FILE-PORT

A web based file manager for JUIT

Project Report submitted in partial fulfillment of the requirement for the degree of

Bachelor of Technology

in

Computer Science & Engineering

under the Supervision of

Mrs. Sanjana Singh

By

Palak Kashyap

111244 (CSE)

to

Jaypee University of Information and Technology

Waknaghat, Solan – 173234, Himachal Pradesh

Certificate

This is to certify that project report entitled “FILE PORT – A web based file manager for JUIT”, submitted by Palak Kashyap (111244) in partial fulfillment for the award of degree of Bachelor of Technology in Computer Science & Engineering to Jaypee University of Information Technology, Waknaghat, Solan has been carried out under my supervision.

This work has not been submitted partially or fully to any other University or Institute for the award of this or any other degree or diploma.

Date: **Mrs. Sanjana Singh**

Assistant Professor

Acknowledgement

I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals. I would like to extend my sincere thanks to all of them.

I am highly indebted to Mrs. Sanjana Singh for her guidance and constant supervision as well as for providing necessary information regarding the project & also for his support in implementing the project.

My thanks and appreciations also go to my colleague in developing the project and people who have willingly helped me out with their abilities.

Date: Palak Kashyap

Table of Content

List of Figures

Abstract

The debate of using paper documents over electronic documents has been a long topic of interest. Although both have their merits and demerits, with advancing technologies, it is a wiser decision to slowly upgrade the processes in any industry and shift to electronic file systems to make best use of the available technologies and make operations more efficient. Every year, loads of documents are generated by every organization. Keeping track of such documents as well as maintaining and storing them is lengthy and cumbersome, and is generally not automated. This file manager aims to enhance the information system in any university and facilitate the transfer and sharing of information, removing any barriers from between. It would help the faculty assign tasks to the students with a click of a button, in an organized fashion, and keep a track of all such assignments. The students can view all the assignments, and upload the solutions on the web. It also aims to improve sharing of documents and file between the faculties from different department, and sharing of important circulars and notices within the university.

CHAPTER 1 : INTODUCTION

1. General Introduction

1.1. What is File Management?

File management involves the migration of files onto an electronic storage medium to allow easy access and sharing. It also involves the ability to easily retrieve the information according to the agreed access right of the file. Thus files are held in a single repository that simplifies managing and retrieving the files when necessary. It is usually also capable of keeping track of the various documents uploaded by any users. Therefore, any file manager is responsible for controlling and organizing documents throughout any organization. Since different organizations have diverse needs, these systems are often custom designed.

1.2. Introduction to FILE-PORT

FILE-PORT is a web based file manager, customized to suit JUIT. The system provides a user friendly interface to allow sharing of files between the faculty and the students. The main aim behind the system is to make the system of document sharing and assignment submission paperless, and shift to electronic form of file system. Each year, load of paper based documents are generated in the university in the form of assignments, lab files, reports etc. Management of such huge amount of paper based files and keeping a track of them is a cumbersome job. Also allocating storage space, issues of lost files, damage to paper based files adds to the problem.

This system provides a solution to this situation by making the file system paperless and automated. The system provides separate functionalities to the faculty and the students. It automatically detects the subjects allocated to the teachers and allows them to upload and share documents according to the respective subjects. It allows them to send alerts and messages in form of emails to other faculty members or the students.

The students can login to the system and check the files or the assignments uploaded by the faculty. These will be displayed according to the subjects that the student has, which will be recognized automatically. The student can then upload the assignment solution on the system.

Since the system is web based, it can be accessed from any location. Also it encourages the users to save paper by shifting to electronic medium of documents and files. Apart from text based, other forms of files can also be shared on the system, like executable codes, images etc. It solves the problem of storing large amount of paper based documents and keeping a track of them. Thus, if inculcated in the JUIT system, it can help to improve the process of file sharing and encourage the users to go paperless.

2. Motivation

The main motivation behind this project is to encourage the users to shift to an electronic form of file system, thereby using electronic form of files and documents rather than paper based documents.

There are a number of problems with traditional approaches to the paper based documents, especially assignments sharing, and management of student assessment, especially in a distance education setting. Due to the benefits it can provide, there has been widespread interest in the use of a web based system for online submission of assignments and other documents. In a study focusing on the use of online assignment management in a programming course Price and Peter (1997) found that students reported a decrease in assignment turnaround time from 2 weeks down to 5 to 7 days. Chi-Sang et al (1999) report a reduction in the assignment submission cycle from 2-3 weeks to 1-2 hours.

When supported appropriately with resources, experience, and minimal automation it has been found that marking online assignments takes 20-30% less time than marking traditional assignments (Jamieson and Jones, 1997). Summons et al (1997) report tutors marking being reduced by an average of 67% by using a semi-automated marking process. The greater the automation used the greater the reduction in marking time.

Appropriate reporting mechanisms combined with online assignment management increase the transparency of the marking process by allowing students to view the progress of their assignment [3]

3. Benefits of using FILE-PORT

There are a large number of advantages of using a web based management system like FILE-PORT over having files spread over various storage location or using conventional manual use and organization of paper documents.

- **Reduced Storage**
  The cost of commercial property and the need to store documentation means that paper based document storage competes with people for space within an organization. Using FILE-PORT can greatly reduce the amount of prime storage space required by paper, as electronic form of files are used.
- **Flexible and faster search and retrieval**
  An effective and efficient search functionality is provided by FILE-PORT, which will reduce the amount of time for file search and retrieval.
- **Documents and Assignment handling all in one place, accessible from anywhere**
  By using FILE-PORT, all aspects of dealing with assignments and documents are maintained in a single location, accessible from anywhere with an internet connection.
- **Reduced paper handling: less to photocopy and print**
  An electronic system reduces the amount of paper handling required. There is no need to photocopy documents nor spend time and money in printing them.
- **Email notifications of key events, easy communication**
  Any system including email notifications of key events: sharing of important notices and circulars, submissions etc. will help in making the communication

between all those involved is easy, lessening delays in the whole process. Email notifications is one of the key features of FILE-PORT.

- **Submission Tracking**

  The logging of every action is integrated into the system so it is easy to track who submitted what and when, and all documents uploaded to the system are stored for easy reference and retrieval.

4. Aim of the Software

The aim of the FILE-PORT is to manage files electronically and improve the flow of information in any organization. This software is built keeping in mind the scenario in Jaypee University of Information Technology and to bring a shift from conventional paper based document sharing system to web based document management. It will help the faculty upload documents to be shared with other faculty or with the students. All these documents can be viewed by the intended viewer and can be further shared. The main aim is to improve the assignment management system by allowing the faculty to upload the assignment on the system, and the students to view the assignments and upload the solutions. Also it will encourage the users to go paperless, thus saving huge amount of paper. Further extensions can include online marking of those assignments that will promote transparency in the University. Classification of the documents will help in easy search and retrieval.

CHAPTER 2 : PROJECT DESIGN

This chapter gives the overall description of FILE-PORT i.e. it’s functionalities and how the user interacts with the software. This is described through use case diagrams and activity diagrams.

1. Functionalities Description

This section describes all the functional requirements that the system must fulfill, and it’s detailed description.

1.1. User Registration

The user should register with the system to gain access of the system. Registration will involve asking the user to enter his personal details, like user ID, email ID, which will be required to send email notifications and the password. The user is also required to enter the type of login, i.e. Faculty/ Students. Different types of details are required for each types of users.

On entering the details the system should check if the user ID is already registered or not, and if it is not, the details will be stored in the system’s database.

1.2. User Login

The system shall identify the user ID and the password, and validate the details to give the user access to the system. The precondition includes that the user is registered with the system.

1.3. Manage profile

The system shall allow the users to edit and update their details entered, or add additional details which will be saved by the system. The user can also delete the profile, which will result in deletion of all the files uploaded by him and his personal data stored with the system.

These user details will help the system to identity the Subjects allocated to each faculty, and will help in document classification. The email ID entered by the user will help the system in sending email notifications to the users.

1.4. Upload File

This functionality is available only to the Faculty, and not the students. After logging into the system, the user can upload a limited number of files on the system. This will include selecting the file and entering the type of document (General document/ Assignment). The system will validate these details, and if found correct, would upload the file onto the system.

This functionality will have a special case of uploading an assignment, which would result in opening of a submission link where the students can submit the solutions.

There will be restrictions on the size of the file being uploaded.

1.5. Access permissions

Any document uploaded by the faculty on FILE-PORT will be visible only to those students who are registered with the same subject.

Apart from that, a faculty can upload a document to be shared only within the faculty of the same department.

1.6. Set Expiration Date

The uploader can set the expiration date for documents, especially the assignments. It can be set in terms of number of days, after which the document will be automatically deleted from the system.

In case of assignment, the system will also close the submission link for that document.

1.7. File Classification

The file classification will be done on the basis of the subject name. While uploading the document, the faculty is required to select the subject to which that document is related to. There is also an option of ‘General’ for general documents that do not fit into any other category.

The pre-conditions includes that the user must be logged into the system and should have selected the file to be uploaded.

For creation of folders for classification, the faculty, at the start of each semester, will have to select the time table which the system will parse and extract the subject names and codes that are allocated to that faculty.

1.8. Search file

This functionality allows user to search any file uploaded on the system. The user will be required to enter a keyword against which the search has to be performed.

The files having name matching to the search keyword will be displayed as a result.

1.9. View/Download file

The user can select the file name and download the file to save it to the computer.

The pre-conditions includes that the file must be uploaded on the system and the user must be logged into the system.

1.10. File Deletion

When a file is no longer required to be uploaded on the system, the user can select and delete the file from the system after logging in.

1.11. Email Notifications

The system will provide email notifications for various events. The notification would be sent to the email id registered with the system.

Specifically, the faculty should be able send an email to another faculty member by entering the email address. Also the faculty can send out a reminder to a specific batch regarding an assignment. The faculty members can also send remarks to specific students of a batch.

1.12. Assignment Solutions Submission

This functionality is available to the students, which will allow them to upload the solutions of the selected assignment within the deadline. The system will show the assignments which are pending and which have been submitted.

2. Use Case Diagram

The FILE-PORT Use Case shows a detailed view of the system and how the actors would interact with each other and with the system. The explanation for each use case is then provided below Figure 1 and helps to understand who the actors are, as well as giving the description of each use case along with its pre and post conditions that should be satisfied once the use case is implemented in the software.

The system provides two types of Login:

1. Faculty Login: The faculty can login using the Faculty Login Option, under which he would be required to enter the faculty Id and the password. The faculty needs to register with the system first before he can log into it. On logging in, the faculty can manage his profile and enter the required details. The faculty has an upload document functionality under which he can upload any document he wants to share on the system. While uploading, the faculty is required to enter the various details like file classification, access permissions etc. A special case of file upload is assignment upload where the faculty can upload an assignment for the students and this would open a submission link for uploading the assignments in the student’s page. The faculty can set an expiration date for the assignment, and on reaching the deadline, the document will be deleted from the system automatically and the submission link

will be closed. The faculty also has an option of explicitly deleting a document when it is no longer required. The faculty can search for a file, and if found, can download or view the file. There is also an option of email notifications, send to the email Id registered with the system. The email can be sent on various events to the user or other users.

2. Student Login: As compared to the faculty, the students has limited number of functionalities available on the system. The student needs to register with the system to avail the functionalities on it. After registration, the student can log into the system. The student can view all the documents that are uploaded by other faculty members, and download them. The documents are classified according to the subjects. The student can also view the assignments uploaded for him and can submit the solutions on the system before the deadline. The student also has a search functionality which allows him to search the system for any documents required using keywords.

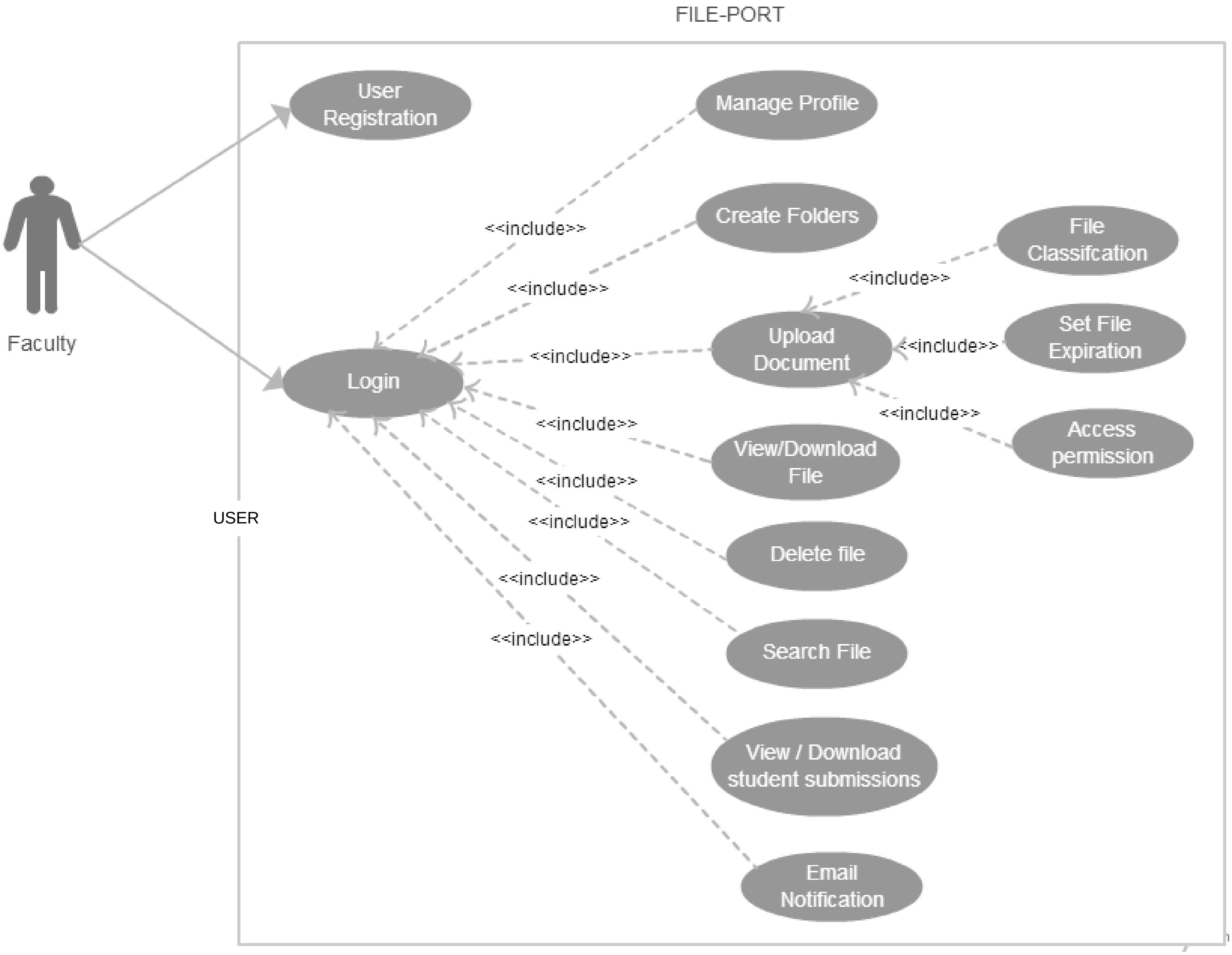


Figure 1: FILE-PORT Use Case Diagram for Faculty User

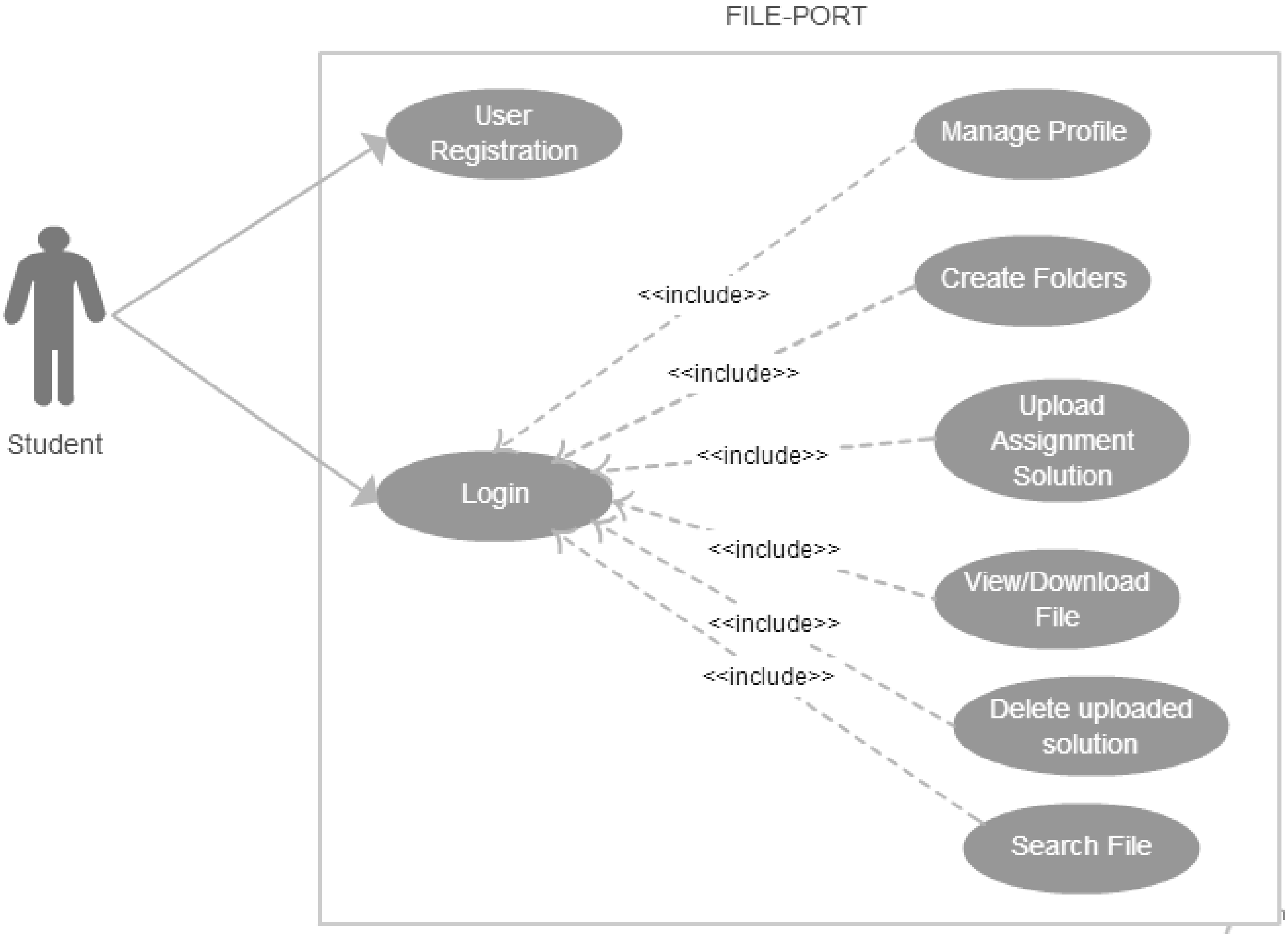


Figure 2: FILE-PORT Use Case Diagram for Student User

Below is the explanation of the different Use Cases in the FILE-PORT Use Case Diagram and the Actors associated with each Use Case. This would give any user a better understanding of the working of the system.

1. USE CASE 1: **Faculty Registration**
   Description: The faculty is required to enter the details like the user ID, First Name, Last Name, Faculty Code, Email ID, Department and the password to register with the system.
   Primary Actor: Faculty
   Pre-Condition: The user has opened the system on the web browser.
   Post- Condition: The user registers with the system

2. USE CASE 2: **Student Registration**
   Description: The student is required to enter the details like the user ID, First Name, Last Name, Email ID, Department, Course Type (B.Tech/M.tech), Batch, Semester and the password to register with the system.
   Primary Actor: Student
   Pre-Condition: The user has opened the system on the web browser.
   Post- Condition: The user registers with the system

3. USE CASE 3: **User Login**
   Description: This details the login process of the user- Faculty/Student to the system to access it. The user will login to the system using a username and password, and also select the Login type.
   Primary Actor: Faculty, Student
   Pre-Condition: The user is already assigned a username and password to access the system, i.e. the user has already registered.
   Post- Condition: The user logs in to obtain access to the system.

4. USE CASE 4: **Manage Profile**
   Description:

(i) The user can update his profile on the system, which contains his details.

(ii) This use case also allows the user to delete his profile from the system.

Primary Actor: Student, Faculty

Pre-Condition: User has logged into the system.

Post- Condition: User profile will be updated / deleted.

5. USE CASE 5: **Create Folders for Faculty**

   Description: The system will identify the subject taught be the faculty from the time table from that semester, and register those courses. This identification will be done with the help of the faculty code entered by the user.

   Primary Actor: Faculty

   Pre-Condition: User has logged into the system.

   Post- Condition: The subject taught be the Faculty are successfully registered and folders are created.

6. USE CASE 6: **Student Folders for Student**

   Description: The system will identify the subject being taught to the student user from the time table from that semester, and register those courses. This identification will be done with the help of the batch and the semester entered by the user.

   Primary Actor: Student

   Pre-Condition: User has logged into the system.

   Post- Condition: The subject being taught to the Student are successfully registered and folders are created.

7. USE CASE 7: **Upload File**

   Description: This use case will help the user to browse files from the computer and upload it onto the system. The file can be any general document to be shared or an assignment.

Primary Actor: Faculty

Pre-Condition: User has logged into the system.

Post- Condition: The user successfully uploads the document.

8. USE CASE 8: **File Classification**

   Description: This allows the uploader to classify files according to the subject it is related to. While uploading, the Faculty user is required to select the folder, according to which the file will be classified.

   Primary Actor: Faculty

   Pre-Condition: User has logged into the system and has selected the file to upload.

   Post- Condition: The user is successfully able to set file classification.

9. USE CASE 9: **Set Expiration Time**

   Description: This helps the user to set the expiration time for the document, especially the assignment being uploaded, after which the document will be deleted automatically from the system.

   Primary Actor: Faculty

   Pre-Condition: User has logged into the system and selected file for uploading.

   Post- Condition: The user successfully sets the expiration time for the file.

10. USE CASE 10: **Access permissions**

    Description: This will help the user to set the access permission for the uploaded files in the system. The uploader can select to whom the file will be visible.

    Primary Actor: Faculty

    Pre-Condition: User has logged into the system and has selected the file to upload.

    Post- Condition: The user is successfully able to set permission for the file.

11. USE CASE 11: **Download/View File**

    Description: This use case helps user to download the selected file to their computer.

    Primary Actor: Faculty, Student

    Pre-Condition: User has logged into the system and the file selected is uploaded to the system.

    Post- Condition: The file is successfully downloaded to the local system.

12. USE CASE 12: **Delete File**

    Description: This helps the user to delete files uploaded on the system by that user.

    Primary Actor: Faculty

    Pre-Condition: User has logged into the system and uploaded a file.

    Post- Condition: The file is successfully deleted from the system.

13. USE CASE 13: **File Search**

    Description: This helps the user to search the files uploaded on the system. The search is done according to the keyword entered by the user.

    Primary Actor: Faculty, Student

    Pre-Condition: User has logged into the system.

    Post- Condition: The user is displayed the result of the search.

14. USE CASE 14: **View / Download Student Submission**

    Description: The faculty can download the submissions by the students in response to an assignment uploaded by the faculty.

    Primary Actor: Faculty

    Pre-Condition: User has logged into the system.

    Post- Condition: (i) The faculty user successfully downloads all the submissions by the students for an assignment.

    (ii) The original assignment and the students submissions are then deleted from the system.

15. USE CASE 15: **Email Notification**
    Description: User gets email notification on various events from the system automatically. The user can also set events explicitly to send notification to other users.
    Primary Actor: Faculty
    Pre-Condition: User has logged into the system.
    Post- Condition: The user sends or receives email notification.

16. USE CASE 16: **Upload Assignment Solutions**
    Description: Allows the student submit the solutions to an uploaded assignment
    Primary Actor: Student
    Pre-Condition: User has logged into the system and there is an open assignment.
    Post- Condition: The user successfully uploads the solution to an assignment.

17. USE CASE 17: **Delete Uploaded Assignment Solutions**
    Description: Allows the student user to delete any previously uploaded solution to an assignment.
    Primary Actor: Student
    Pre-Condition: User has logged into the system and there is an open assignment.
    Post- Condition: The user successfully deletes the uploaded solution to an assignment.

3. Activity Diagram

This section discusses the Activity diagram and describes the flow of Activities in the System.

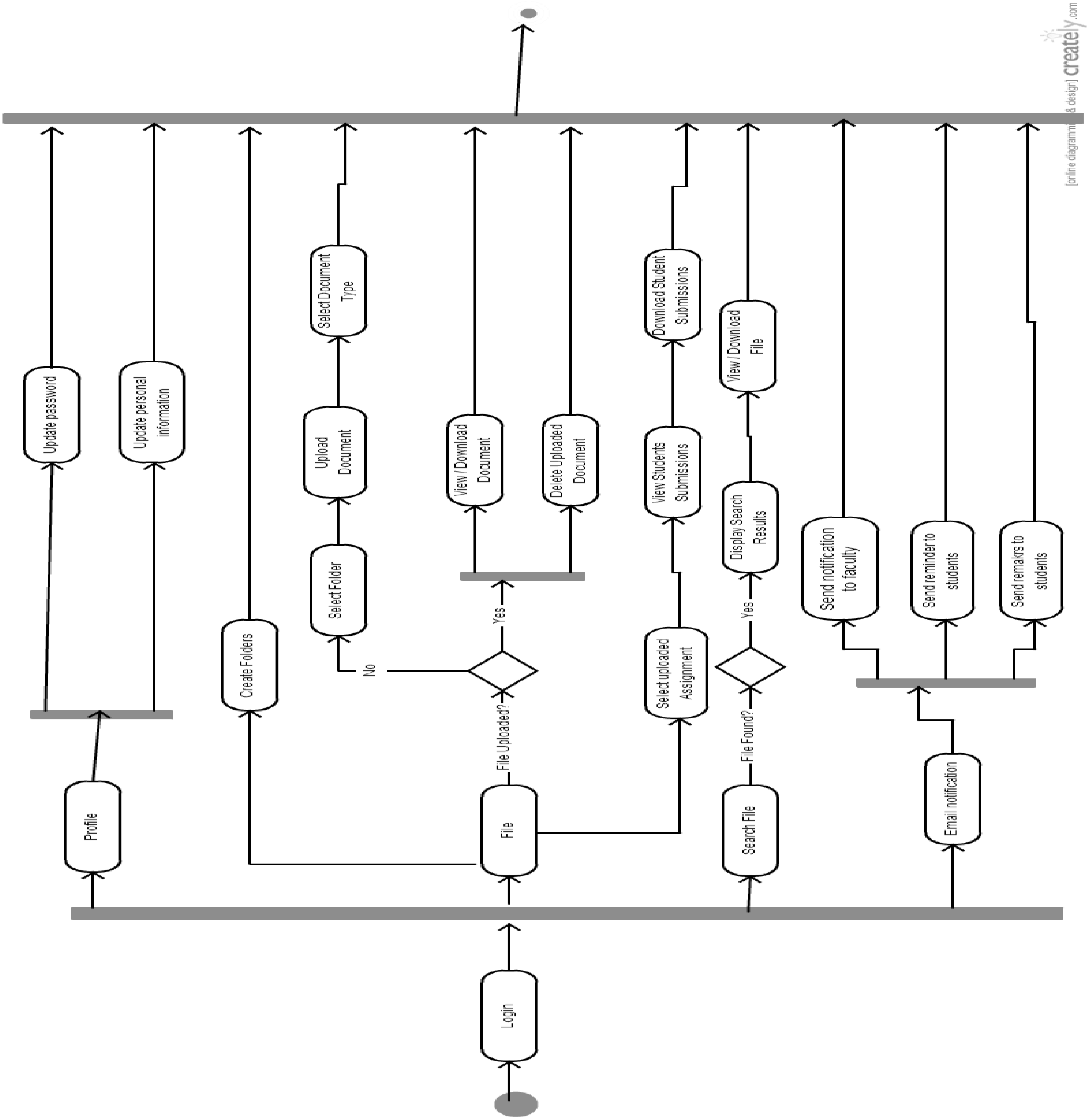


Figure 3: FILE-PORT Activity Diagram for Faculty user

1. FACULTY USER: The flow begins when the user starts the web application. A login screen appears in the web browser, prompting the users to enter their user ID and password. Users have a defined set of options for navigation once they log into the system:

   - **Profile:** This tab appears for the users and allows them to edit and update their personal information or password on their profiles or delete their profiles from the system.

   - **File:** The faculty user first of all has to create folders, which will be done using the Faculty code for that user. The subject taught be the faculty for that semester will be extracted and registered with the system automatically.
     If a file is already uploaded onto the system, the user has various options regarding it. The user can view/download that file onto the local system. The user can also delete any previously uploaded file from the system.
     If the user wants to upload any new file, the first thing required is to select the folder to which he wants to upload the file. Next he has to select the file and enter it's type (General Document / Assignment). This will successfully upload the document on the system.
     The user can also view the solutions uploaded by the students in response to an assignment. For this, the user needs to click on the assignment to view the list of students who have uploaded the solution and click on download to download the solutions. Downloading the solutions will result in deletion of the original document and the student submissions from the system.

   - **Search File:** The user will enter the search keyword. If the file is found, the results will be displayed. The user can then download it, else the system will display no file found message.

- **Email Notification:** The system implicitly sends email notifications to the user on various events, like successful registration with the system. The user can also send email notifications to other faculty users explicitly by entering the email ID. The user also has the option of sending reminder for a specific assignment to the students. Apart from all this, the Faculty user can send remarks to specific student through the system.

- **Logout:** This will logout the user form the system and end the running session. To use the functionalities of the system again, the user will have to login again.

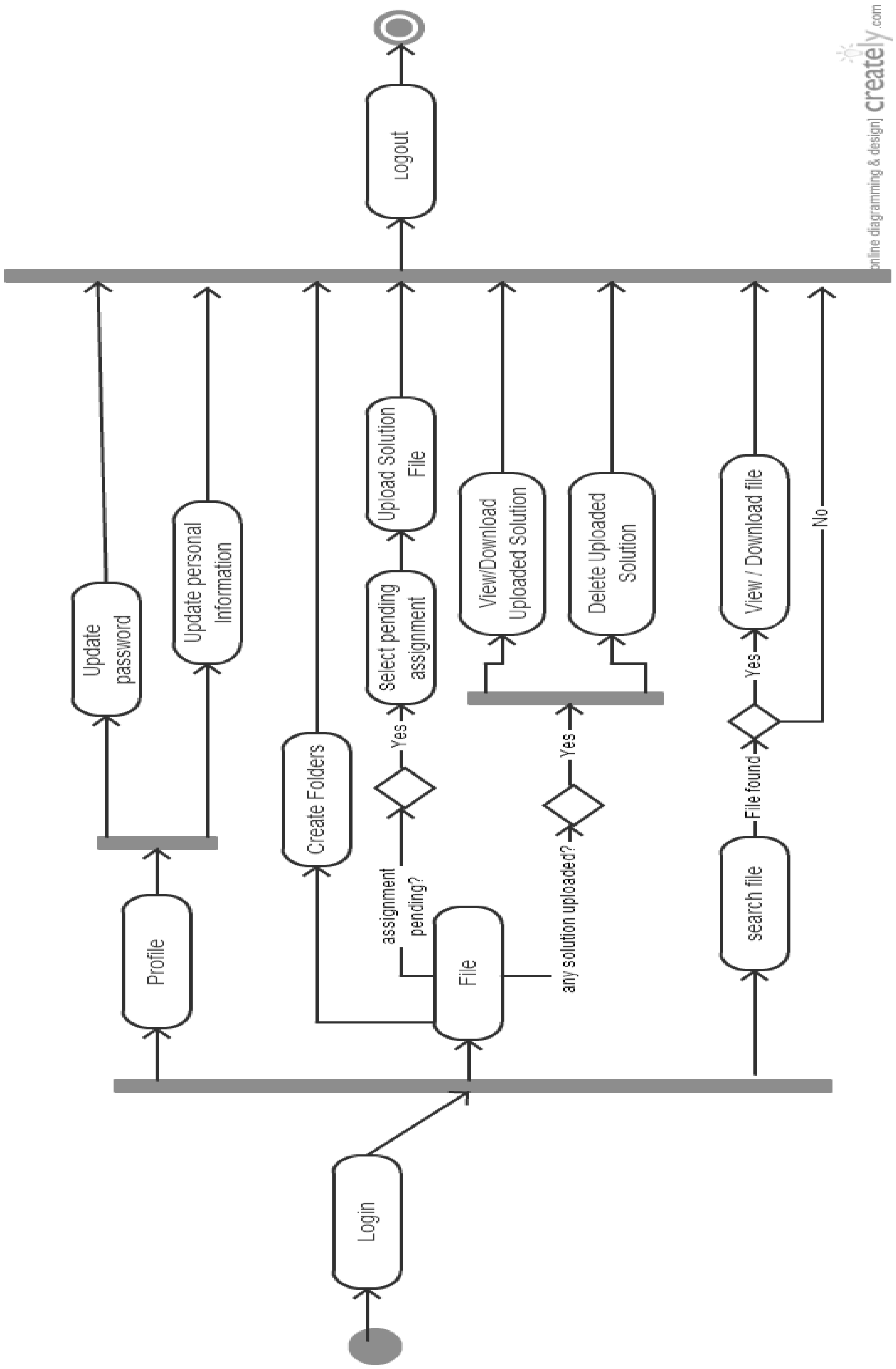


Figure 4: FILE-PORT Activity Diagram for Student user

2. STUDENT USER: The flow begins when the user starts the web application. A login screen appears in the web browser, prompting the users to enter their user ID and password. Users have a defined set of options for navigation once they log into the system:

    - **Profile:** This tab appears for the users and allows them to edit and update their personal information or password on their profiles or delete their profiles from the system.

    - **File:** To start with, the student is required to create folders, i.e. registers the subjects being taught to the Student user in that semester with the system. The system will use the batch and the semester of the student to automatically identify the subjects from the time table and register them with the system.
      The student can view the list of assignments uploaded by the faculty, which are categorized as pending/submitted. If any assignment is pending, the user can upload the solution to it by selecting it.
      If the user has uploaded any assignment solution, he can view a list of it. The user can select the file to view/download it. Alternatively, the user can also delete any uploaded solution if he wants to.

    - **Search File:** The user will enter the search keyword. If the file is found, the user can download it, else the system will display no file found message.

    - **Logout:** This will logout the user form the system and end the running session. To use the functionalities of the system again, the user will have to login again.

4. Product Perspective

The FILE-PORT is a web-based system and can be accessed using a web browser like: Internet Explorer8.0 and above, Mozilla Firefox, and Google Chrome.

5. Software Interface

This section list the requirements of the system that is the user needs to run the FILE-PORT efficiently. It lists the Operating System required that will allow the system to run effectively, the interface that is be required to run the application, the driver required for running the web applications.

1. Operating System: Windows (Vista/Windows 7 or above) or MAC OS
2. Web Brower: Internet Explorer (9.0 and above), Mozilla Firefox (3.0 and above), or Google Chrome.
3. Drivers: Java Runtime Environment

5. Assumptions and dependencies

The assumptions and dependencies are as follows:

1. The system is completely dependent on the availability of an Internet connection.
2. We assume that the organization has enough server space to store all the files.
3. We assume that users of the system adhere to the system's minimum software requirements.
4. It is assumed that a faculty user has a 3 digit user id, while a student has 6 digit userid.

CHAPTER 3 : IMPLEMENTATION

This chapter includes the detailed design, procedure and technologies used to build FILE-PORT. The system's mock screen shots of currently implemented functionalities are also shown.

1. Technologies Used

The following technologies has been used for the development of the system:

1.1. Front-End Technologies

Being a web application, the front end, i.e. the user interface is designed using HTML, CSS and JavaScript.

Tools like Adobe Photoshop CS5.1 will be used to enhance the look of the front end.

Web page designing will be done using Adobe Dreamweaver CS5.5

1.2. Back End Technologies

The language used for back end development of web applications is JAVA.

MySQL is used as database system.

Tool used for back end development is NetBeans IDE 7.2.

2. Database Structure

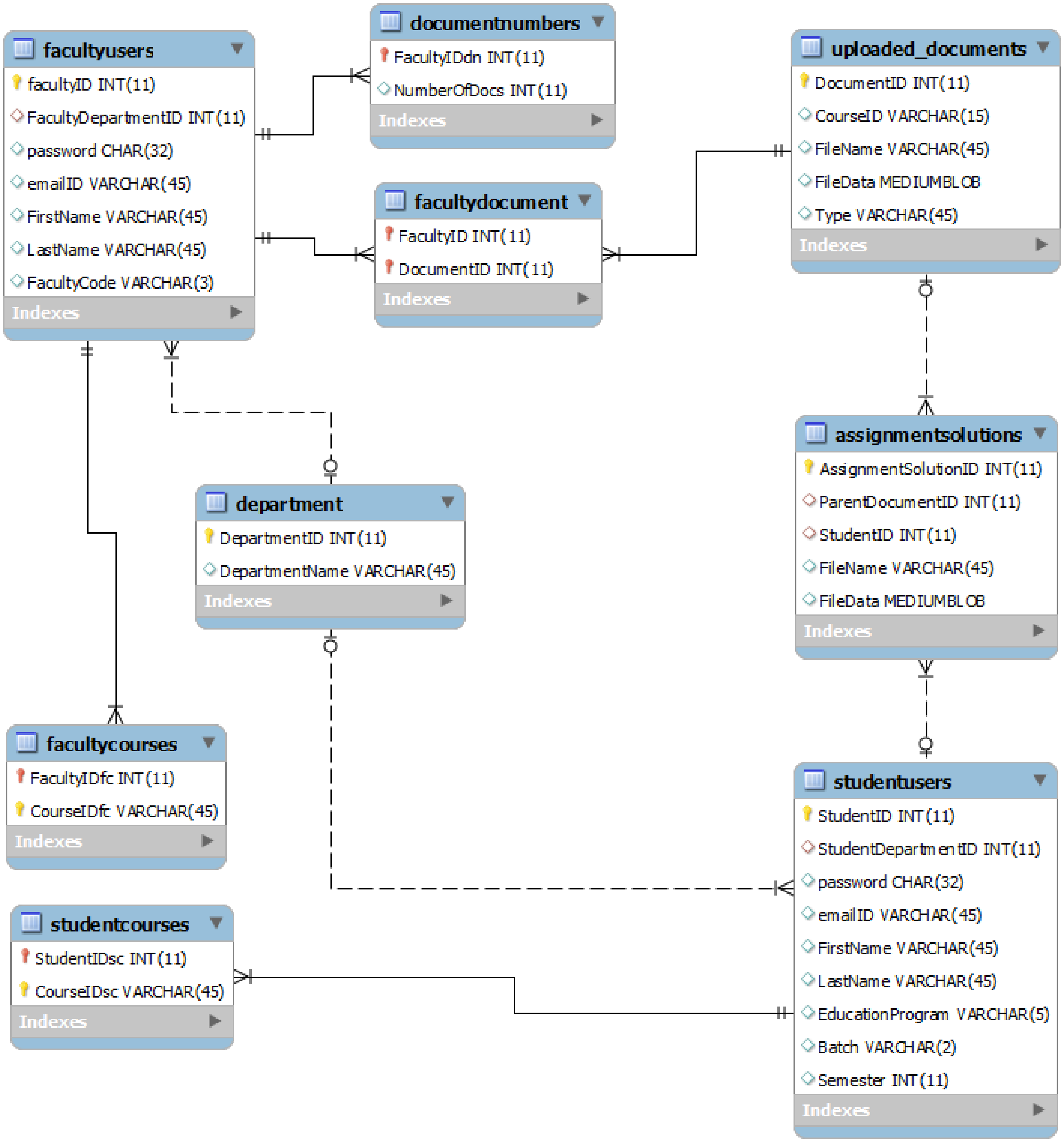


Figure 5: Database structure of FILE-PORT system

3. Time Table Parsing For Creating Classification Folders

The system uses the subjects to which the file is related to classify them. This is one of the key features of the system. The identification of these subjects is done automatically by the system, and the information is extracted from the time table for that semester. Thus, there is no need for a user to manually enter these details, and all the files are classified according to the subject, providing convenience to the users.

So before upload of any file, or before doing anything related to the documents/files in the system, it is necessary that these subjects are registered with the system and folders are created. If they are not, a message is prompted on the screen asking the user to register these subjects. This can be done with just a click of a button.

Given below is detailed description of this functionality for each user type:

- FACULTY USER: For a faculty user, the system identifies what all subjects he is teaching in that semester. For this, the system using the faculty code (example- SAS for Mrs. Sanjana Singh). Then the system parses the time table for that semester and identifies all the subject taken by that Faculty, using that Subject code. Once the whole time table is scanned, the subjects found are registered by entering the details in the database and displayed on the screen.
- STUDENT USER: For a student user, the system identifies all the subjects being taught to that student in that semester. For this, the student's batch and the semester is used. The system reads the time table and finds out the subjects using the batch and the semester. Once the whole excel sheet is read, the subjects found are registered with the database system and displayed on the screen.

Creation of such folders and registration of subjects creates a common link between the Faculty and the Students, identifying the subject the user is teaching and the subject being taught respectively. These common subjects will later be used to identify which all documents are to be displayed to any student.

3.1. Flow Diagram

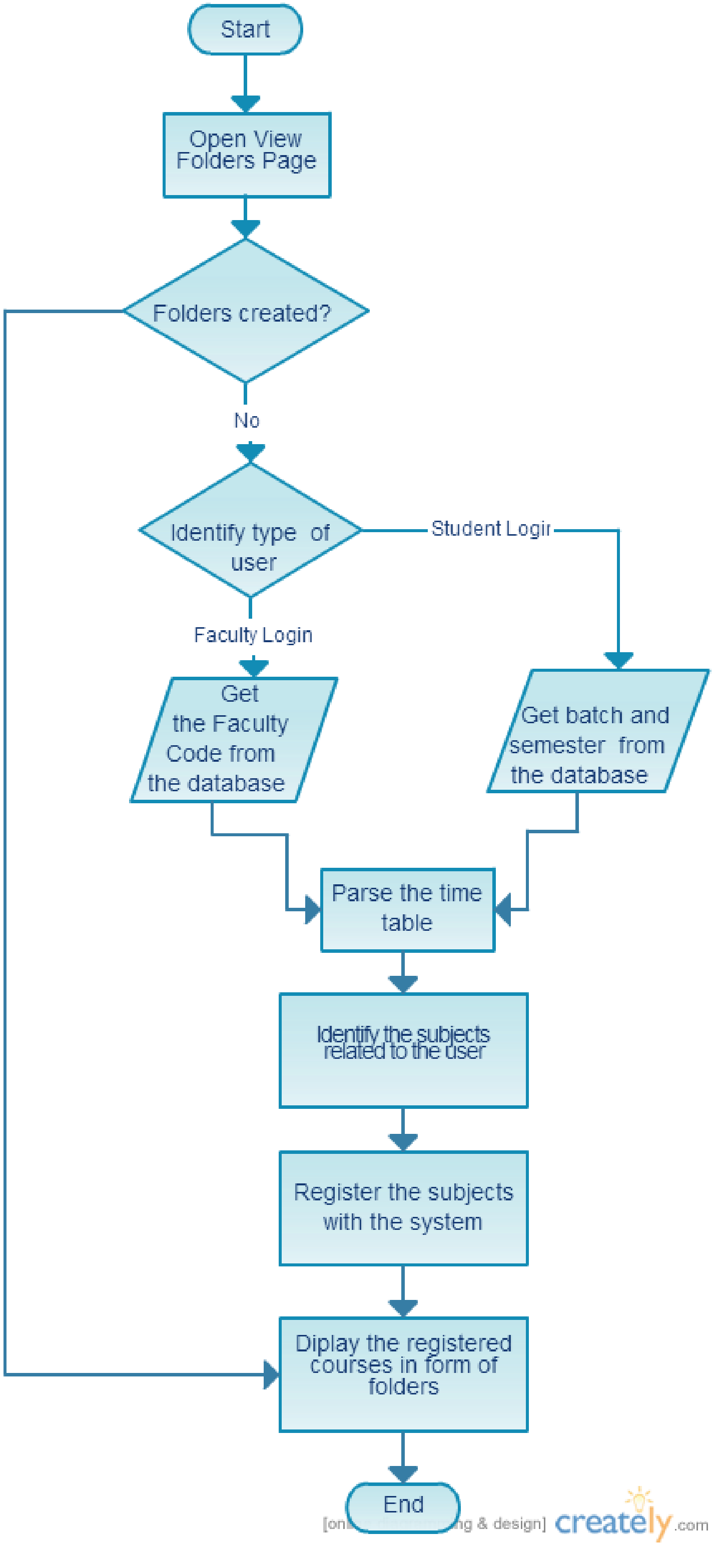
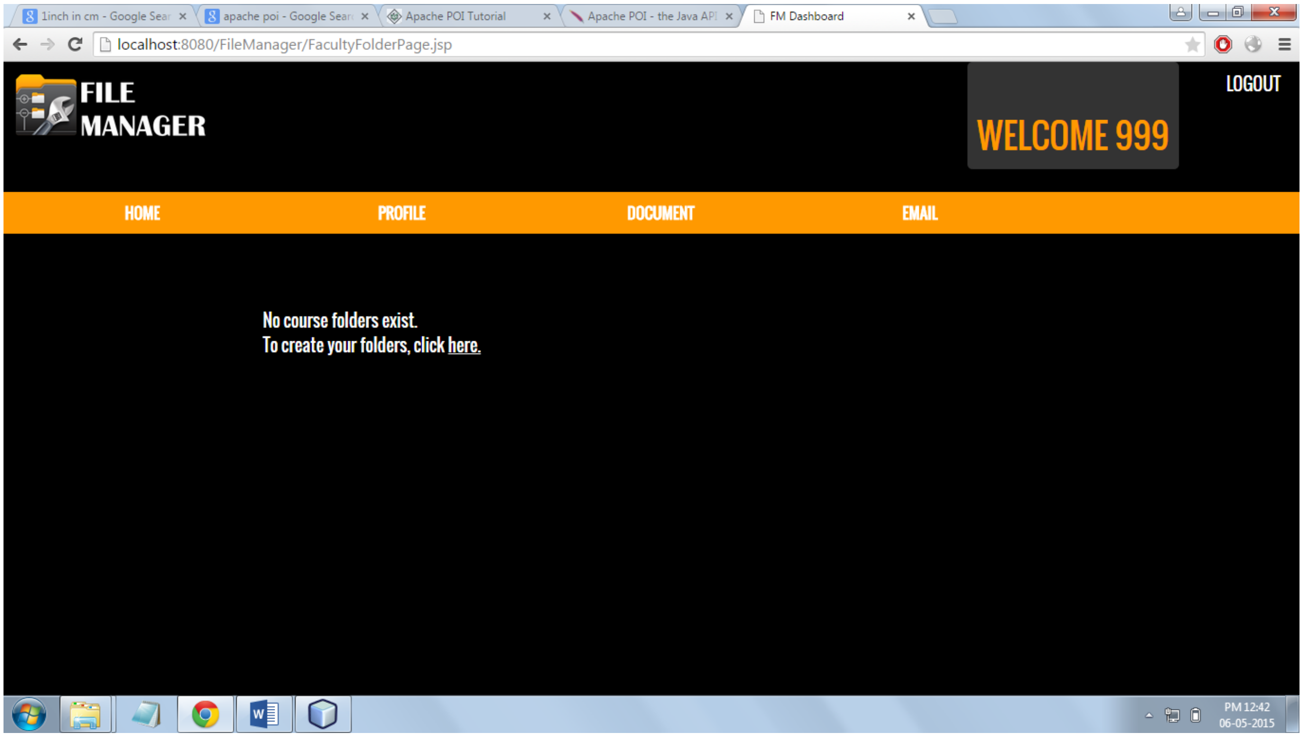

Start
Open View Folders Page
Folders created?
No
Identify type of user
Student Logir
Faculty Login
Get the Faculty Code from the database
Get batch and semester from the database
Parse the time table
Identify the subjects related to the user
Register the subjects with the system
Diplay the registered courses in form of folders
End
[onl ng & design] creately.com

Figure 6: Flow chart for time table parsing for creating classification folders

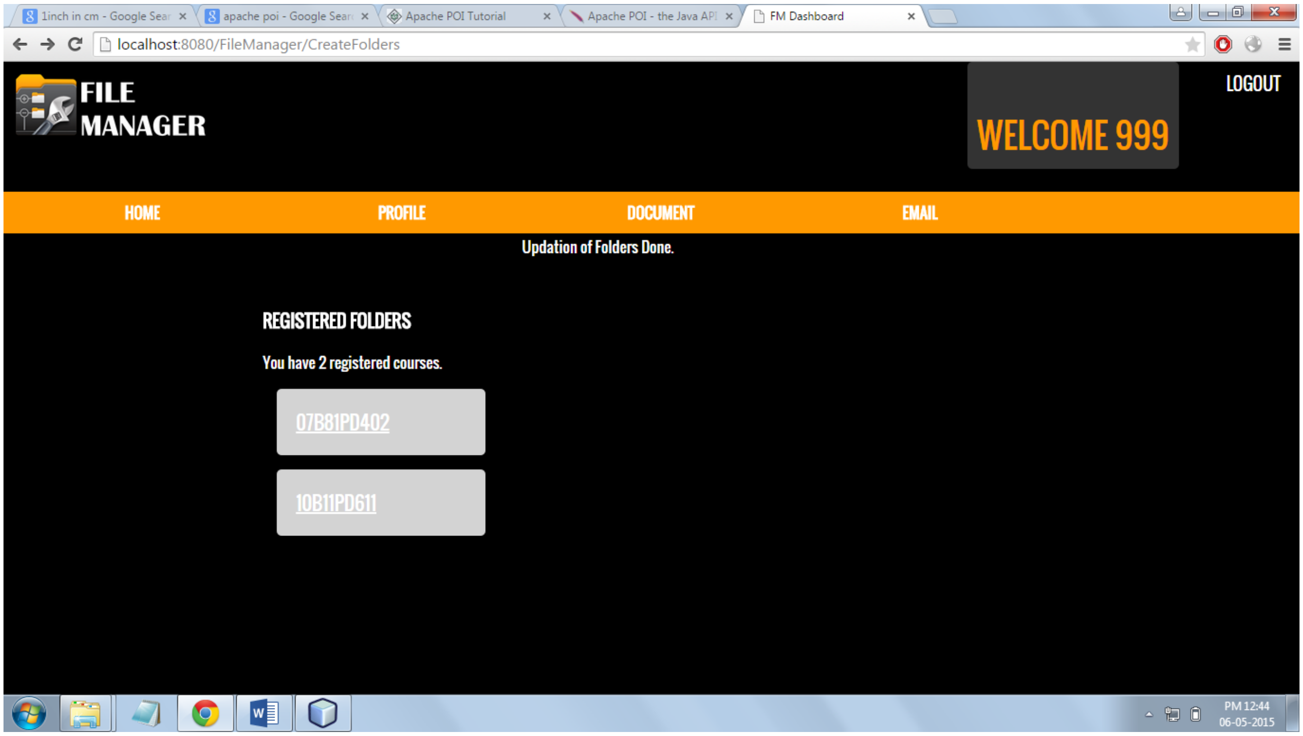
3.2. User Interface



No course folders exist.
To create your folders, click here.

Figure 7: UI for View Folders when no subject is registered

When no subjects are registered, the system displays message as shown above. On clickng the hyperlink, the user makes the system read the time table and scan and identify all the subjects related to the user.



Updation of Folders Done.

REGISTERED FOLDERS

You have 2 registered courses.

Figure 8: UI for View Folders when subjects are registered

When the user selects to register the courses, and the scanning of time table is done, all the subjects found are registered with the database and displayed on the screen. The UI is same for both the faculty and the student users.

3.3. Extracting information of subjects from the time table

In the above figures, we have extracted the subjects by entering ‘RSD’ as the faculty code. The following entries are read from the time table Excel file and this information is used by the system.

L-10B11PD611 C1 (RSD) TR3
L-07B81PD402 ALL (RSD) CR8

Figure 9: Subject registered in time table for Faculty Code ‘RSD’

Apart from just the lectures, the system also identifies the labs take by the Faculty, so that file related to it can also be uploaded on the system. Similarly for the students, both the lectures and the lab subjects are identified.

3.4. Apache POI API

Apache POI is a popular API that allows JAVA programmers to create, modify, and display MS Office files using Java programs. It is an open source library developed and distributed by Apache Software Foundation to design or modify Microsoft Office files using Java program. It contains classes and methods to decode the user input data or a file into MS Office documents.

The system uses this API to read the time table Excel file through MS Excel. HSSF is used to read the Excel file using Java (XLS).

4. Upload File

A Faculty user has the functionality of uploading a file to the system. The user has to select the folder on which the file has to be uploaded, thus a prerequisite includes that the folders are created. On selecting the folder, the user is forwarded to a page with a form. The user need to select the file that he has to upload, and select the type of document. The file uploaded can be of 2 types:

- General document: When the faculty user uploads such a document, it is simply displayed on the student's side when the student selects that folder.
- Assignment: When an assignment is uploaded, a link is opened on the student's end, where he can submit the solution to the assignment. Also, expiration time for such documents can also be set.

In the database, the file is stored as a MEDIUM BLOB. The file name stores is same as the file name of the original file. Since the datatype MEDIUM BLOB is used, there is a constraint on the size of the file that can be uploaded. Thus the user cannot upload files more than 16MB, and doing so would display an error message.

Also, there is a limit on the number of documents that any Faculty user can upload. So any user cannot upload more than 10 files.

4.1. Flow Chart

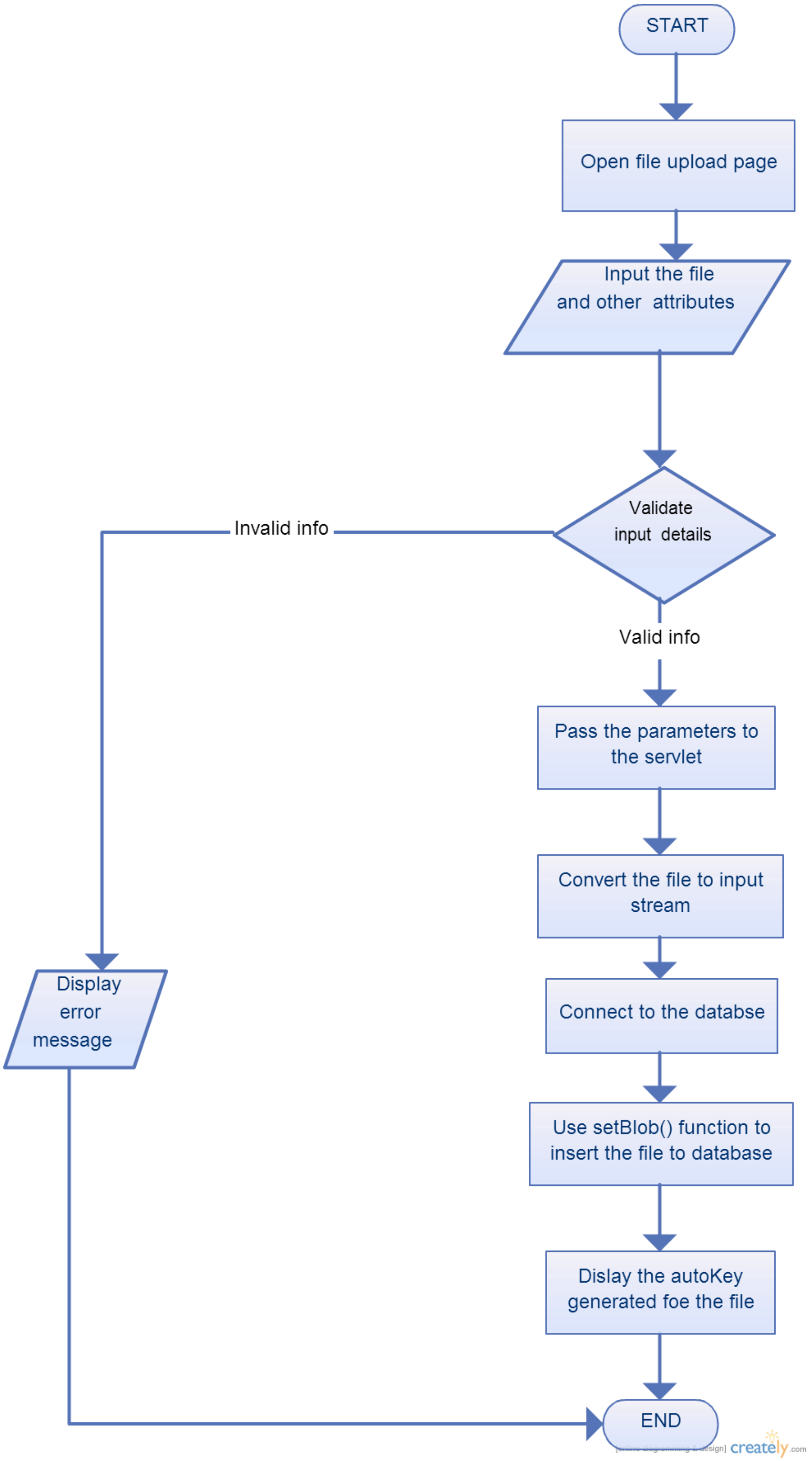
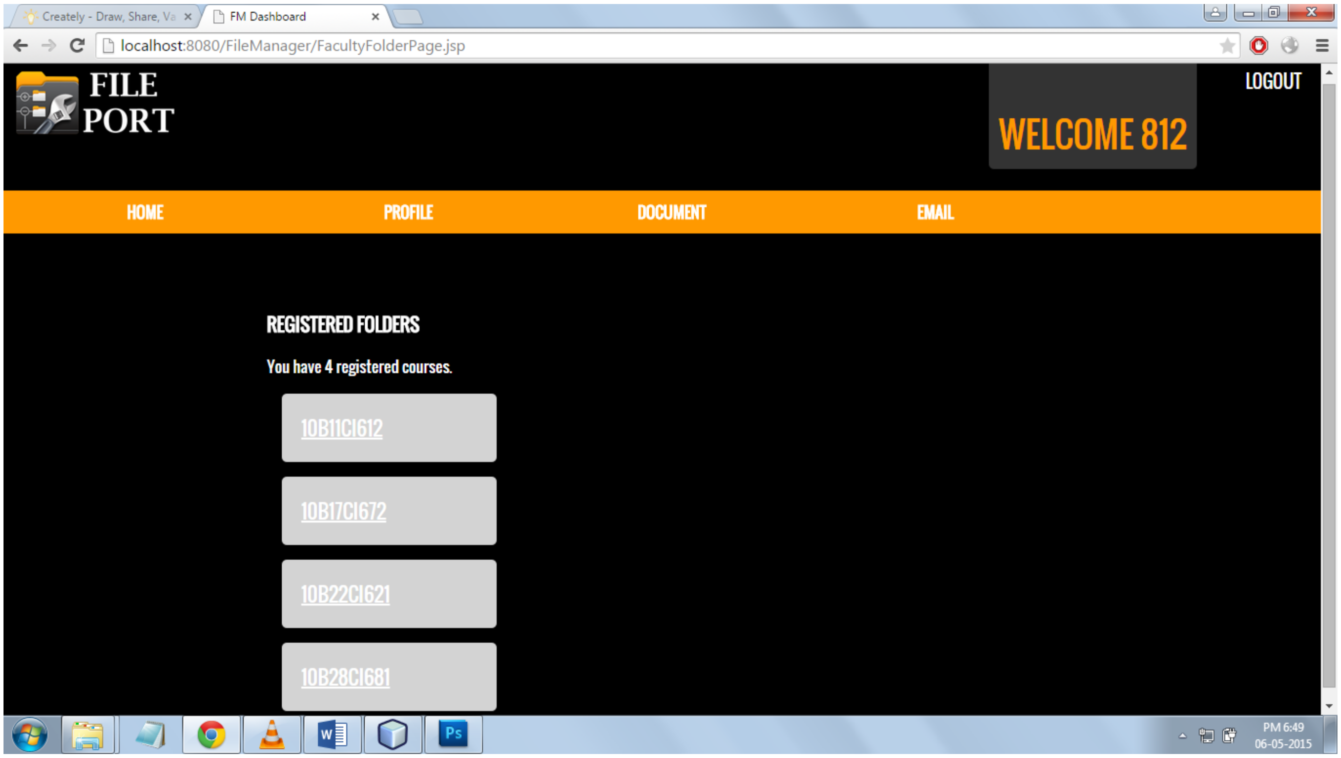

START
Open file upload page
Input the file and other attributes
Validate input details
Invalid info
Valid info
Pass the parameters to the servlet
Convert the file to input stream
Connect to the databse
Use setBlob() function to insert the file to database
Dislay the autoKey generated foe the file
Display error message
END
creately.com

Figure 10: File Upload Flow Chart

4.2. User Interface

Firstly, the user is required to select the folder in which the file is required to be uploaded. If the folders are not created, the user needs to click on the link and register the subjects with the system.

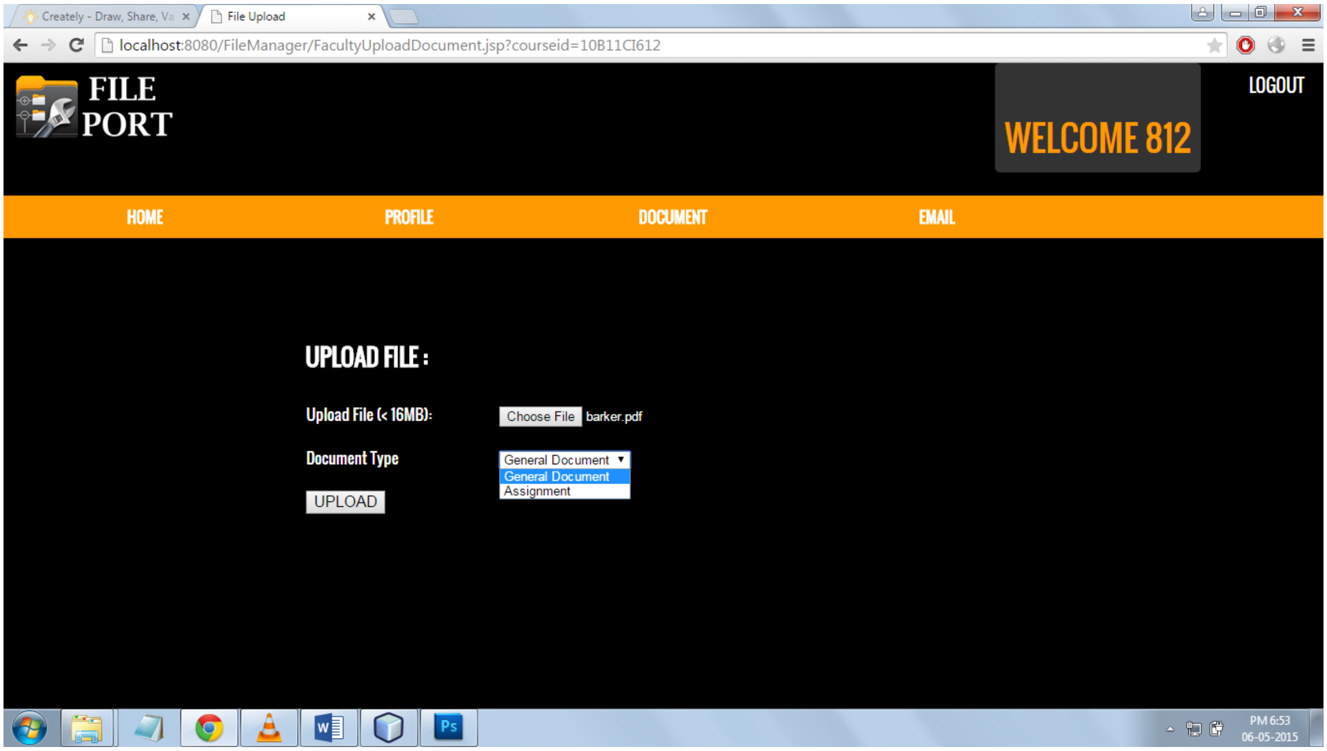
A Faculty will choose he is teaching and related to which the document is. When the student selects the same folder, the uploaded document will be displayed.



You have 4 registered courses.

Figure 11: UI for file upload – Displaying registered folders

On selecting the folder, the user is send to another page where he is required to select the file and enter the details. The file selected should not be more than 16MB. If it is, an error message will be displayed. The user is also asked to select the type of file (General document/Assignment)



localhost:8080/FileManager/FacultyUploadDocument.jsp?courseid=10B11CI612

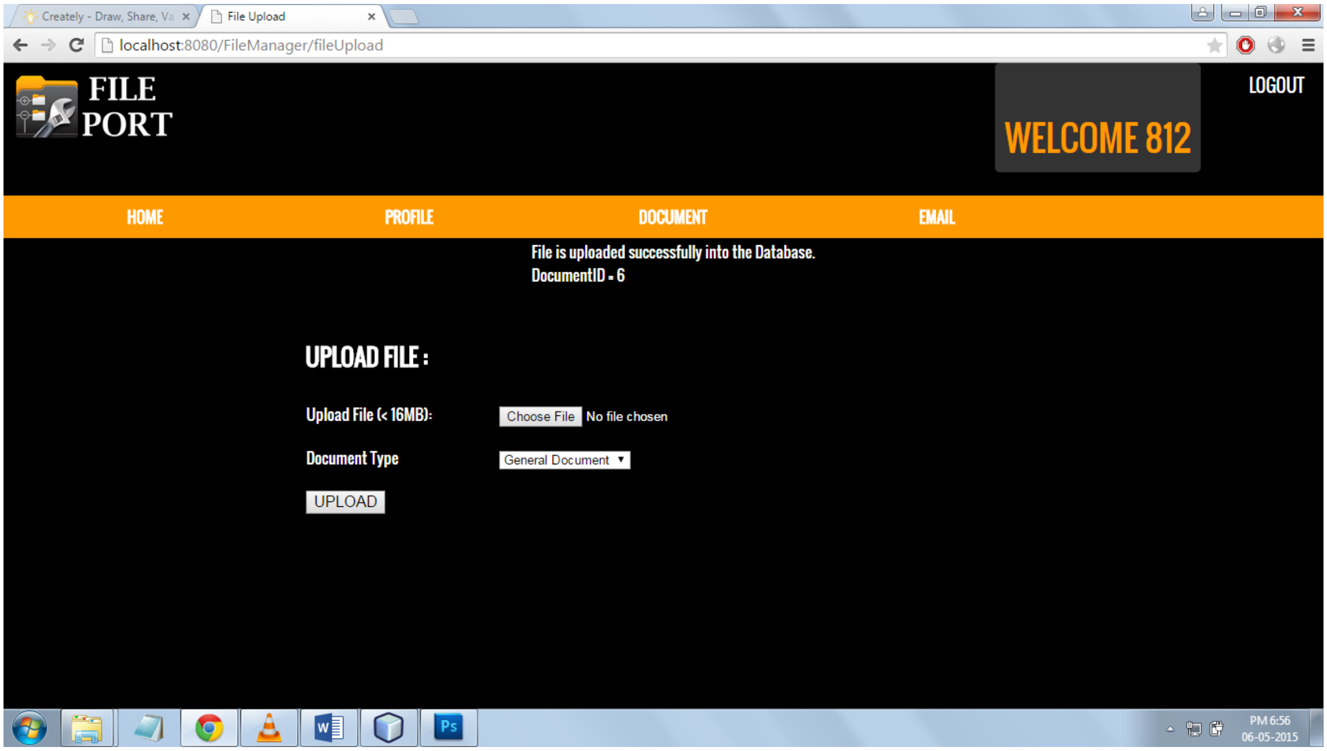
UPLOAD FILE :

Upload File (< 16MB): barker.pdf

Document Type: General Document

Figure 12: UI for file upload - Select file to upload

When the details are validated, the parameters are passes to a servlet where the uploading happens. When the upload is complete, the fileID generated for the fils displayed along with a success message, as seen in the figure below, where the document ID for the uploaded file is 6.



File is uploaded successfully into the Database.
DocumentID = 6

UPLOAD FILE :

Upload File (< 16MB): No file chosen

Document Type: General Document

Figure 13: UI for file upload – Message displayed after successful upload

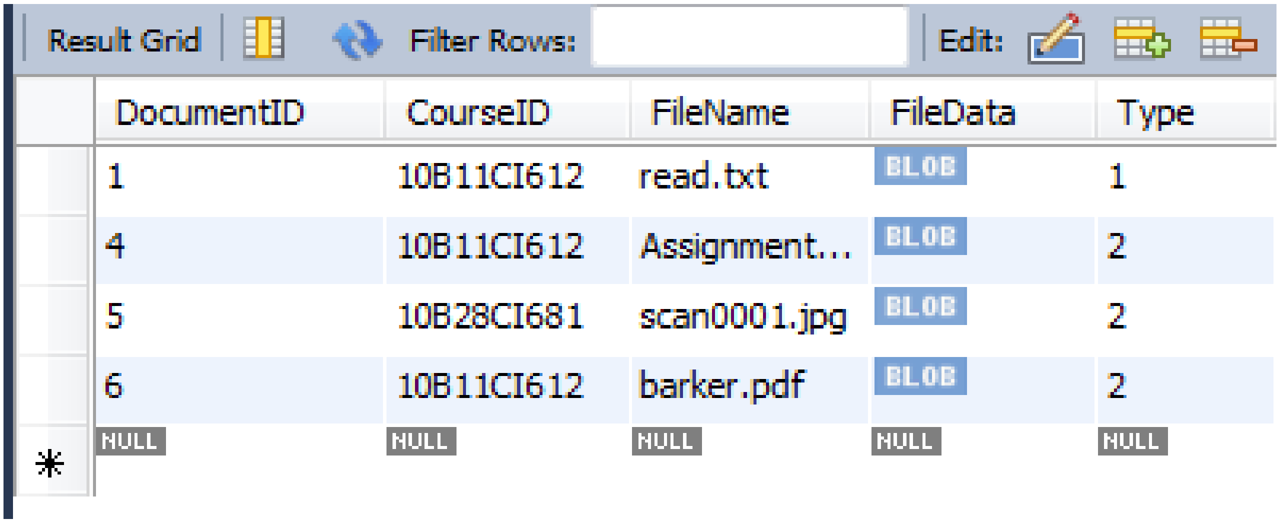
4.3. BLOB datatype

A blob is a collection of binary data stored as a single entity in a database management system. It can store variable amount of data. In MySQL, BLOB values are treated as binary strings (byte arrays).

There are 4 BLOB types and they differ only in the max length of the values they hold:

- TINYBLOB: 255 bytes
- BLOB: 65,535 bytes (64 KB)
- MEDIUMBLOB: 16,777,215 bytes (16 MB)
- LONGBLOB: 4 GB

In this system, we are storing files as MEDIUMBLOB. So the maximum file size can be 16MB.



DocumentID	CourseID	FileName	FileData	Type
1	10B11CI612	read.txt	BLOB	1
4	10B11CI612	Assignment...	BLOB	2
5	10B28CI681	scan0001.jpg	BLOB	2
6	10B11CI612	barker.pdf	BLOB	2
NULL	NULL	NULL	NULL	NULL

Figure 14: Values stored in databse after file upload

4.4. Database VS. File System

There were two option to store the files uploaded by the user – store it in the database directly or store the path of the file in the database, and file in the file system.

In this application, the files have been stored directly into the database as it more advantages over the file system storage, few of them given below:

1. **ACID compliant transactions**

   If there is a problem uploading the file, one can easily cancel the entire update and roll back to the previous state. If such an issue occurs with a file system, one has to manually search for the wrongly uploaded file and take the required steps.

2. **Backup**

   Backup routine becomes much simpler, as we need just backup a database rather than worrying about some arbitrary folders on the disk.

   Restore routine also becomes greatly simplified as you need just restore a database rather than a database as well as some bit of the file system.

3. When storing files in the database we can guarantee that the data is consistent.
4. Keeping track of changes/deleted items is difficult in file system. Tools for replication and change tracking for file systems aren’t available for huge file storages. Change tracking is available on database.
5. For transferring file, you need to open ftp and develop some tools to replicate them. Database access is standard it’s not dependent on ftp/http like tools.
6. While using database, your web server does not need any write permission on any folder. [1]

5. View or Download Document

This functionality has many aspects:

- A faculty user can view/download any file that he has previously uploaded.
- A faculty can view/download files that are uploaded by other faculty members, of the same department, i.e. any document that meets the access criteria for that user can be viewed/download.
- A student user can view/download documents uploaded by their faculty members.
- A students can view/download any assignment solution that he has previously uploaded on the system.

A list of files is displayed and the user can select multiple files that he wants to download. On clicking on download, the path where the files are saved is displayed.

5.1. Flow Chart

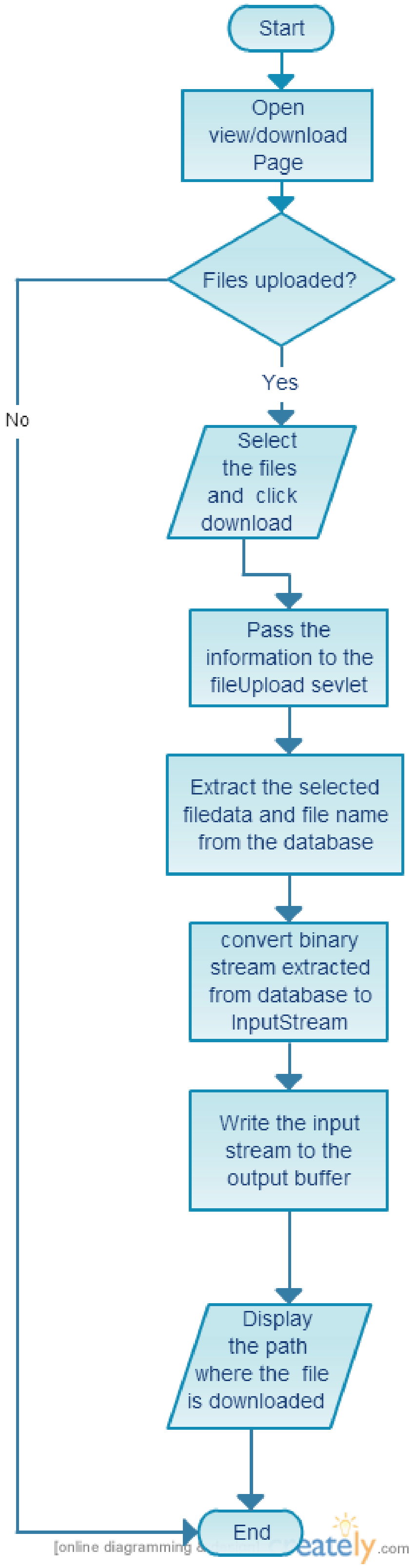
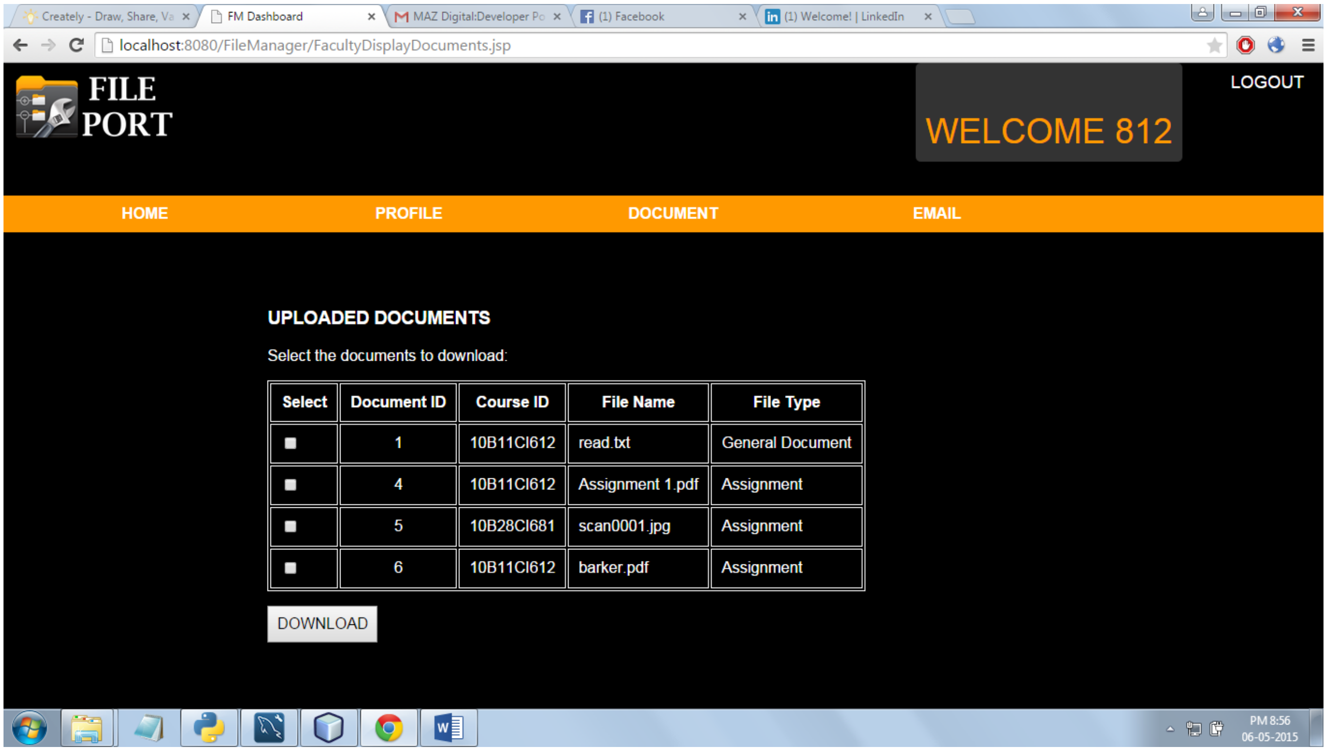

Start
Open view/download Page
Files uploaded?
Yes
No
Select the files and click download
Pass the information to the fileUpload sevlet
Extract the selected filedata and file name from the database
convert binary stream extracted from database to InputStream
Write the input stream to the output buffer
Display the path where the file is downloaded
End
[online diagramming
eately.com

Figure 15: File View / Download Flow Chart

5.2. User Interface

The figure below shows the UI for viewing/downloading files uploading by the faculty on the system. The documents are shown in a tabular form, with their details. The user can click on the checkboxes to select multiple files and click on download to download them.



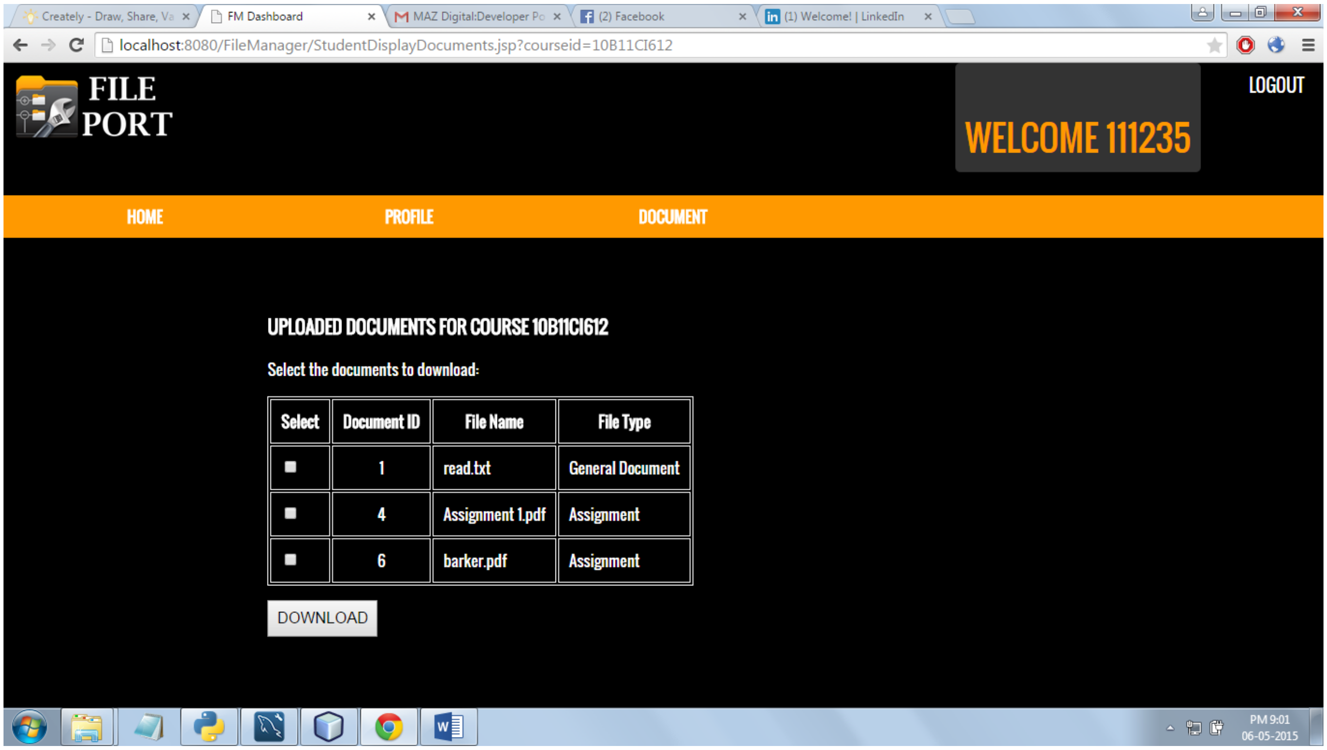
UPLOADED DOCUMENTS

Select the documents to download:

Select	Document ID	Course ID	File Name	File Type
■	1	10B11CI612	read.txt	General Document
■	4	10B11CI612	Assignment 1.pdf	Assignment
■	5	10B28CI681	scan0001.jpg	Assignment
■	6	10B11CI612	barker.pdf	Assignment

Figure 16: UI for File View/Download – Displaying uploaded docuemnts by faculty user

The next figure shows the UI where the students can download the documents uploaded by their faculty. Before navigating to this page, the students user is required to select the folder whose files he wants to sees. On clicking the folder, the files uploaded on it are diplayed, along with their details. The students user can select the files he want to download and click download. On successful download, the path where the files are saved is diplayed.



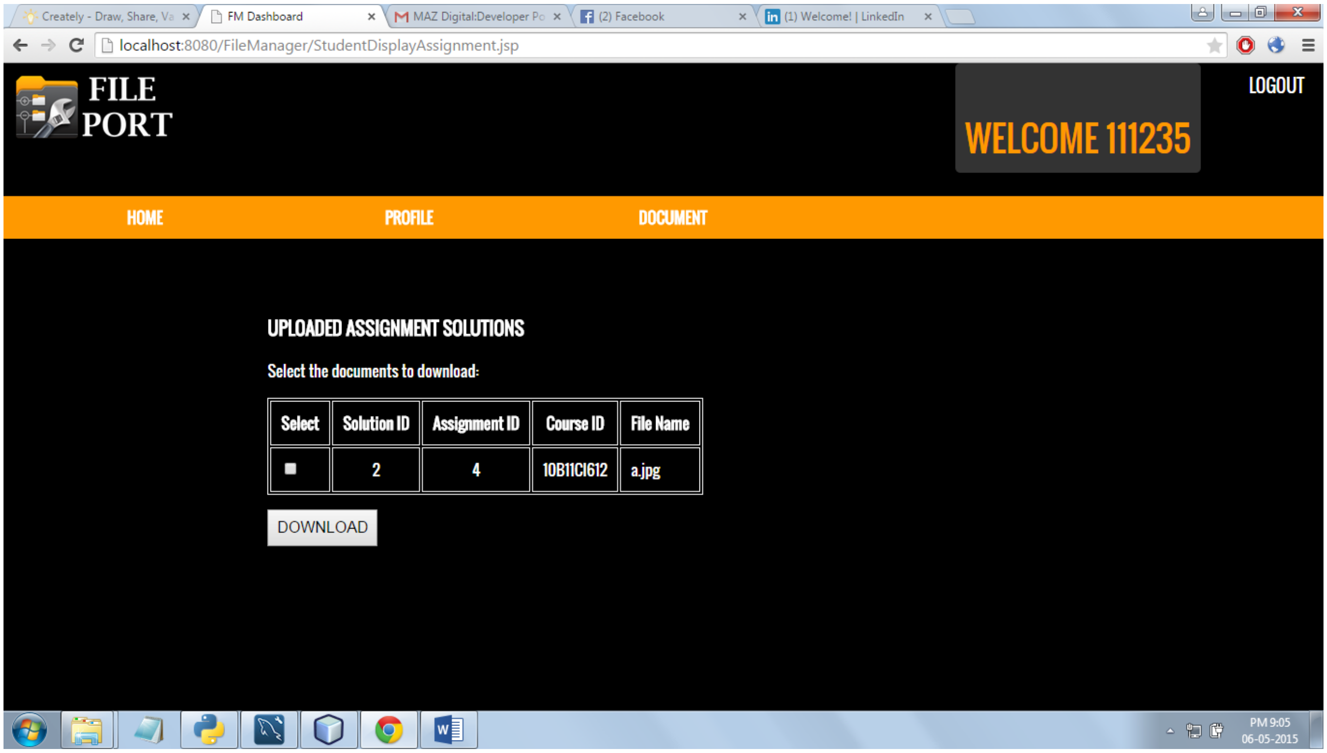
UPLOADED DOCUMENTS FOR COURSE 10B11CI612

Select the documents to download:

Select	Document ID	File Name	File Type
■	1	read.txt	General Document
■	4	Assignment 1.pdf	Assignment
■	6	barker.pdf	Assignment

Figure 17: UI for File View/Download – Displaying documents uploaded by faculty on student's end

The figure below shows the UI where the student user can view all the solutions uploaded by him in response to the assignments. All the details are given along with that document as shown. The student can select the documents he wish to download and clickon download. On suceessfull download, the path where the files are saved is diplayed.

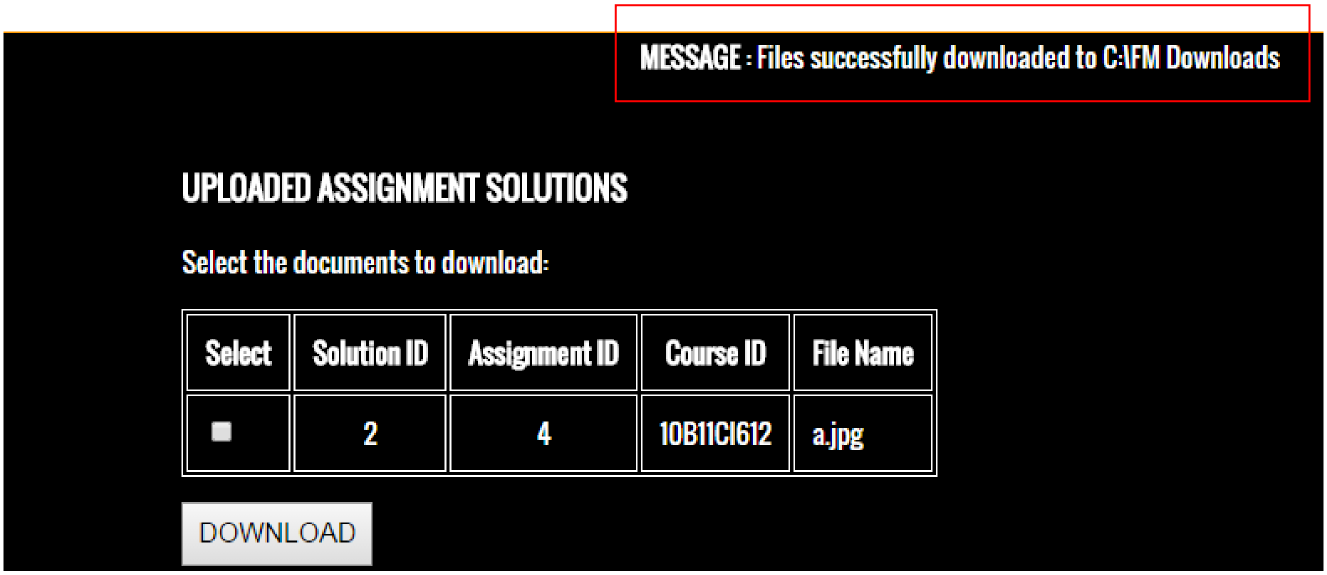


UPLOADED ASSIGNMENT SOLUTIONS

Select the documents to download:

Select	Solution ID	Assignment ID	Course ID	File Name
■	2	4	10B11CI612	a.jpg

Figure 18: UI for File View/Download – Displaying assignment solutions uploaded by the faculty



MESSAGE : Files successfully downloaded to C:\FM Downloads

UPLOADED ASSIGNMENT SOLUTIONS

Select the documents to download:

Select	Solution ID	Assignment ID	Course ID	File Name
■	2	4	10B11CI612	a.jpg

Figure 19: UI for File View/Download – Message displaying the path where file is stored

6. Delete Uploaded Files

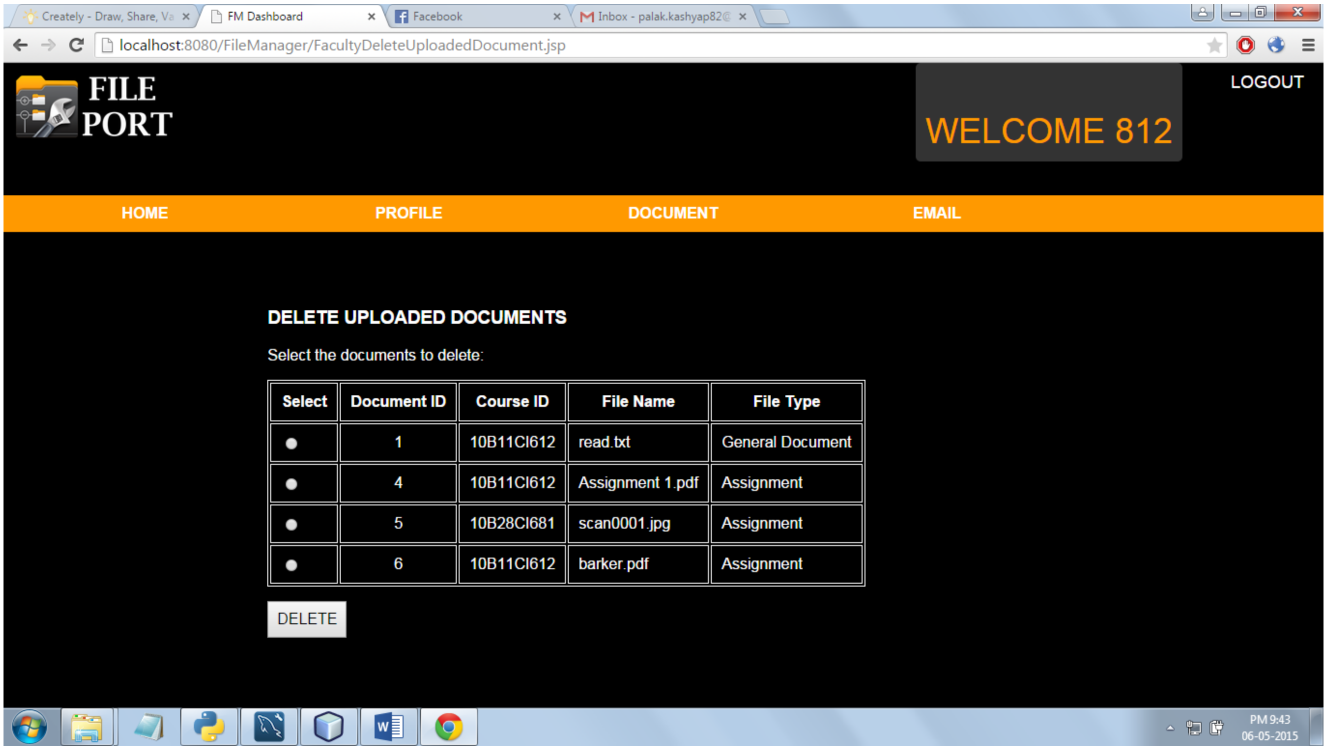
This features varies with different users. It has a slightly different functionality for student users than that for faculty users. The type of document both can delete from the system is different.

- Faculty users: The faculty users can delete only those files which are uploaded by them on the system.
- Student users: The student users can delete only those files which are uploaded as a solution to any assignment.

The system displays a list of files uploaded by the user, along with it's details. The users can then select the files and delete them.

6.1. User Interface

The figure below shows the UI for Faculty user to delete a document.



DELETE UPLOADED DOCUMENTS

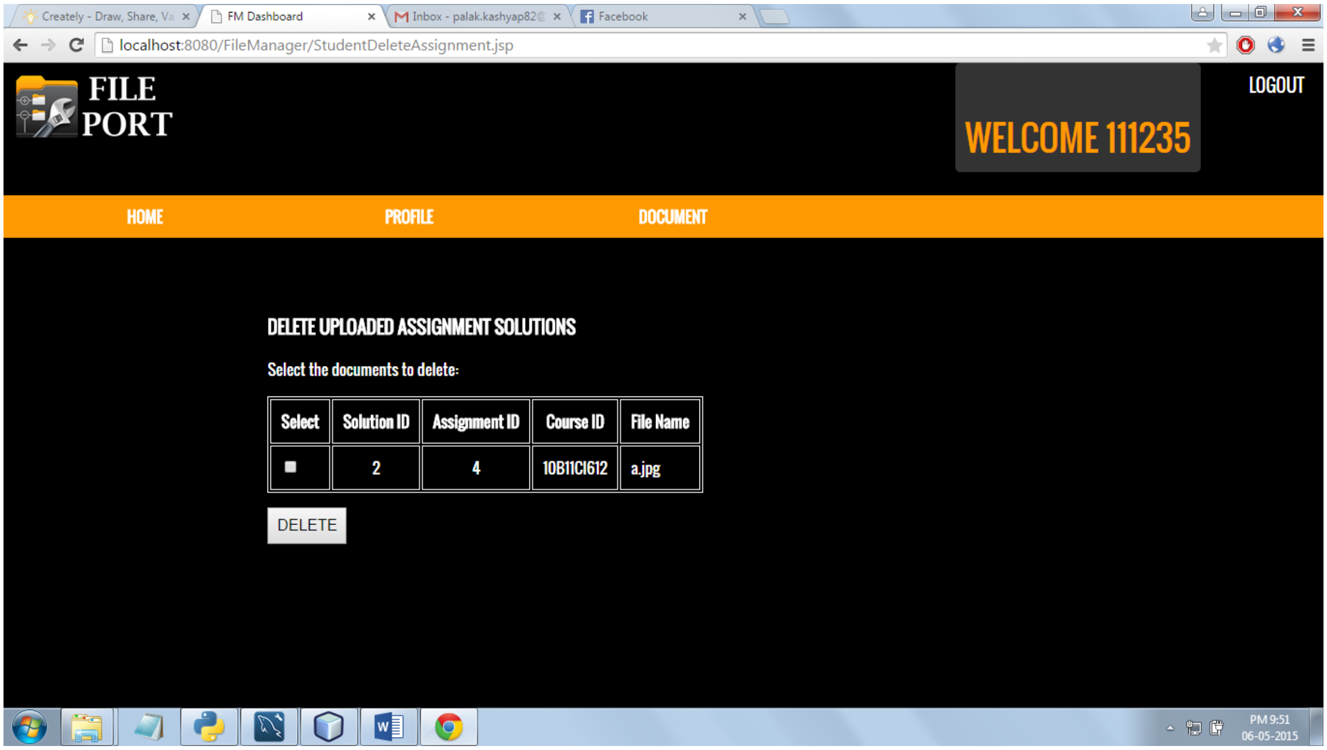
Select the documents to delete:

Select	Document ID	Course ID	File Name	File Type
●	1	10B11CI612	read.txt	General Document
●	4	10B11CI612	Assignment 1.pdf	Assignment
●	5	10B28CI681	scan0001.jpg	Assignment
●	6	10B11CI612	barker.pdf	Assignment

Figure 20: UI for File Delete – Displaying files that can be deleted by faculty user

All the files uploaded by the faculty user is listed in a tabular form with all the details. The user can select one file at a time, and click the DELETE button to delete the file from the system. On successful deletion, the system displays a message on the screen.

The next figure shows the UI for student users to delete the files uploaded as a solution to any assignment.



DELETE UPLOADED ASSIGNMENT SOLUTIONS

Select the documents to delete:

Select	Solution ID	Assignment ID	Course ID	File Name
■	2	4	10B11CI612	a.jpg

Figure 21: UI for File Delete – Displaying uploaded assignment solutions that can be deleted by the student

All the files uploaded by the student user is listed in a tabular form, along with their details. The user can select multiple files at a time and click on DELETE button to delete it from the system.

On successful deletion, a success message is displayed on the system, else an error message is displayed.

7. Email Notification

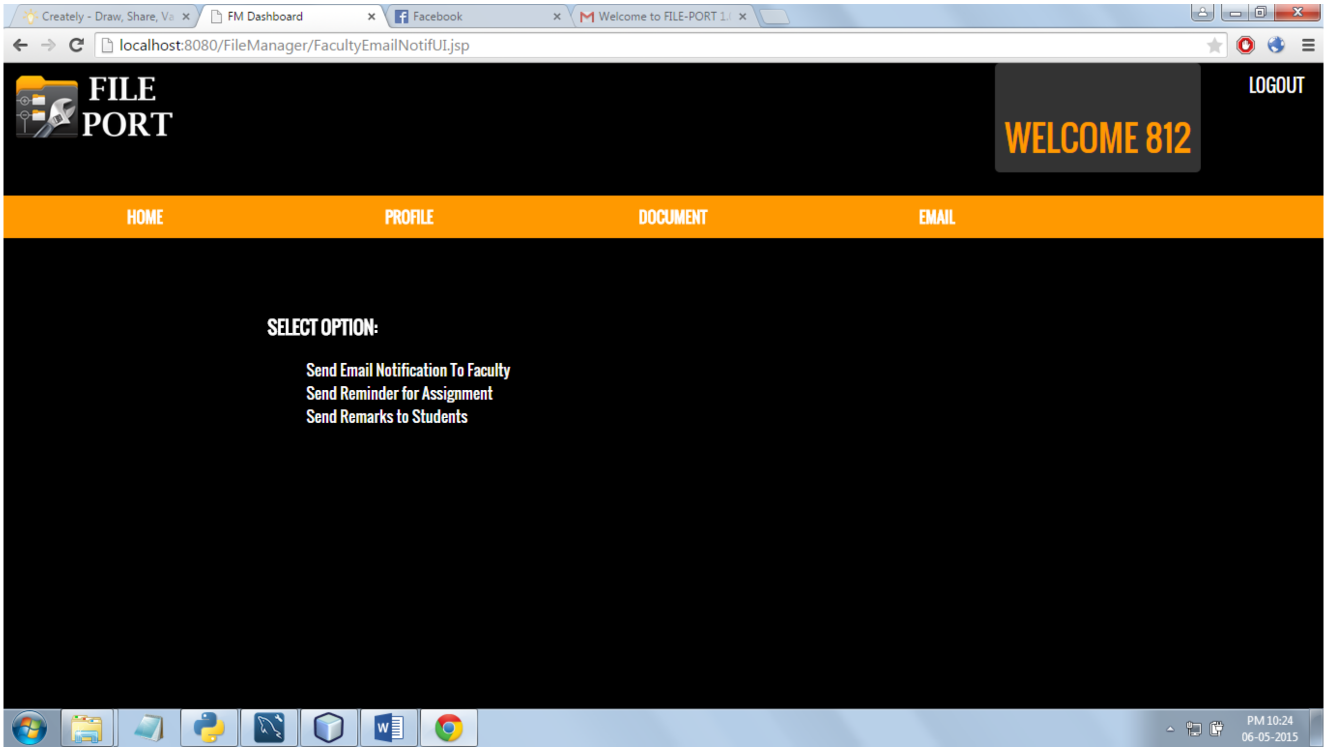
Email notification is one of the highlights of the FILE-PORT system. The use of this functionality is limited only to the Faculty users. The basic idea is to send an email through the system with a message. The system identifies the email id registered with it to send implicit email IDs. Sometimes, email IDs can also be asked for explicitly.

This functionality has various aspects:

- The system sends email notifications implicitly on various events like successful registration.
- The faculty user can send a message explicitly through the system to any other user.
- The faculty member can send a reminder to a batch for an assignment.
- The faculty user can send remarks to individual students through the system.

7.1. User Interface

Given below is the UI where all the options available are displayed



SELECT OPTION:

Send Email Notification To Faculty
Send Reminder for Assignment
Send Remarks to Students

Figure 22: UI for Email Notifications – Displaying the options available to the faculty

The previous figure shows the 3 options that tha Faculty user has under the functionality of Email Notification.

The figure below shows the UI for sending the email to other faculty users through the system. The user is required to enter the ID for the recipient user, the subject of the email, and the message, and click on SEND to send the email.

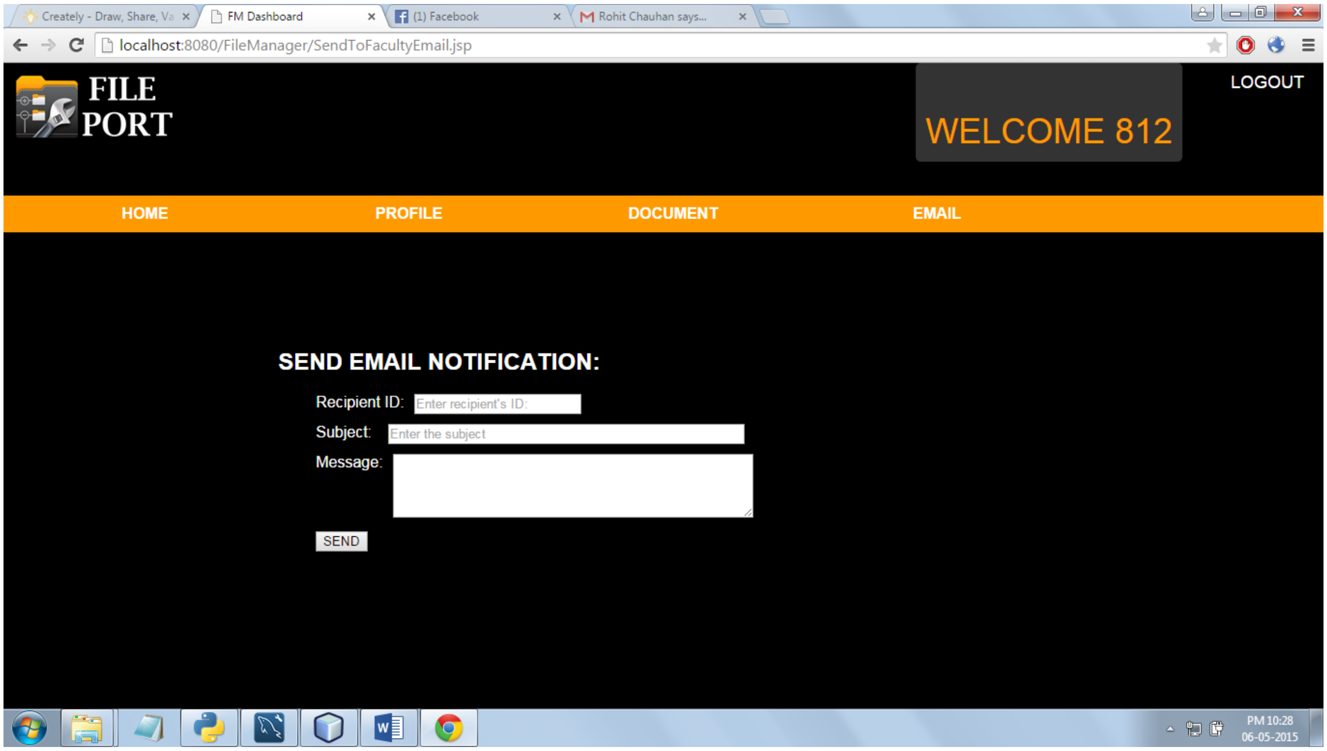
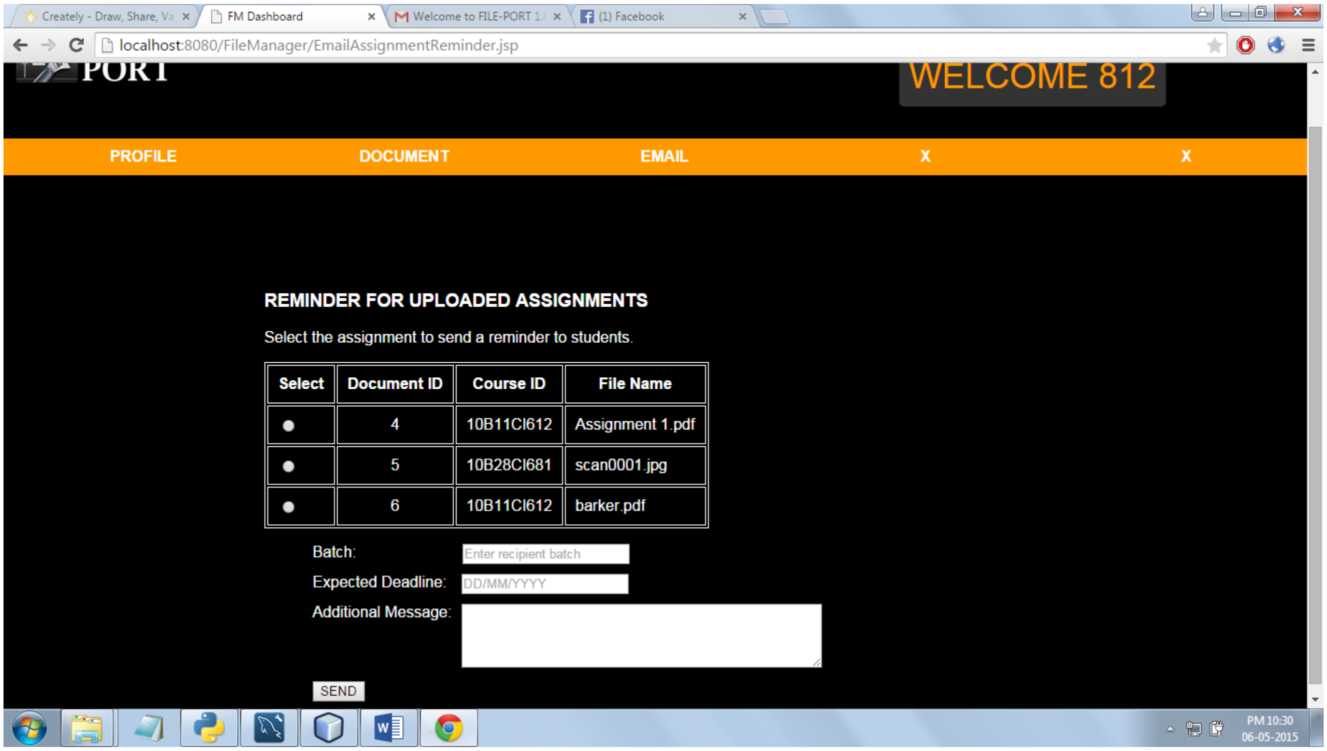


Figure 23: UI for Email Notifications – Seding email notification to another faculty

In the next figure, the UI for sending assignment reminders to a specific batch is shown. The user is required to select the assignment for which the reminder is to be sent, the expented deadline, and any additional message. The system then sends out an email to all the students of that batch, who are registered with the system.



REMINDER FOR UPLOADED ASSIGNMENTS

Select the assignment to send a reminder to students.

Select	Document ID	Course ID	File Name
●	4	10B11CI612	Assignment 1.pdf
●	5	10B28CI681	scan0001.jpg
●	6	10B11CI612	barker.pdf

Batch: Enter recipient batch

Expected Deadline: DD/MM/YYYY

Additional Message:

Figure 24: UI for Email Notifications – Sening assignment reminder to students

7.2. JavaMail API

The module uses JavaMail API. The JavaMail API provides a platform-independent and protocol-independent framework to build mail and messaging applications. The JavaMail API is available as an optional package for use with the Java SE platform and is also included in the Java EE platform.[4] The javax.mail package defines classes that are common to all mail systems. The javax.mail.internet package defines classes that are specific to mail systems based on internet standards such as MIME, SMTP, POP3, and IMAP. [5]

In the implementation, we use the SMTP Server provided by Gmail.

7.3. Steps to send email using JavaMail API

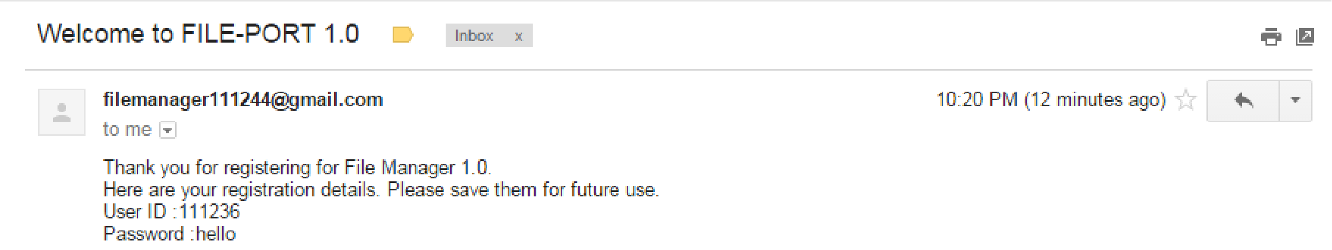
There are following three steps to send email using JavaMail. They are as follows:

1. Get the session object that stores all the information of host like host name, username, password etc.
2. compose the message
3. send the message

7.4. Implicit messages through the system

The system sends some email notifications implicitly, like on successful registration with the system for both faculty and the students.

Given below is a sample of such email sent by the system.



Thank you for registering for File Manager 1.0.
Here are your registration details. Please save them for future use.
User ID :111236
Password :hello

Figure 25: Preview of the email notification sent on successful registration

8. Upload Assignment Solutions

The online assignment submission system is an integral part of FILE-PORT.

When a faculty uploads an assignment on some folder in the system, the assignment is displayed on the student's end, under that same folder. The student can select any assignment and upload a single file as a solution to it. Thus uploading as assignment by the faculty results in opening up a link on the student's end where the student can submit the solution.

For the convenience of the student, the system also displays which assignments are pending, and which all have been submitted. A student can select only a pending document and upload a solution.

8.1. Flow Chart

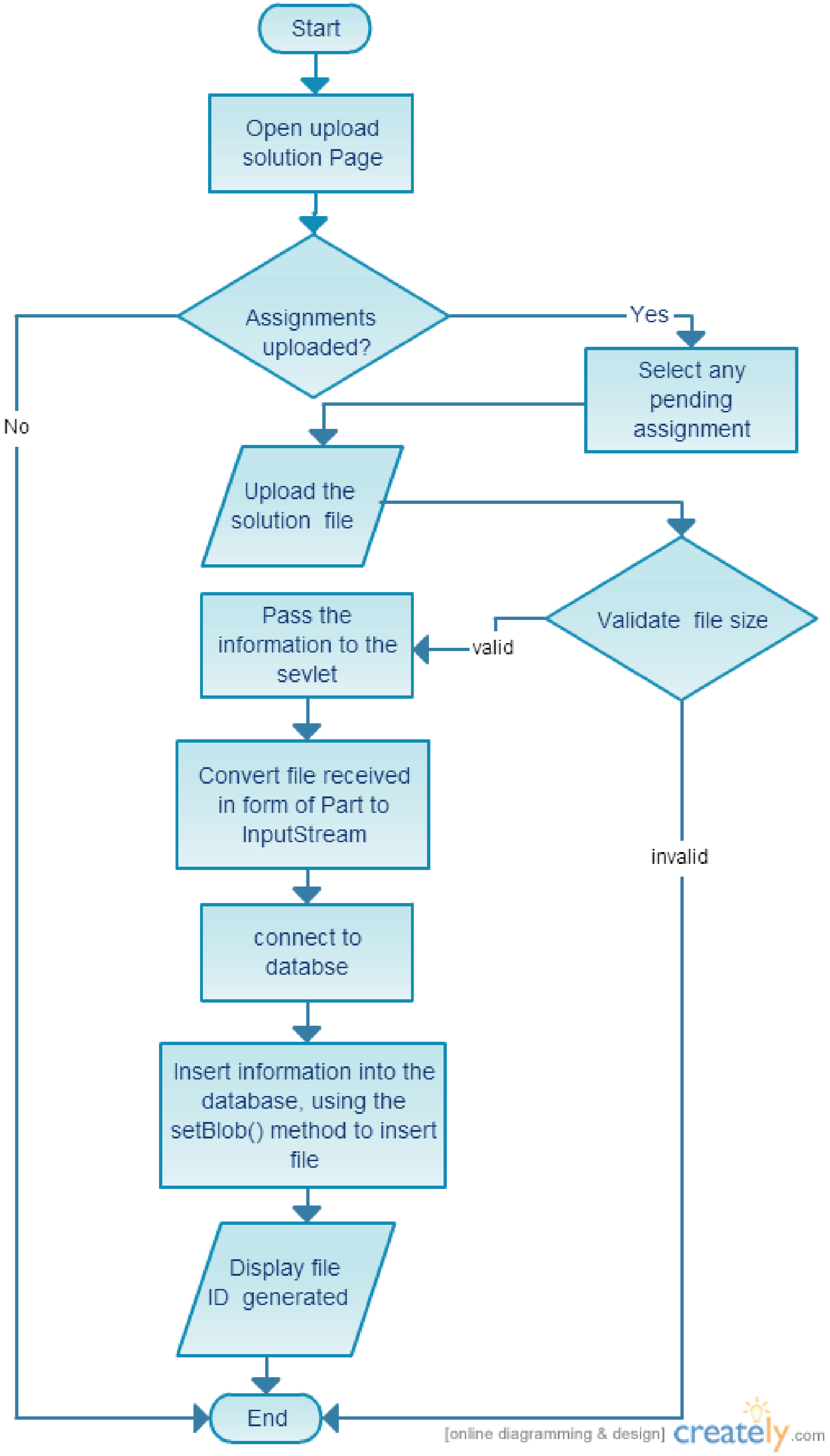
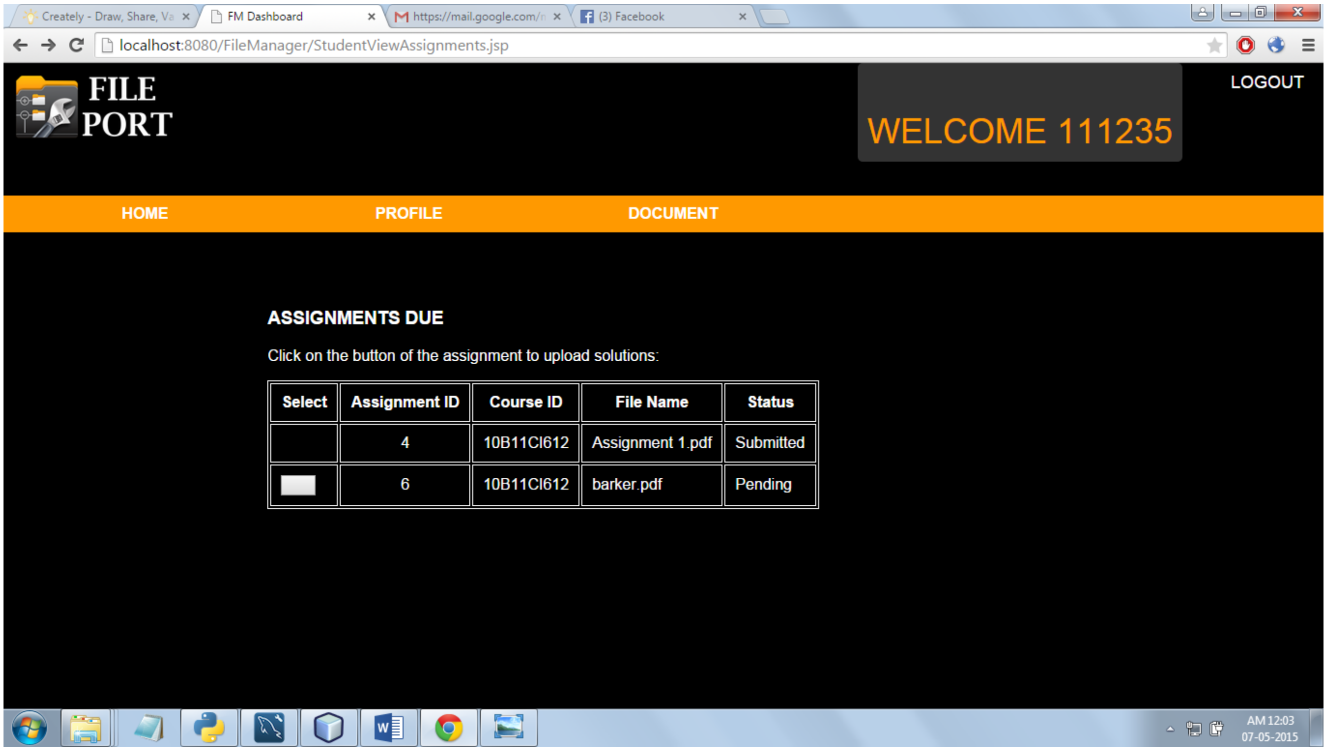

Start
Open upload solution Page
Assignments uploaded?
Yes
No
Select any pending assignment
Upload the solution file
Validate file size
valid
invalid
Pass the information to the sevlet
Convert file received in form of Part to InputStream
connect to databse
Insert information into the database, using the setBlob() method to insert file
Display file ID generated
End
[online diagramming & design] creately.com

Figure 26: Upload assignment solutions flow chart

8.2. User Interface

Given below is the screenshot of the UI which displays the list of all assignments for a student user. All the assignment are displayed in a tabular form, along with their details. The last column shows the status of the assignment – Submitted/Pending. The students can upload solutions to only the pending assignments.

For the pending assignments, a button is displayed clicking on which the student is taken forward to the next page where he can upload the solutions.



ASSIGNMENTS DUE

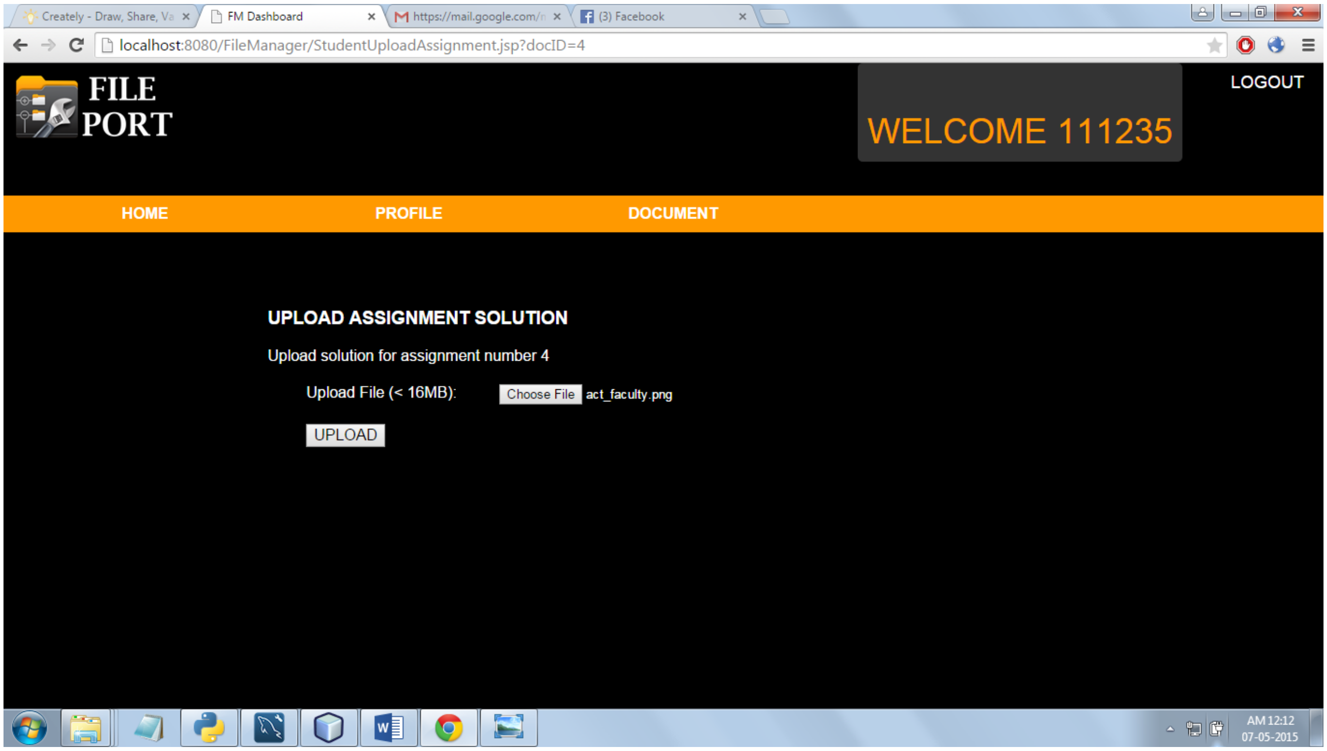
Click on the button of the assignment to upload solutions:

Select	Assignment ID	Course ID	File Name	Status
	4	10B11CI612	Assignment 1.pdf	Submitted
□	6	10B11CI612	barker.pdf	Pending

Figure 27 : UI for upload ssignment solution - Displaying the avaible assignments

The next figure shows the page where the students can select and upload a file as a solution to any assignment. The size of the file should not be greater tha 16MB.

On clicking upload, the file is successfully inserted into the system and the ID generated for the file is displayed.



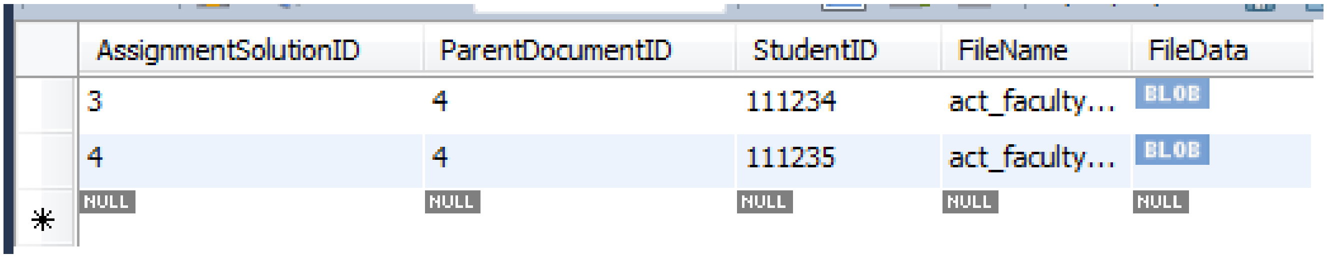
UPLOAD ASSIGNMENT SOLUTION

Upload solution for assignment number 4

Upload File (< 16MB): act_faculty.png

Figure 28: UI for upload assignment solutions - Select file to upload as assignemnt solution

The image below shows the files stored in the databse system.



	AssignmentSolutionID	ParentDocumentID	StudentID	FileName	FileData
	3	4	111234	act_faculty...	BLOB
	4	4	111235	act_faculty...	BLOB
*	NULL	NULL	NULL	NULL	NULL

Figure 29: Assignment solution file as stored in the database

9. View/Download Students Submissions

When a faculty uploads any document, it opens up a link on the student's side where he can upload the solutions to the assignments. All these solutions that are uploaded by the students can be viewed by the faculty on their portal with a click of a button.

The faculty is required to select the assignment whose solutions, which are submitted by the students, is to be viewed. For this a list of all the assignments is generated and the user needs to select one of them.

On selecting an assignment, the faculty can see the whole list of students who have uploaded the solution for that assignment. The list contains the student's userid and the file name.

On clicking download, the faculty user can download the original assignment and all the submissions in a single so. Once they are successfully downloaded, the assignment and the solutions are deleted from the system.

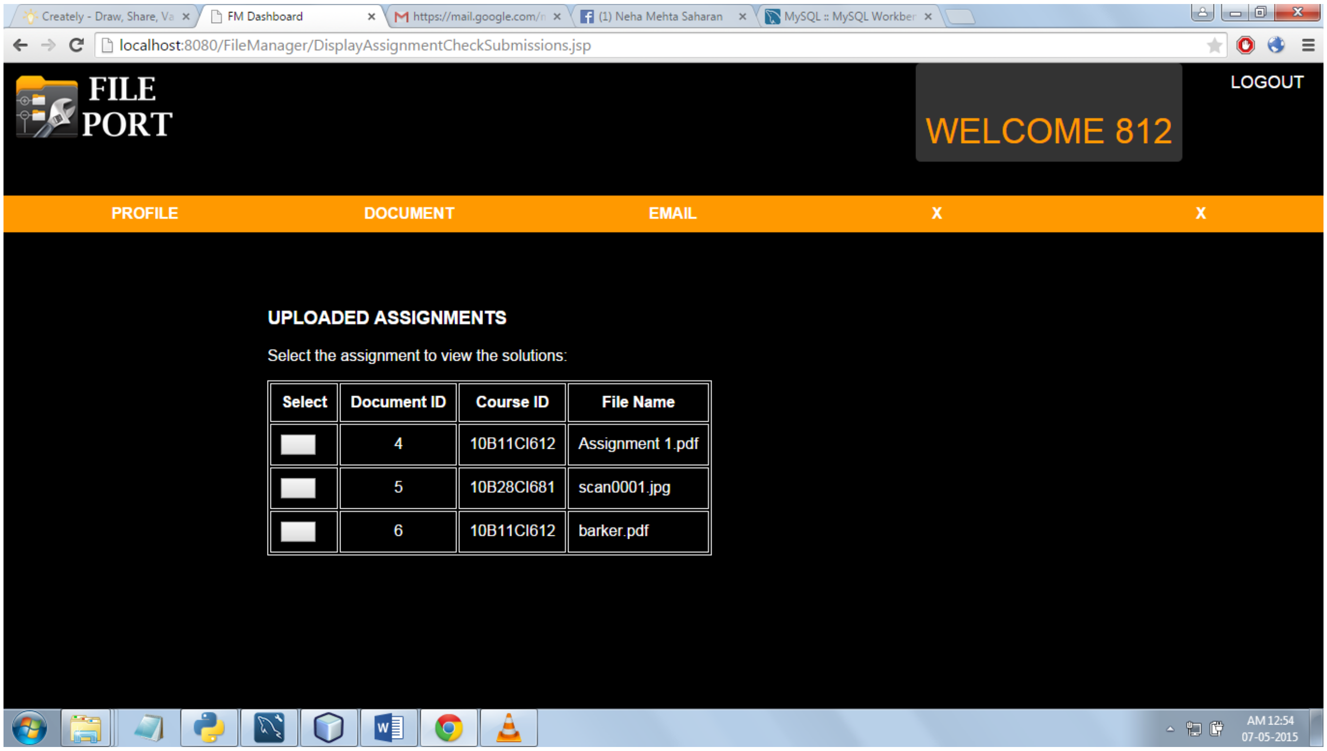
The file downloaded are saved together in a folder named 'Solutions_AssignmentNum<Assignment Number>'

9.1. User interface

Given below is a screenshot of the page where is user is made to select the assignment whose solutions he wants to view.

These are listed in a tabular form with their details along with them. The user can click on the button corresponding to the assignment whose solution he wants to see.

This will forward the user to the page with the list of students who have submitted the solutions.



UPLOADED ASSIGNMENTS

Select the assignment to view the solutions:

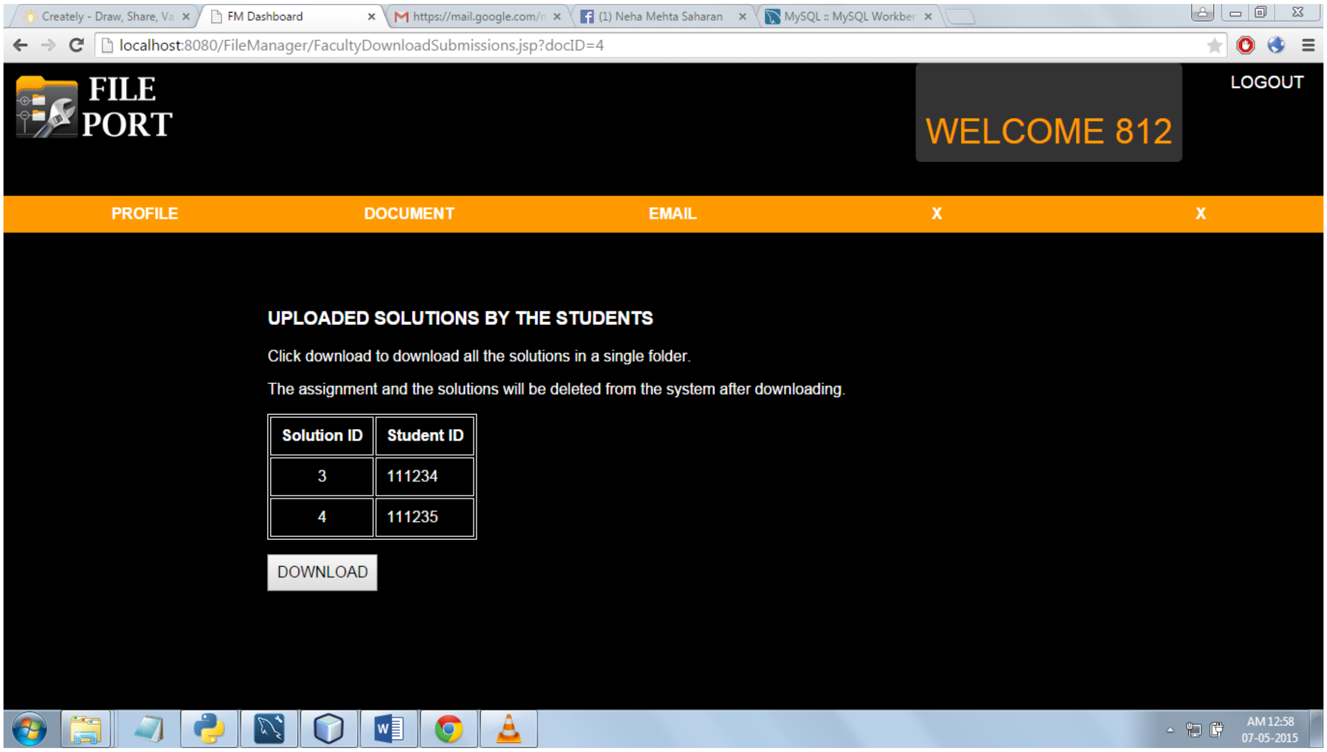
Select	Document ID	Course ID	File Name
□	4	10B11CI612	Assignment 1.pdf
□	5	10B28CI681	scan0001.jpg
□	6	10B11CI612	barker.pdf

Figure 30: UI for View/Download Assignment Solutions – Displaying the available assignments

On selecting the assignment, the faculty can view the list of students who have uploaded the solutions. The list is in a tabular form, with the Solution ID and the Student ID mentioned.

The user when clicks on DOWNLOAD, all the solutions are downloaded on the Faculty’s local system, along with the original assignment. Then the original assignment and all the submissions are deleted from the system.

The file downloaded are saved together in a folder named ‘Solutions_AssignmentNum<Assignment Number>’.



UPLOADED SOLUTIONS BY THE STUDENTS

Click download to download all the solutions in a single folder.

The assignment and the solutions will be deleted from the system after downloading.

Solution ID	Student ID
3	111234
4	111235

Figure 31: UI for View/Download Assignment Solutions – Displaying list of assignment solutions submitted

10. User Registration

The user need to register with the system to gain access to it and use it’s functions. There are two types of registration:

- Student Registration / Sign Up: For this, the student is required to enter the following details:

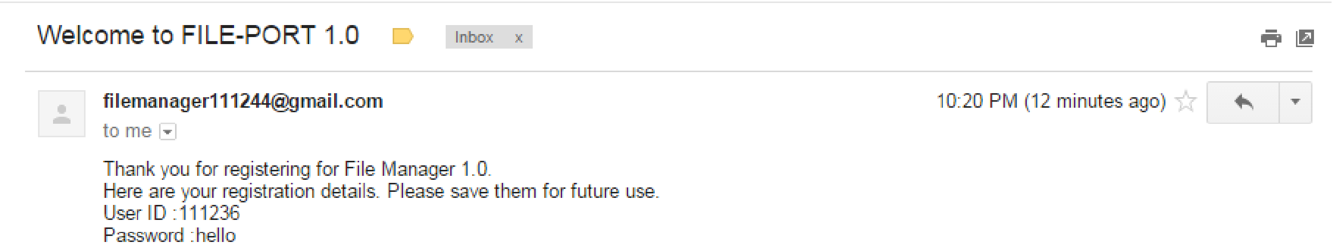
  1. User ID
  2. First Name
  3. Last Name
  4. Email Id
  5. Department
  6. Course Type (B.Tech/M.Tech)
  7. Batch

8. Semester
9. Password

- Faculty Registration / Sign Up: For this, the faculty user is required to enter the following details:
  1. User ID
  2. First Name
  3. Last Name
  4. Faculty Code
  5. Email
  6. Department
  7. Password

On successful registration with the system, an email notification is also sent by the system automatically on the registered email ID.

The front end of this functionality is developed using HTML and CSS, and JavaScript is used for form validations. JAVA servlets is used for backend. JDBC is used for communication with the database.



Thank you for registering for File Manager 1.0.
Here are your registration details. Please save them for future use.
User ID :111236
Password :hello

Figure 32: Preview of the email notification sent on successful registration

10.1.Flow Chart

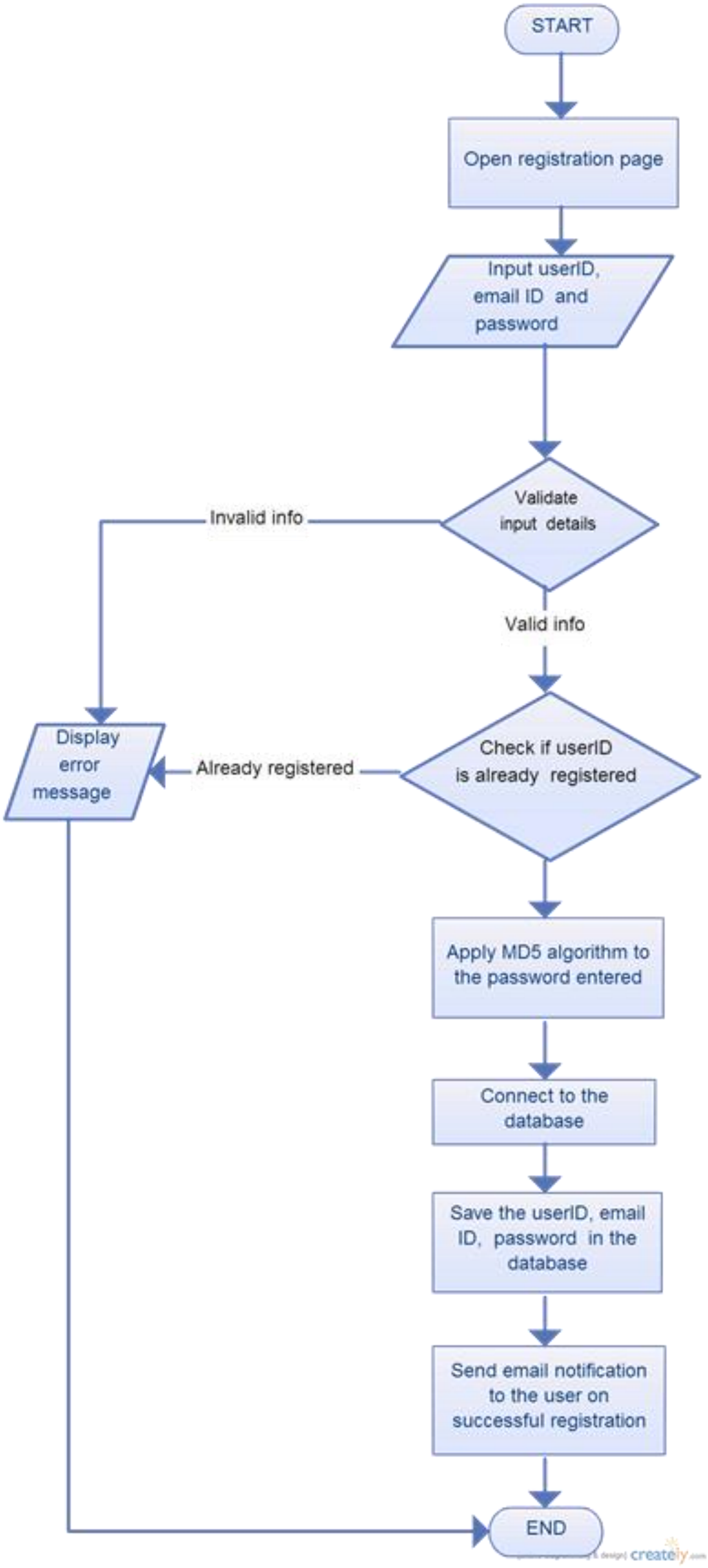
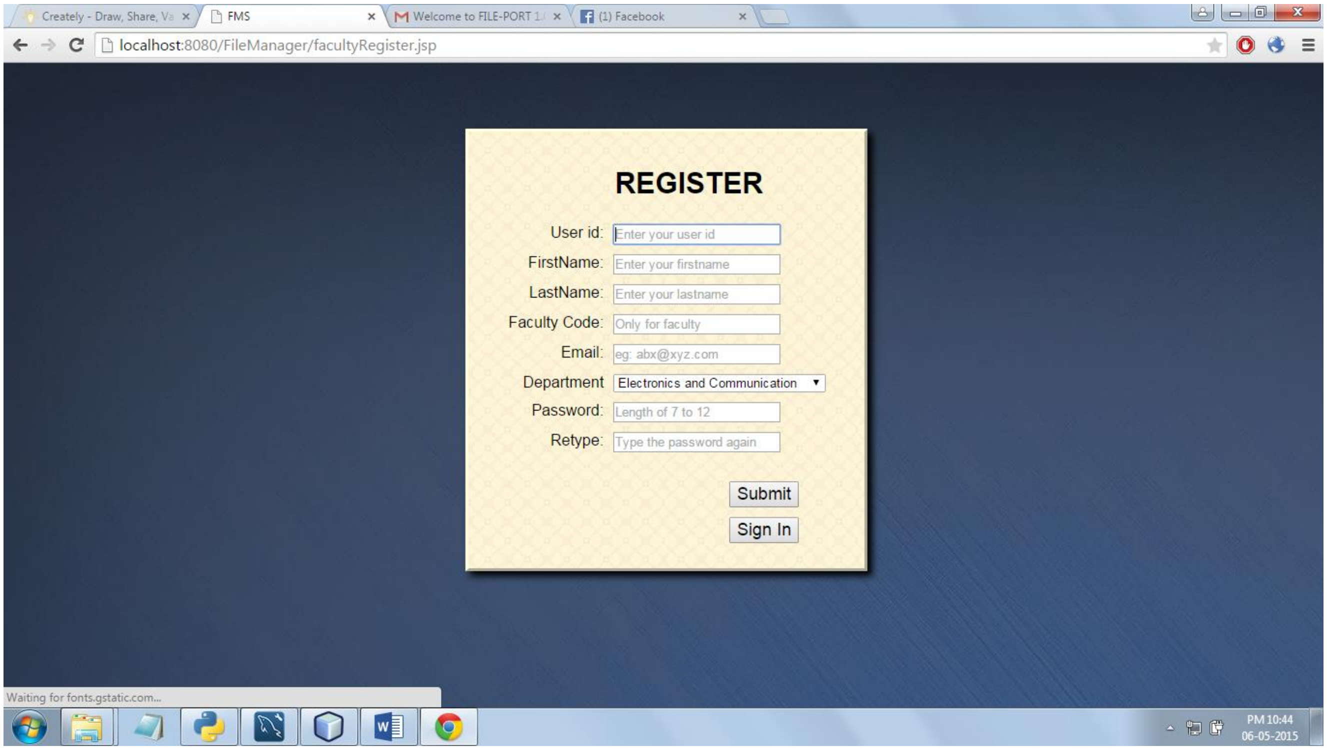

START
Open registration page
Input userID, email ID and password
Validate input details
Invalid info
Valid info
Check if userID is already registered
Already registered
Display error message
Apply MD5 algorithm to the password entered
Connect to the database
Save the userID, email ID, password in the database
Send email notification to the user on successful registration
END
creately

Figure 33: Register User Flow Chart

10.2. User Interface

Given below is a screen shot of the user interface of the registration page for Faculty type users. The user is required to enter all the details and click on submit to register with the system.

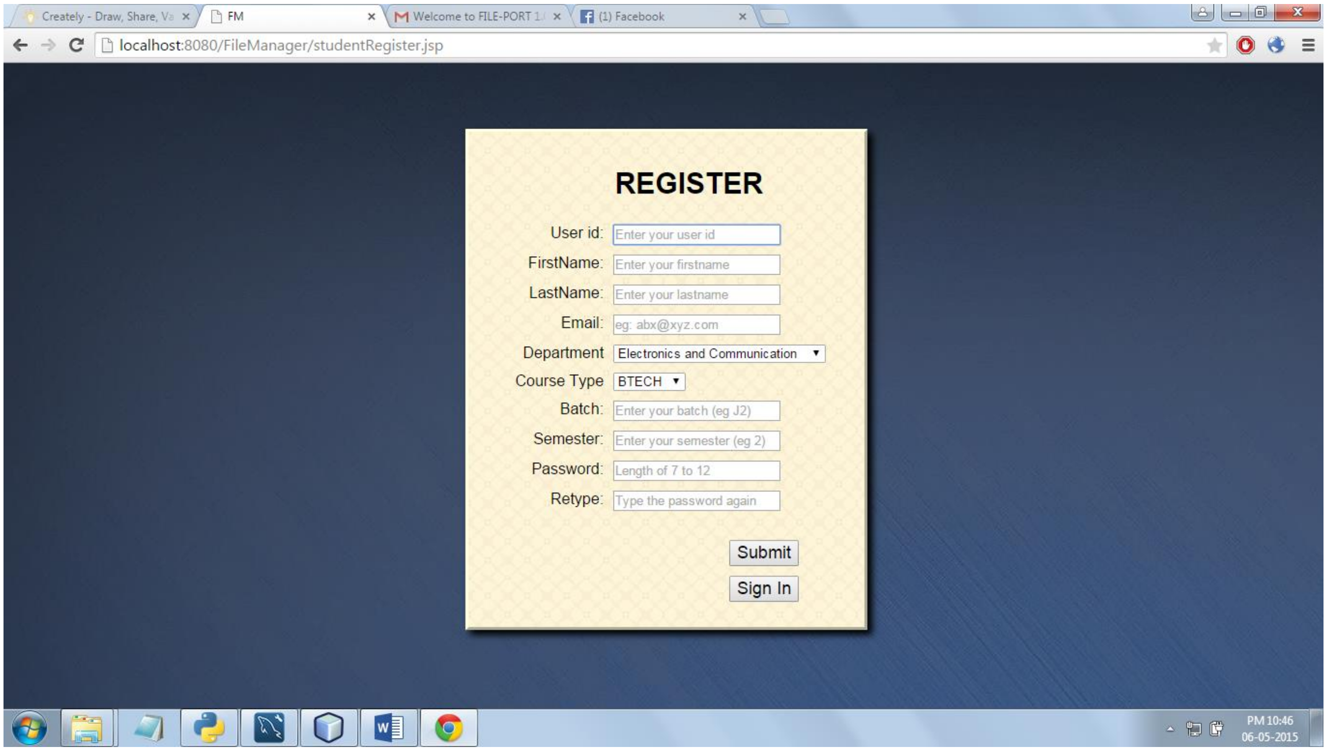

Creately - Draw, Share, Va
FMS
Welcome to FILE-PORT 1/
(1) Facebook
localhost:8080/FileManager/facultyRegister.jsp
Waiting for fonts.gstatic.com...
PM 10:44
06-05-2015

REGISTER

User id: Enter your user id
FirstName: Enter your firstname
LastName: Enter your lastname
Faculty Code: Only for faculty
Email: eg: abx@xyz.com
Department Electronics and Communication
Password: Length of 7 to 12
Retype: Type the password again

Figure 34: UI for Register User – Faculty Registration

The next figure give a screenshot of the user interface of the registration page for Student type users. The user is required to enter all the details and click on submit to register with the system.



Creately - Draw, Share, Va
FM
Welcome to FILE-PORT 1.
(1) Facebook
localhost:8080/FileManager/studentRegister.jsp
PM 10:46
06-05-2015

REGISTER

User id: Enter your user id

FirstName: Enter your firstname

LastName: Enter your lastname

Email: eg: abx@xyz.com

Department: Electronics and Communication

Course Type: BTECH

Batch: Enter your batch (eg J2)

Semester: Enter your semester (eg 2)

Password: Length of 7 to 12

Retype: Type the password again

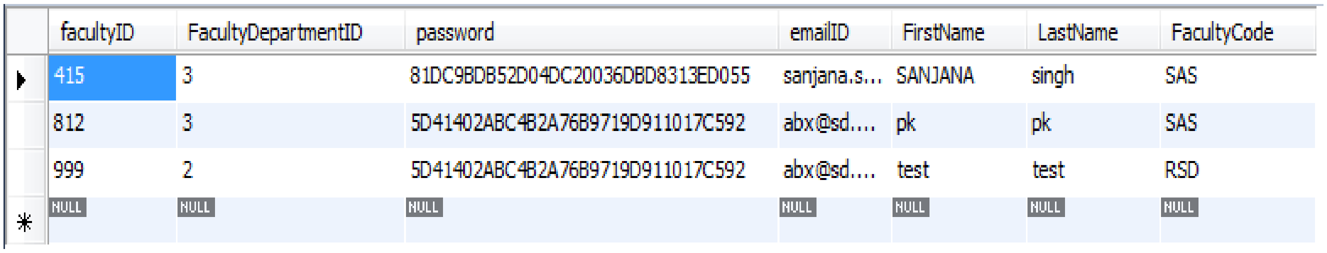
Figure 35: UI for Register User – Student Login

10.3. MD5

MD5 which stands for Message Digest algorithm 5 is a widely used cryptographic hash function producing a 128-bit (16-byte) hash value, typically expressed in text format as a 32 digit hexadecimal number. The idea behind this algorithm is to take up a random data as an input and generate a fixed size “hash value” as the output. The input data can be of any size or length, but the output “hash value” size is always fixed. It provides one way hashing, once an input is hashed, it cannot be converted back to the original text.

In the implementation, MesseageDigest class is directly used to convert passwords to MD5 hash. This MessageDigest class provides applications the functionality of a message digest algorithm, such as MD5. Message digests are secure one-way hash functions that take arbitrary-sized data and output a fixed-length hash value. [6]

The system uses MD5 algorithm to store the passwords in the database.



	facultyID	FacultyDepartmentID	password	emailID	FirstName	LastName	FacultyCode
▶	415	3	81DC9BDB52D04DC20036DBD8313ED055	sanjana.s...	SANJANA	singh	SAS
	812	3	5D41402ABC4B2A76B9719D911017C592	abx@sd....	pk	pk	SAS
	999	2	5D41402ABC4B2A76B9719D911017C592	abx@sd....	test	test	RSD
✳	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Figure 36: Passwords stored as MD5 hash in the database

11. User Login

Once a user is registered with the system, he needs to log in to use the functionalities of the system.

While logging into the system, the user must enter the Login type, i.e. Faculty/Student along with the user ID and the password.

The front end of this functionality is developed using HTML and CSS, and JavaScript is used for form validations. JAVA servlets is used for backend. JDBC is used for communication with the database.

11.1. Flow Chart

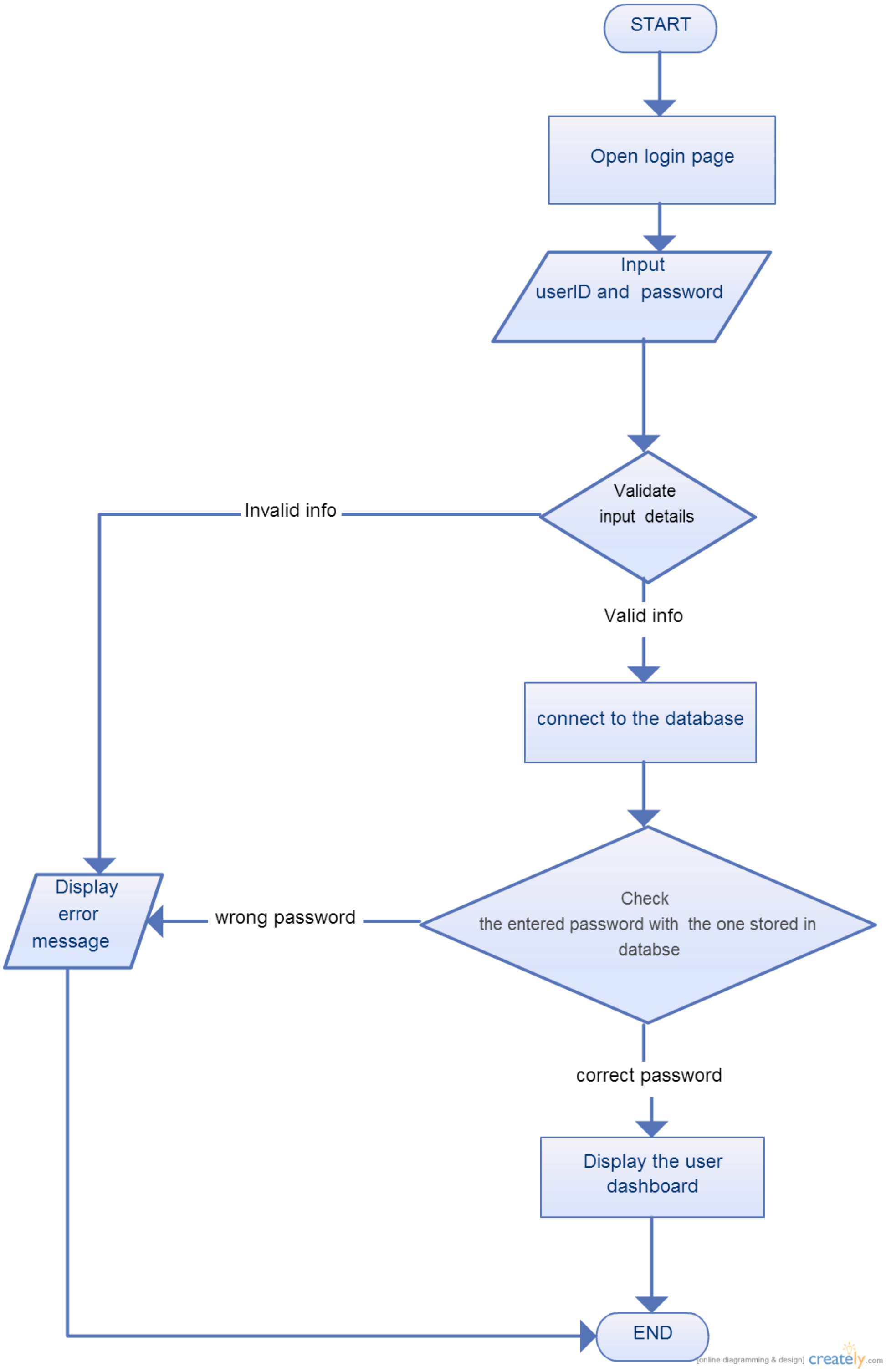
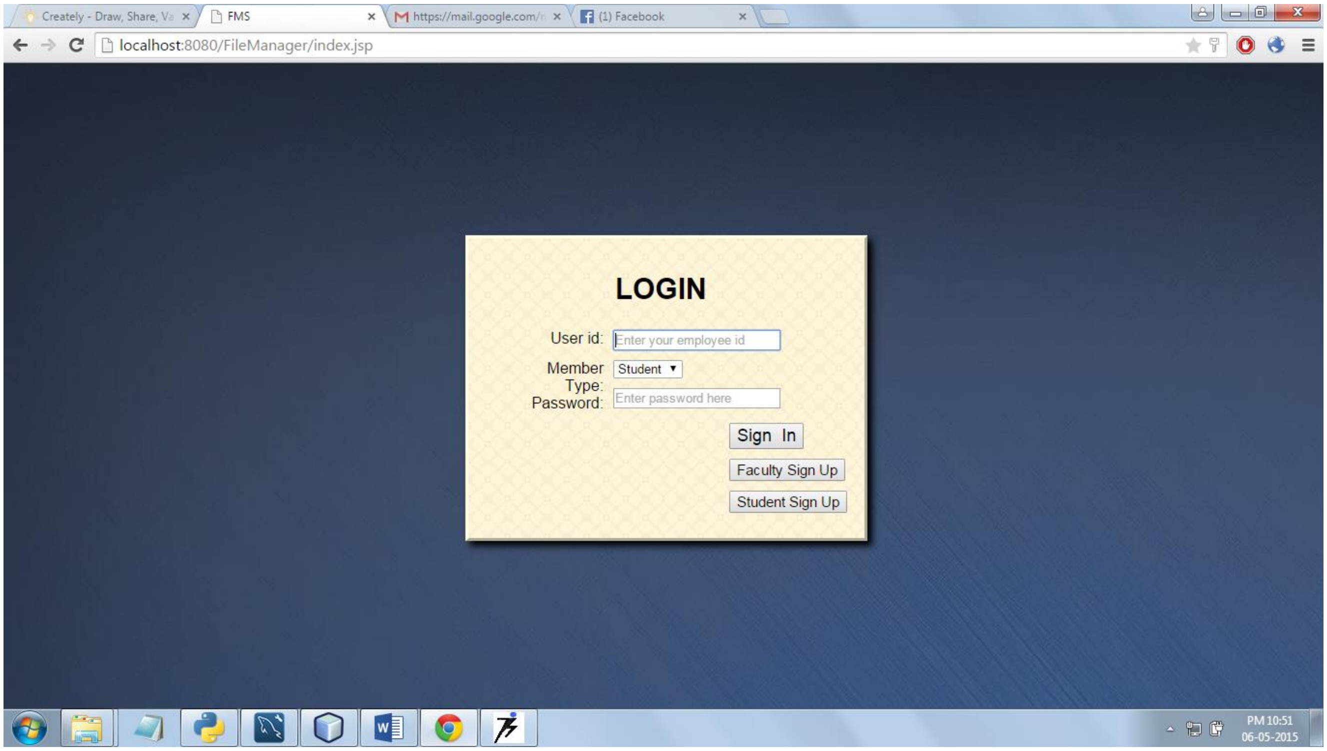


Figure 37: User Login Flow Chart

11.2. User Interface

Given below is a screen shot of the user interface of the registration page.



User id: Enter your employee id

Member Type: Student

Password: Enter password here

Figure 38: UI for Login Page

CHAPTER 4 : CONCLUSIONS AND FUTURE WORK

This Chapter includes the Conclusion reached after creating few modules of the Software in regards to the functional requirements of the System. It also list the future work that is intended to be accomplished and discusses the limitations to system.

1. Conclusions

Various aspects regarding the project has been discussed throughout the report. The basic idea of the FILE-PORT is to improve the flow of information, and make the process of sharing of documents more efficient. It aims to make the process of sharing documents go paperless and encourage a transition to electronic form of file sharing. There are many benefits of shifting from the traditional paper based documents system to web based system, and this project promotes this change by providing an efficient system for the same.

FILE-PORT provides a user friendly way to share documents and has been developed to suit the requirements at JUIT. Various features has been added so that the system can be integrated with the functioning of the university. The system allows the faculty to upload the documents or file, that are required to be shared among the users. The students can use the system to view and download the files being shared from any location. Thus it provides an efficient way of document sharing.

One of the main aim of FILE PORT is to make the assignment system go paperless and make it more efficient. This system allows the Faculty to upload the assignments on the system. The students can then view the assignments and submit the solutions online. This makes the collection process easier as we get all the assignments at a single space.

One of the highlights of the system is the classification system, where the documents are classified according to the subjects they are related to. The system identifies the subjects allocated to each user dynamically by parsing the time table. Thus a user views documents of only those subject which has been allocated to him.

The email notification system integrated with FILE-PORT provides an easy way of communication, where the faculty users can send messages to other users, using the email IDs registered with the system. This therefore allows a faculty to sends reminders and remarks to students, reducing the gap between them.

2. Future Work

For future work, following can be taken up:

- Integrate text editors with the system, to allow users to create text documents online and save them with the system.
- Integrate tools which would check plagiarism in the assignment solutions uploaded by the students, and display the results.
- The project can have an online marking system which allows the Faculty to mark the assignment solutions submitted. These marks can be viewed by the students as well. This will promote transparency in the system.
- The project can be scaled to include an online examination system as well, which will allow students to give tests from any location. The results will be compiled automatically and saved with the system.

3. Limitations

Although the best efforts have been put to develop a flawless software system, still a few limitations of the system have been identified. These limitations are as follows:

- The system currently works only for B.Tech students, and not for those who have taken up an M.Tech course at JUIT.
- The system also doesn’t work for the fourth year students. Although they can be registered with the system, but the functionalities may not work properly for them.
- The system downloads all the assignment submissions in one go. If a faculty wants to download only few of them, he won’t be able to do so.
- While downloading the files, the user cannot explicitly enter the path where he wants the files to be saved. The location where the files are downloaded and saved is predefined, and cannot be changed.

REFERENCES

[1] Akash Kava. (2009, August 23) *Huge File Storage in Database instead of File System* [Online]. Available: http://akashkava.com/blog/127/huge-file-storage-in-database-instead-of-file-system/. [Accessed 17 Dec 2014].

[2] S. Barker, B. Fiedler, P. Johnson, "Paperless assignments: Moving forward or marking time?," in *Proceedings of ASCILITE - Australian Society for Computers in Learning in Tertiary Education Annual Conference 2008* , pp. 45-55. Available: http://www.editlib.org/p/45953/. [Accessed 20 September 2014]

[3] D. Jones, S. Behrens, " Online assignment management: an evolutionary tale," in *Proceedings of the 36th Annual Hawaii International Conference on System Sciences, 6-9 Jan. 2003,* Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.95.5214&rep=rep1&type=pdf. [Accessed 15 September 2014].

[4] Oracle. (1999, February 11). *JavaMail* [Online]. Available: http://www.oracle.com/technetwork/java/javamail/index.html. [Accessed 16th Dec 2014]

[5] JavaMail. (1999, February 11) *JavaMail API documentation* [Online]. Available: https://javamail.java.net/nonav/docs/api/. [Accessed 16th Dec 2014]

[6] Oracle. *Class MessageDigest* [Online]. Available: http://docs.oracle.com/javase/7/docs/api/java/security/MessageDigest.html. [Accessed 16th Dec 2014]

APPENDIX A

Description of Tools

ADOBE DREAMWEAVER

Adobe Dreamweaver (formerly Macromedia Dreamweaver) is a proprietary web development tool originally created by Macromedia in 1997. It is now developed and maintained by Adobe Systems, which acquired Macromedia in 2005.

The tool is available for OS X and for Windows. Following Adobe's acquisition of the Macromedia product suite, releases of Dreamweaver subsequent to version 8.0 have been more compliant with W3C standards. Recent versions have incorporated support for web technologies such as CSS, JavaScript, and various server side scripting languages and frameworks including ASP (ASP JavaScript, ASP VBScript, ASP.NET C#, ASP.NET VB), ColdFusion, Scriptlet, and PHP.

With Dreamweaver, it's easy to create and edit cross-platform, cross-browser pages. It provides advanced design and layout tools, as well as making it easy to use Dynamic HTML features such as animated layers and behaviours without writing a line of code. Browser-targeting checks your work for potential problems on all popular platforms and browsers.

FEATURES:

1. Adobe Dreamweaver is a web design and development application that provides a visual WYSIWYG editor (colloquially referred to as the Design view) and a code editor with standard features such as:
   - Syntax highlighting
   - Automatic ode completion
   - Code collapsing
   - Real-time syntax checking

- Code introspection for generating code hints to assist the user in writing code.

2. Design view facilitates rapid layout design and code generation as it allows users to create and manipulate the layout of HTML elements.
3. Dreamweaver features an integrated browser for previewing developed webpages in the program's own preview pane in addition to allowing content to be open in locally installed web browsers.
4. It provides transfer and synchronization features, the ability to find and replace lines of text or code by search terms or regular expressions across the entire site, and a templating feature that allows single-source update of shared code and layout across entire sites without server-side includes or scripting.
5. Since version 5, Dreamweaver supports syntax highlighting for the many languages.
6. After version CS6, Adobe Dreamweaver is available in the following languages: Brazilian Portuguese, Simplified Chinese, Traditional Chinese, Czech, Dutch, English, French, German, Italian, Japanese, Korean (Windows only), Polish, Russian, Spanish, Swedish and Turkish.

NETBEANS IDE

NetBeans IDE is an open-source integrated development environment. It is primarily intended for development in Java, but also supports other languages, in particular PHP, C/C++ and HTML5. The application is cross-platform and runs on Microsoft Windows, Mac OS X, Linux, Solaris and other platforms supporting a compatible JVM. Among other features are an Ant-based project system, Maven support, refactoring, and version control (supporting CVS, Subversion, Git, Mercurial and Clearcase).

Modularity: All the functions of the IDE are provided by modules. Each module provides a well-defined function, such as support for the Java language, editing, or support for the CVS versioning system, and SVN. NetBeans contains all the modules needed for Java development in a single download, allowing the user to start working immediately.

Modules also allow NetBeans to be extended. New features, such as support for other programming languages, can be added by installing additional modules.

ADOBE PHOTOSHOP

Adobe Photoshop is a raster graphics editor developed and published by Adobe Systems for Windows and OS X. It can edit and compose raster images in multiple layers and supports masks, alpha compositing and several color models including RGB, CMYK, Lab color space, spot color and duotone. Photoshop has vast support for graphic file formats but also uses its own PSD and PSB file formats which support all the aforementioned features. In addition to raster graphics, it has limited abilities to edit or render text, vector graphics (especially through clipping path), 3D graphics and video. Photoshop's feature set can be expanded by Photoshop plug-ins, programs developed and distributed independently of Photoshop that can run inside it and offer new or enhanced features.

MYSQL WORKBENCH

MySQL Workbench is a visual database design tool that integrates SQL development, administration, database design, creation and maintenance into a single integrated development environment for the MySQL database system. Workbench is available on Windows, Linux and Mac OS X.

- DESIGN
  MySQL Workbench enables a DBA, developer, or data architect to visually design, model, generate, and manage databases. It includes everything a data modeler needs for creating complex ER models, forward and reverse engineering, and also delivers key features for performing difficult change management and documentation tasks that normally require much time and effort.
- DEVELOP
  MySQL Workbench delivers visual tools for creating, executing, and optimizing SQL queries. The SQL Editor provides color syntax highlighting, auto-complete,

reuse of SQL snippets, and execution history of SQL. The Database Connections Panel enables developers to easily manage standard database connections, including MySQL Fabric. The Object Browser provides instant access to database schema and objects.

- ADMINISTER
  MySQL Workbench provides a visual console to easily administer MySQL environments and gain better visibility into databases. Developers and DBAs can use the visual tools for configuring servers, administering users, performing backup and recovery, inspecting audit data, and viewing database health.