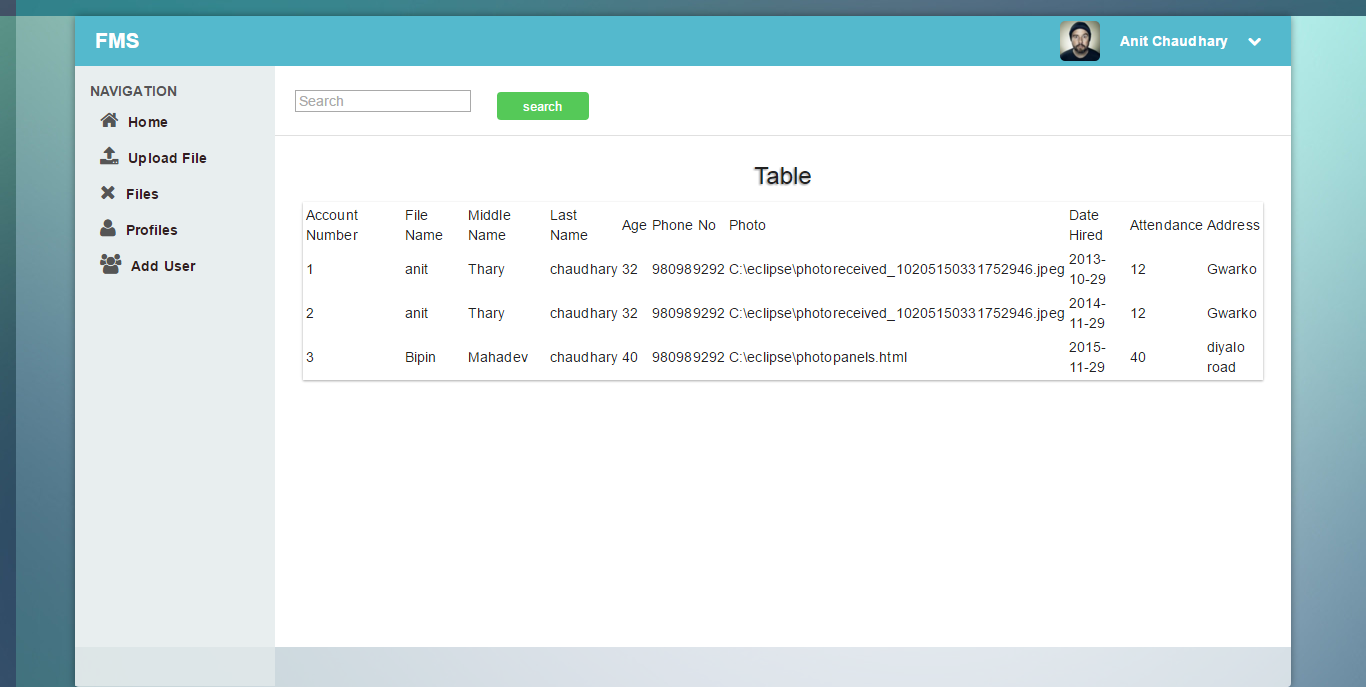


Figure 12: Searching

# Bibliography

[1] Software Engineering Fundamentals, Robert S Pressman

[2] W3 School, Web: http://www.w3schools.com/accessed at: 3th April,2016

[3] “The Aptech Advanced Series”, Publication: November 2000

[4]Spring Framework ,Web:http://www.tutorialspoint.com/spring/ accessed at: 4th April 2016

[5]Hibernate, Web: <http://www.tutorialspoint.com/hibernate/> accessed at: 2th April, 2016