RIFLESSO

(ONLINE STUDENT COMPLETE SYSTEM WITH BIOMETRIC AUTHENTICATION)

Enrolment Number - 081444, 081433

Name - Sonali Sharma, Shubham Singhal

Name of Supervisor - Mr. Nitin Rakesh

May- 2012

Submitted in partial fulfillment of the Degree of

Bachelor of Technology

in

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING &

INFORMATION TECHNOLOGY

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

TABLE OF CONTENT

CERTIFICATE

This is to certify that the work titled "**Riflesso (Online Complete Student system with Biometric Authentication)**" submitted by "**Shubham Singhal & Sonali Sharma**" in partial fulfillment for the award of degree of "**Bachelor of Technology**" of Jaypee University of Information Technology, Waknaghat has been carried out under my supervision. This work has not been submitted partially or wholly to any other University or Institute for the award of this or any other degree or diploma.

Signature of Supervisor .....

Name of Supervisor Mr. Nitin Rakesh

Designation Senior Lecturer

Dated 31/05/12

ACKNOWLEDGEMENT

We wish to express our profound sense of gratitude to our supervisor, **Mr. Nitin Rakesh**, for his guidance, support and encouragement throughout the course of the project.

We would take this opportunity to thank the Head of Department of Computer Science and Engineering and Information Technology, **Brig. (Retd.) S. P. Gherera**, for providing all the technical facilities and being available at any time to help with the proceedings.

In the end, we would also like to thank all the panel members grading the project over the year for their valuable comments and evaluation.

Signature of the Student	Sonali Sharma	Signature of the Student	Shubham
Name of Student	Sonali Shrama	Name of Student	Shubham Singhal
Date	1/6/12	Date	1/6/12

ABSTRACT

The project named Riflesso, an intranet based Student System has tried to make all the processes related to a student in an educational institution. The processes like attendance and examinations have been automated. This project adds to the existing Student Management Systems and capitalizes over those systems by automating various other processes that a regular system does not encompass.

The project uses a hardware device, a Biometric Fingerprint Machine, for taking the attendance and also validates the identity of the student in many other processes. The use of the hardware device helps in managing the student information and also keeps the data consistent.

This project report summarizes the approach of the project and explains various modules of the system. Approach of various modules and their working will be explained as each chapter, showing the flow of data, how data is accessed and is getting updated. Various diagrams showing the flow of information and snapshots of the system will be provided explaining it in better way.

The report initially gives the introduction of the project, motivation behind the project and scope of the project. Then the approach of the project is discussed and various modules of the system are introduced and their brief description is given. The subsequent chapters will explain each module in detail describing the flow of process and data. Codes of various modules will be provided in the appendix for reference.

LIST OF FIGURES

CHAPTER 1

1. INTRODUCTION

RIFLESSO is an intranet and internet based software comprising of many small management systems, developed for daily student attendance, examinations, notifications in colleges and institutes. It facilitates to access the attendance information and examination marks of a particular student through his/her own student account that is accessible to the student through his user-id and password.

The automation of attendance is facilitated by the use of biometric fingerprint device. This device will record the daily attendance of the student by recording his/her fingerprint. This will automatically update the attendance record of the student.

An examination portal is developed, which will conduct the examination online and automate the process of examination and evaluation of a student by marking the paper just after the examination and updating the marks of the student.Intimation to the parents, concerned faculty and other people about the student is done through notifications.

In notifications, an e-mail is sent to the people about the students' attendance in a particular subject, overall performance and marks in the examinations. These notifications will also be sent to the students also. The aim of developing this system is to automate the tasks, which a student performs as part of an educational institution on a regular basis, as much as possible. This system also seeks to ease up the common processes; an institution does, as part of the student management system.

1.1 Motivation

The general system that has existed for years for storing student information in an educational institution was on paper and pen. With increase in use of computers for automating processes ranging from storing information to shopping, the need was felt to use this automation in education for storing student related data. Through this we cut down the paper record which takes enormous time for handling, and also reduce human effort. Also in case of manual system they need a lot of time, manpower etc. to write and maintain the data.

With time maintaining the consistency of data and handling of legacy data became the issue. Also in case of failure, backup of data also came into picture.

With these issues, student management systems came into picture where almost all work is computerized. Systems with user-friendly interface and properly structured programs with linked database systems came to access the data in an ordered fashion. Searching, inserting, updating, deleting of data became easy. So the consistency of data is maintained. Maintaining backup also became easy. It can do it within a few seconds.

But these systems were very static, in the sense that all the changes had to be done manually by a user of the system. The changes could not occur automatically. Also these systems were of use for only the administration. They were not of use to the student and were not accessible to them.

Thus the developed software through this project made a complete student system that manages student data, updates data dynamically, is accessible to the students, includes faculty. It encompasses everyone in an educational institution as a part of this system.

1.2 Scope

The scope of the project is the server of the institution and all the computers that are attached to the server. It is based on client-server architecture in a intranet of an organization, therefore it is accessible by any system on the intranet by authenticating the identity. Also certain modules of the system can be made online.

1.3 Feasibility Study

1.3.1 Economic Feasibility

The system being developed is economic with respect to an educational institution. Initial hardware cost of buying the biometric device will be little high. Also the number of biometric devices required will also depend upon the size of the institution. After installation, there is no additional cost and system is economic. It is cost effective as it eliminates paper records and also cuts the cost of people required into maintaining these records.

The system is also time effective as all the processes are done automatically and all the reports can be generated as and when required.

Also the results obtained contain minimum error and are highly accurate as all the calculations are automatic and there is no human-interferenee in taking the data.

1.3.2 Technical Feasibility

The system requires the installation of a central data server and biometric device. Driver for the device, which is available with the device, has to be installed on the server. Network wire connection need to be made between devices and server. No other software is needed to install for running the system.

1.3.3 Behavioral Feasibility

Upon installation, the working of the system is quite easy and the interface of the system is user-friendly for all the users of the system be it student, faculty or administrator. No special training is required for operating the system.

1.4 Technical Specifications

1.4.1 Technology Used

The technology used for developing the system is ***PHP***. Basic HTML and JAVASCRIPT is used for creating the interface of the system. PHP is the server-side scripting language.

Back-end database is made in **MySql**.

Windows Apache (Wamp) Server is used as a proxy-server to simulate the client-server architecture.

To incorporate every tool into one integrated tool, **EasyPHP** has been used.

EasyPHP is a software package including PHP as the server-side scripting language, Apache web server, the SQL server MySQL, as well as development tools such as the database manager PhpMyAdmin, the debugger Xdebug and many others.

1.4.2 System Requirements

Operating System: Windows Xp/ Vista/ 7

Browser: Any modern browser such as Google Chrome, Safari, Mozilla Firefox, Internet Explorer 8 & above

Computer of minimum RAM size 512 MB, Hard Disk 60GB and Dual Core processor is advisable for better performance results.

1.5 Organization of Report

The report is organized into different chapters.

Chapter 1 introduces the project and explains the objective of the project, the motivation behind using the project. It entails the feasibility of the system and the technical requirements and specifications.

Chapter 2 describes the approach of the project, various modules in the project and the user specifications and privileges. Various diagrams show the flow of the project and the users and outputs of the project. Database design of the system is also been mentioned.

The following chapters explain each of the modules in detail.

Chapter 3 explains the LOGIN AND STUDENT DETAILS module, the flow of data and the working of the module.

Chapter 4 explains the COLLEGE AND DEPARTMENT DETAILS module, its use and scope.

Chapter 5 gives details about the EXAMINATION PORTAL. Various processes in this portal are specified and role of different users using this module is shown.

Chapter 6 introduces the ATTENDANCE AND NOTIFICATIONS module. Various issues in the ATTENDANCE module are discussed and their solutions are discussed. This is the last chapter explaining the last module of Rifleeso. This module is of NOTIFICATIONS sent to the various people.

Appendix is given in the end giving the codes for various modules provided in order to show the working of the project.

References to different books and online content are given at the last.

CHAPTER 2

2. APPROACH

The system includes biometric device as an input device and forms the basis of automation in the whole website. Therefore biometric device will always be directly or indirectly linked to very part of the system.

In order to simplify the system and differentiate the various tasks of the system with respect to the biometric device, as linked to the tasks, the system is developed into various modules that are linked to each other.

Certain modules can also be put online on the web that can be accessed through the internet. These modules would not be directly linked to the biometric device but will access the data stored through device.

Considering all these things the project has been developed in 5 modules that along with biometric device will complete the system.

The five modules are

- **College and Department details**
- **Login and Student details**
- **Attendance**
- **Examination Portal**
- **Notifications**

All of these modules are linked with each other and access and update the same database. Some of these modules directly use the biometric device for input device and authentication while some of these modules only access the data stored through the device.

- **College and Department details**

This module deals with the college details and different departments in the college. It contains all the details of the faculty members. This module is not linked with the biometric device and is there to maintain the integrity of the students' academic information. All the subjects' details of a particular student will be taken through this module.

- **Login & Student details**

This module is for accessing student's own data by logging into his/her own account. This module is designed so that a student can see his progress over the time in a particular course. This module will access the data stored by the biometric device and does not use the biometric device directly.

- **Attendance**

This module deals with keeping track of the students' attendance and storing it in the system. This module uses the biometric device as the input device for storing the attendance by matching the fingerprint of the student and authenticating his/her identity.

- **Examination Portal**

This is the module having an online exam portal where examinations are conducted on the systems and results are computed automatically. Regular assignments can also be uploaded by the faculty to the students. Biometric device will be used to authenticate the id of the student in the examination.

- **Notifications**

This module deals with the notifications that are made to the faculty, guardians, registrar, etc. If the attendance or marks are below the required criteria, there will be a notification sent to all mentioned above. This module sends these notifications based on the data stored by the biometric device.

2.1 Methodology

The users of the system are differentiated on the basis of their privileges and their level of authorization to access the data.

There are 3 types of users

1. **Student:** Students can only see their data.
2. **Faculty:** Faculty can see the student's data and his/ her own. A faculty member can update the marks of a student. He also can upload any assignment related to his subject.
3. **Administrator:** The administrator or admin plays a very important role in the system. It has got access to all the data and is responsible for consistency and integrity of data. He will get the student registered. He will put the paper for examination online and is the only one to access it. He also takes care of updating the attendance regularly.

2.2 Context Diagram

The diagram below shows the inputs of the system and the users of the system giving the input. The outputs of the system are mentioned and the people receiving those output. The biometric

fingerprint device is a key input device in the whole system that will be used by the student for authenticating his identity.

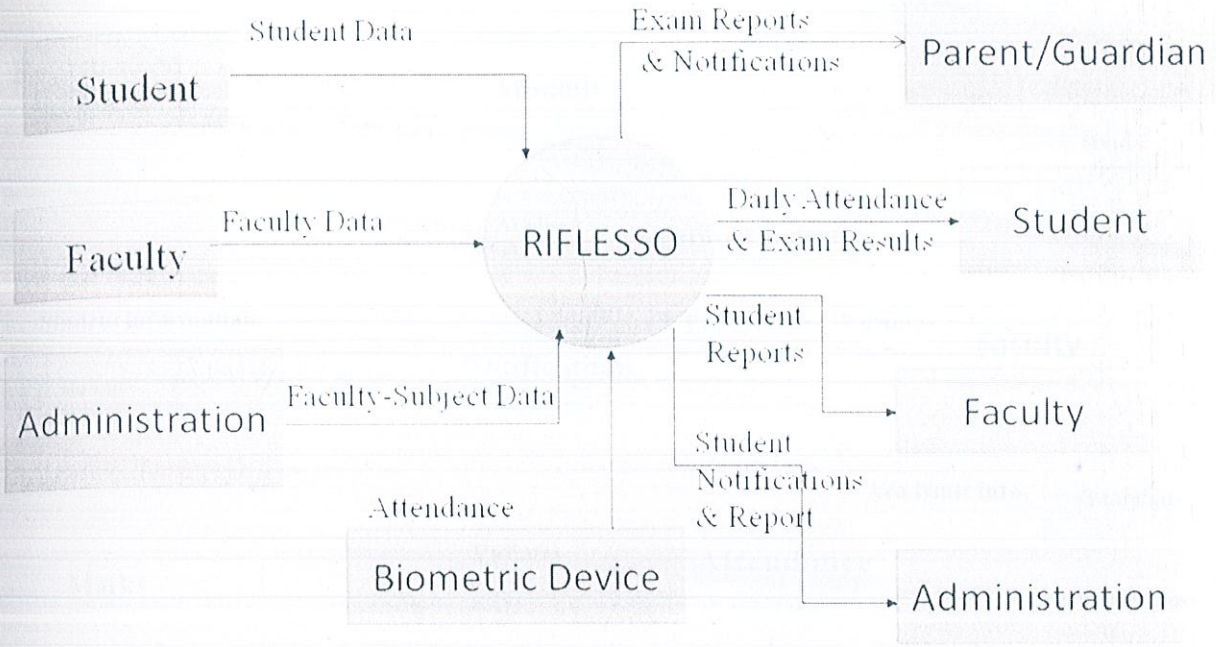


Fig. 2.1 Context Diagram

2.3 Data Flow Diagram

The data flow diagram shows the major processes of the system and the flow of data among these processes. The five major processes are

1. Registration: The student registers by filling in all the personal information and registering the thumbprint through the biometric device. Upon successful registration a numeric id is generated by the biometric device which becomes the user-id of the student. All this information is stored in student database.

2. Faculty Information: The faculty also has to feed in its information such as name, department, subject etc. Upon this the subjects are assigned to the student as per his branch.

3. Attendance: The students enters the attendance through the biometric device where the fingerprint is matched and corresponding values of id's are updated which gets stored in the student database.

4. Examination: Examinations are conducted through the examination portal and the marks are calculated after the exam and get updated on the student database.

5. Notifications: The notifications are sent to the concerned people of a student where addresses are taken from the student and faculty database respectively.

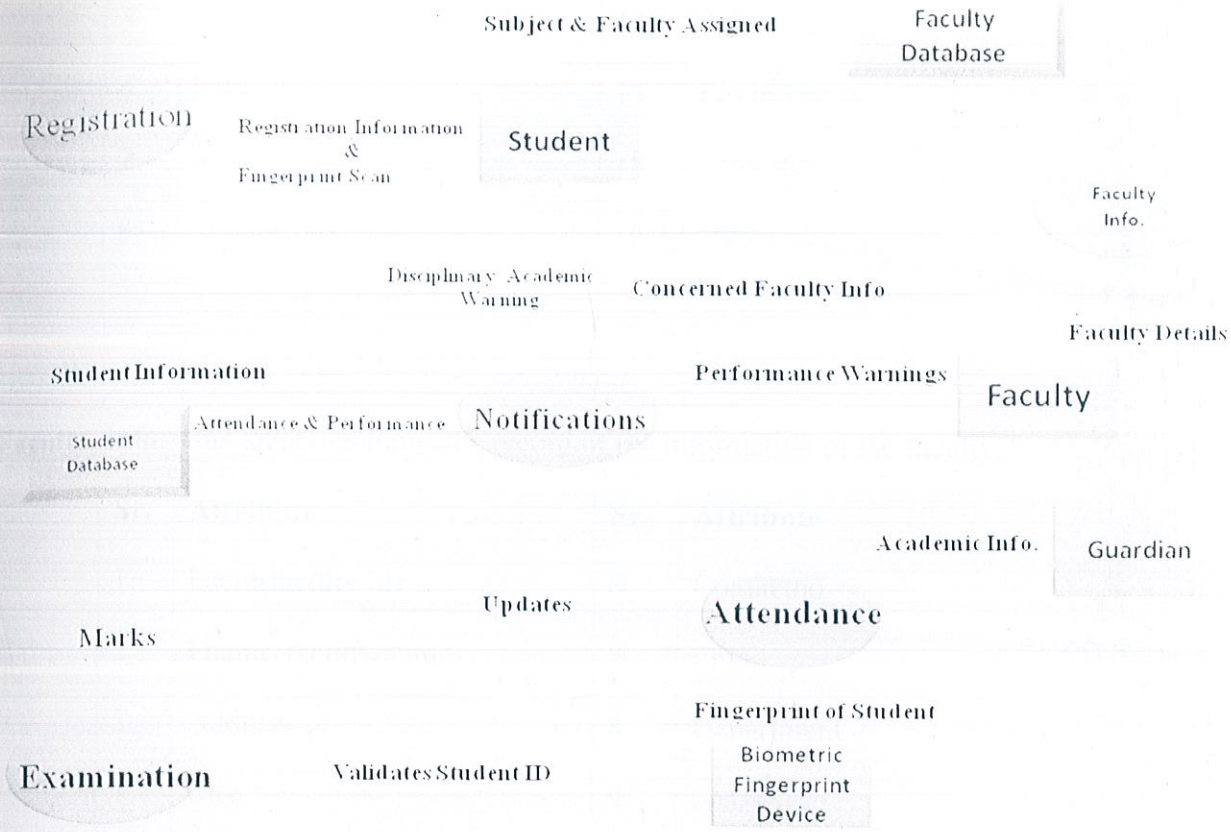


Fig. 2.2 Data Flow Diagram

2.4 Database Design

Riflesso contains two databases 'student' and 'attendance'. There are several different tables that will keep the records of the data entered in the database.
The different tables in the database student are:

Student_info: This table takes in the data from the student at the time of registration. The structure of the table is:

Sr.	Attribute	Sr.	Attribute
1	Student_name	9	Email
2	Father_name	10	Pemail(parent email)
3	Mother_name	11	Tenth(marks)

4	Dob(date of birth)	12	Twelth(marks)
5	Address	13	Password
6	City	14	Logintype
7	State	15	Semester
8	Phone_no	16	Id

Here ID is the primary key.

Faculty_info: This table maintains the record of the information of the faculty.

Sr.	Attribute	Sr.	Attribute
1	Facid(faculty id)	6	Contactno
2	Fname(faculty name)	7	Email
3	Address	8	Department
4	City	9	Fpassword
5	state	10	Logintype

Facid is the primary key.

Login_info: This table helps maintaining the administrator login details.

Sr.	Attribute	Sr.	Attribute
1	id	3	Logintype
2	password		

The primary key is id.

Subject: The information regarding the subjects is entered to this table.

Sr.	Attribute	Sr.	Attribute
1	Subid(subject id)	3	Subdesc(subject description)

2	subname(subject name)	4	Semester

The primary key is Subid.

Test: The date of exam, subject, timing, etc. is added to this table.

Sr.	Attribute	Sr.	Attribute
1	testid	5	Duration
2	test name	6	Totalquestions
3	testdesc	7	Testsecretcode
4	testdate	8	Testtime

The primary key is testid.

Question: After the test frame has been given, questions are added to the test through this table.

Sr.	Attribute	Sr.	Attribute
1	testid	6	Option
2	Qnid(question id)	7	Optiond
3	question	8	Correctanswer
4	optiona	9	Marks
5	optionb		

The primary key is qnid.

The foreign key is testid from the table test..

Student_test: when the student takes the test , his tabulated marks are added to this table.

Sr.	Attribute	Sr.	Attribute
1	id	3	Finalmarks

2	testid

The primary key is testid.

The id is the foreign key.

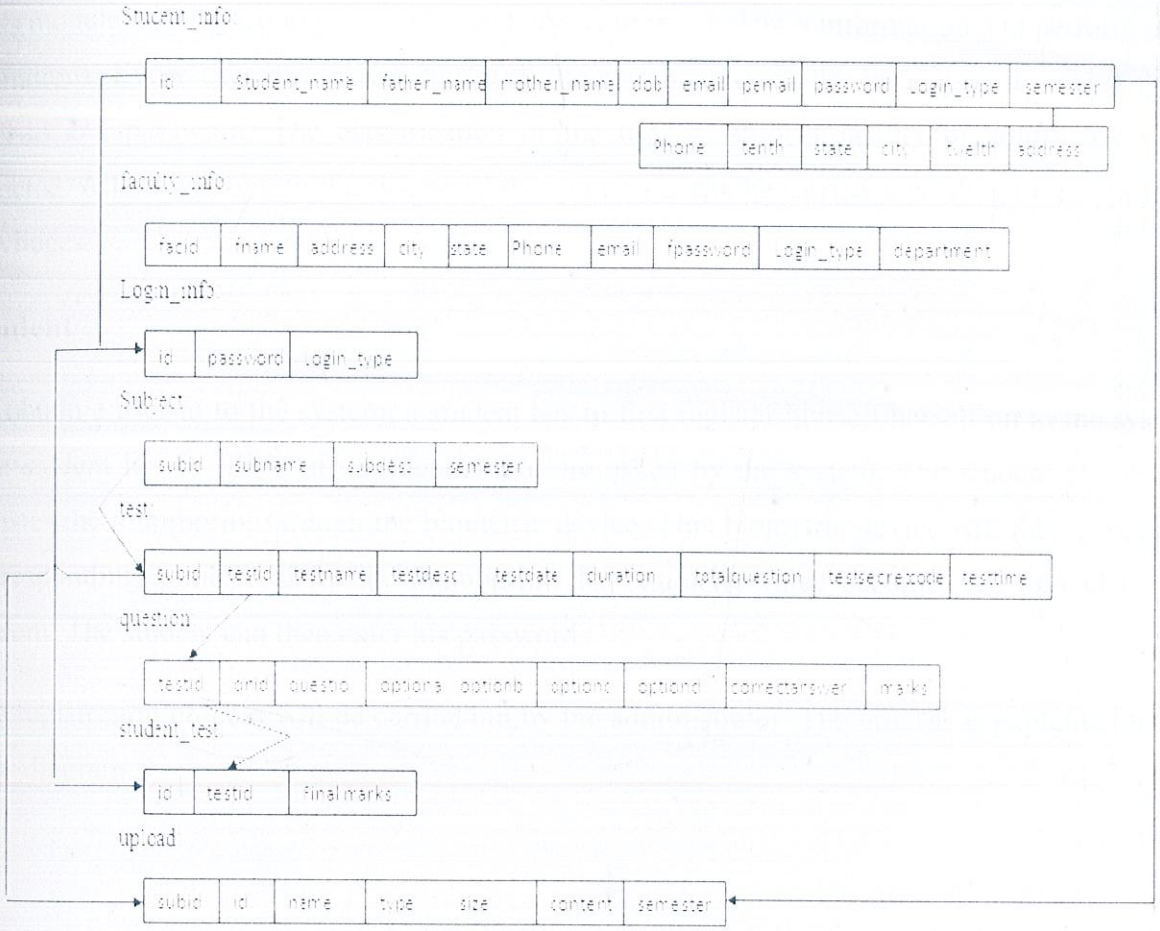
Upload: The assignments uploaded for the students are stored in this table.

Sr.	Attribute	Sr.	Attribute
1	subid	5	Size
2	Id	6	Content
3	name	7	Semester
4	Type		

The primary key is id.

Subid is the foreign key here.

Mapping to relations:



CHAPTER 3

3. LOGIN AND STUDENT DETAILS

This module is designed to give authorized access to the student's information and perform other operations. Every user has to first log in to the system through the login page by entering his user-id and password. The classification of the user as student faculty or admin has to be selected. After logging into the system further processes can be carried according to the specified privileges.

Student

To obtain a user-id to the system, a student has to first register himself/herself on to the system. The student has to fill in all the details that are asked by the system. The student also has to register the thumbprint through the biometric device. This biometric device will then generate a corresponding numeric id for the thumbprint. This numeric value becomes the user-id for the student. The student can then enter his password.

The registration process will be carried out by the administrator. The process is explained by the flow diagram

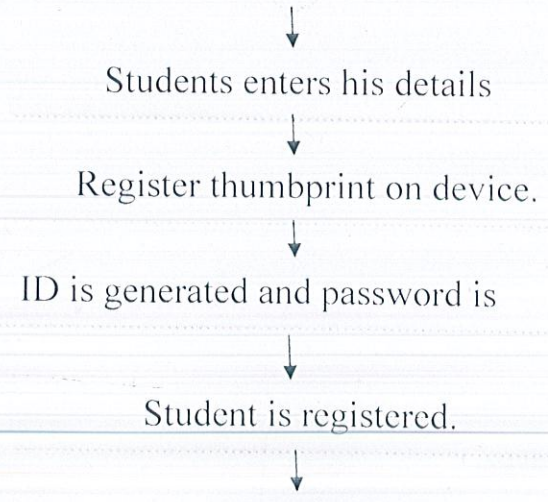


Fig. 3.1 Flowchart of registration

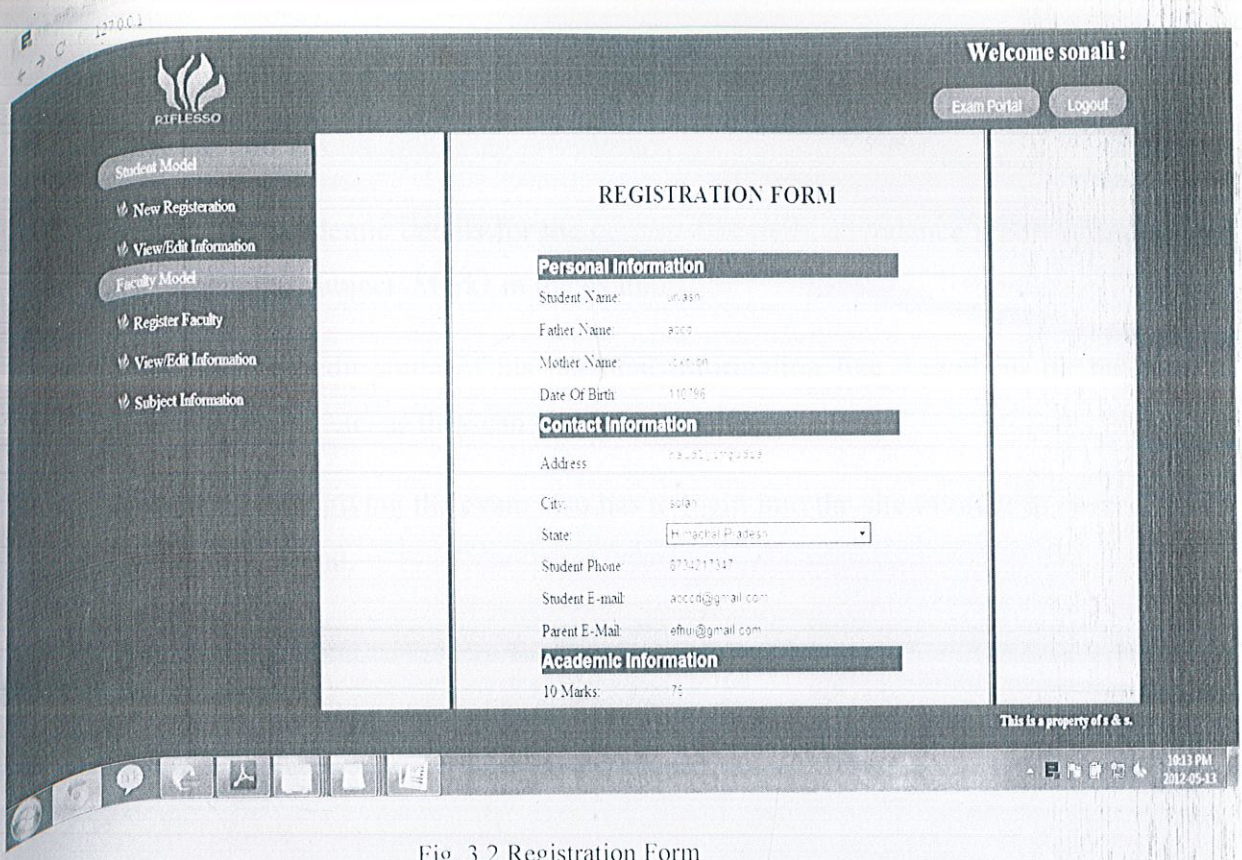


Fig. 3.2 Registration Form

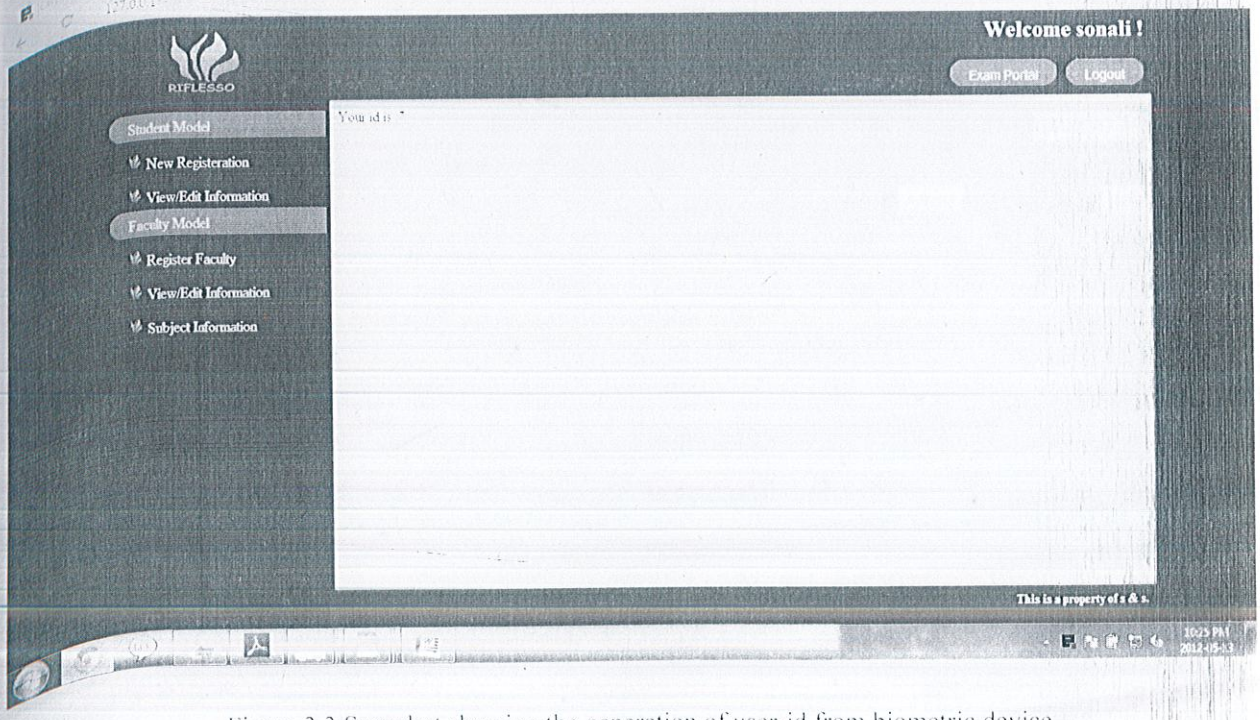


Figure 3.3 Snapshot showing the generation of user-id from biometric device.

The student can now login by login details and selecting login type as student. Fig. 3.3 shows

Privileges of Student

The student as the user has the following privileges:

- View his/ her academic details for the course. The daily attendance report of the student for a particular subject. Marks in the examination.
- The student can edit some of his personal information like e-mail id, phone number, parents e-mail id etc. as they can change with time.
- The student while giving the exam also has to login into the site in order to proceed to the examination portal.

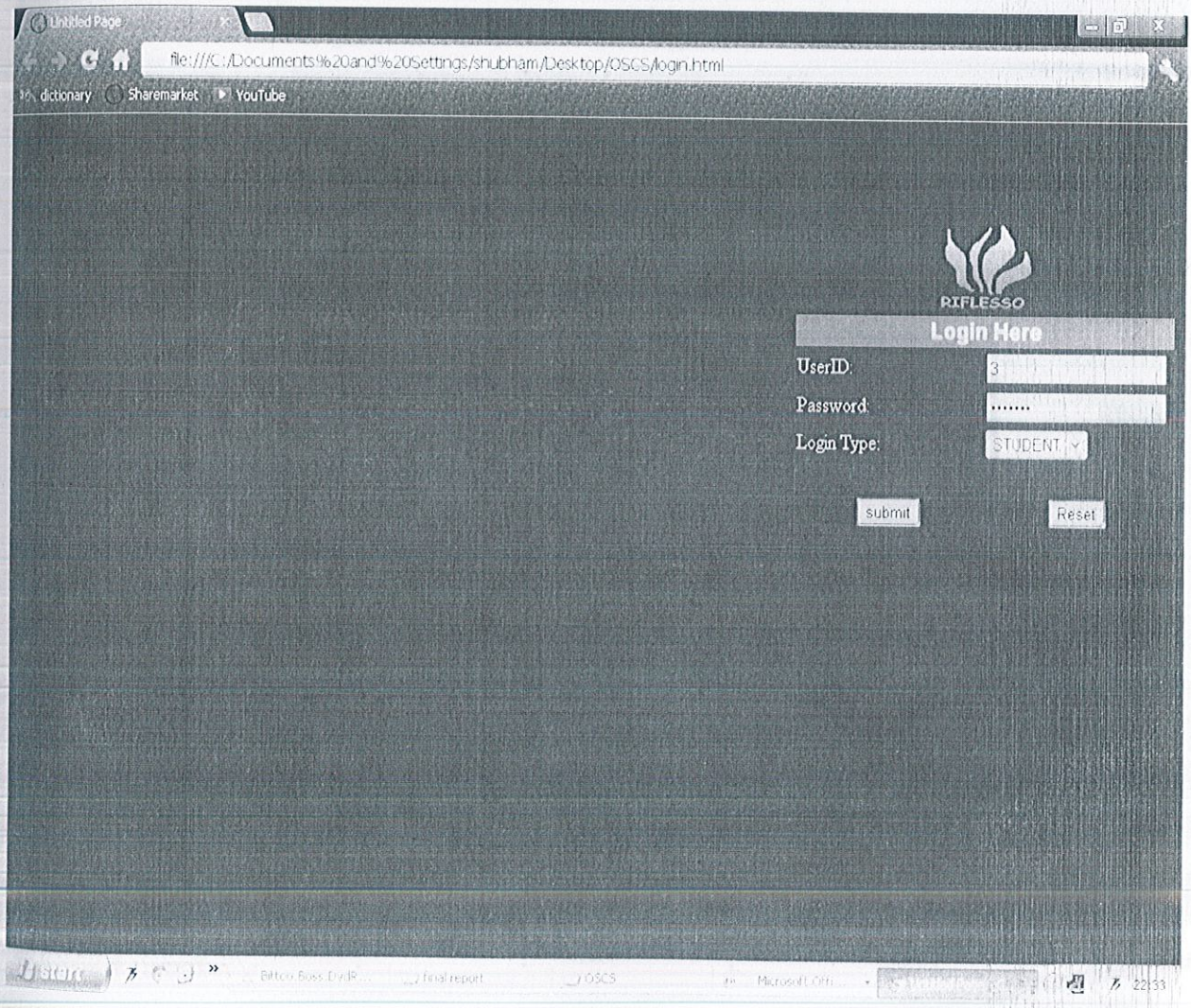


Fig. 3.4 Login as student

Faculty

Every faculty member can login in to the system by providing their login information and selecting the login type as faculty.

Privileges to faculty

- View student information.
- Give internal marks.
- Upload assignments for the students.
- View and update his/her information.

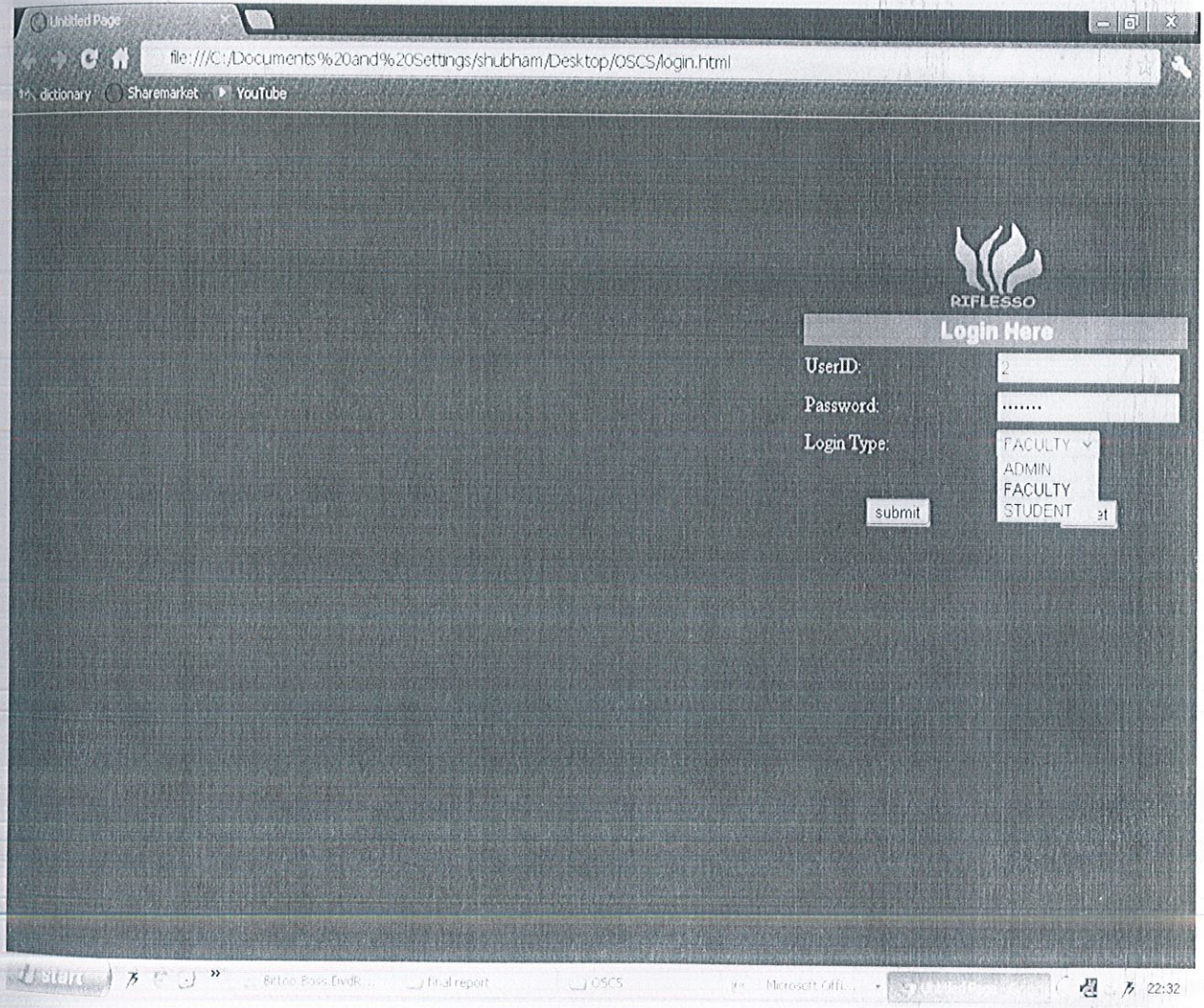


Fig. 3.5 Login as Faculty

Administrator

The administrator will also login the same as student and faculty. Being an administrator does not allow him to not pass the authentication.

Privileges to Admin

Admin doesn't require any authorization as all the system is run by the admin. From registering a candidate to setting up of an exam paper, admin has all the privileges to him.

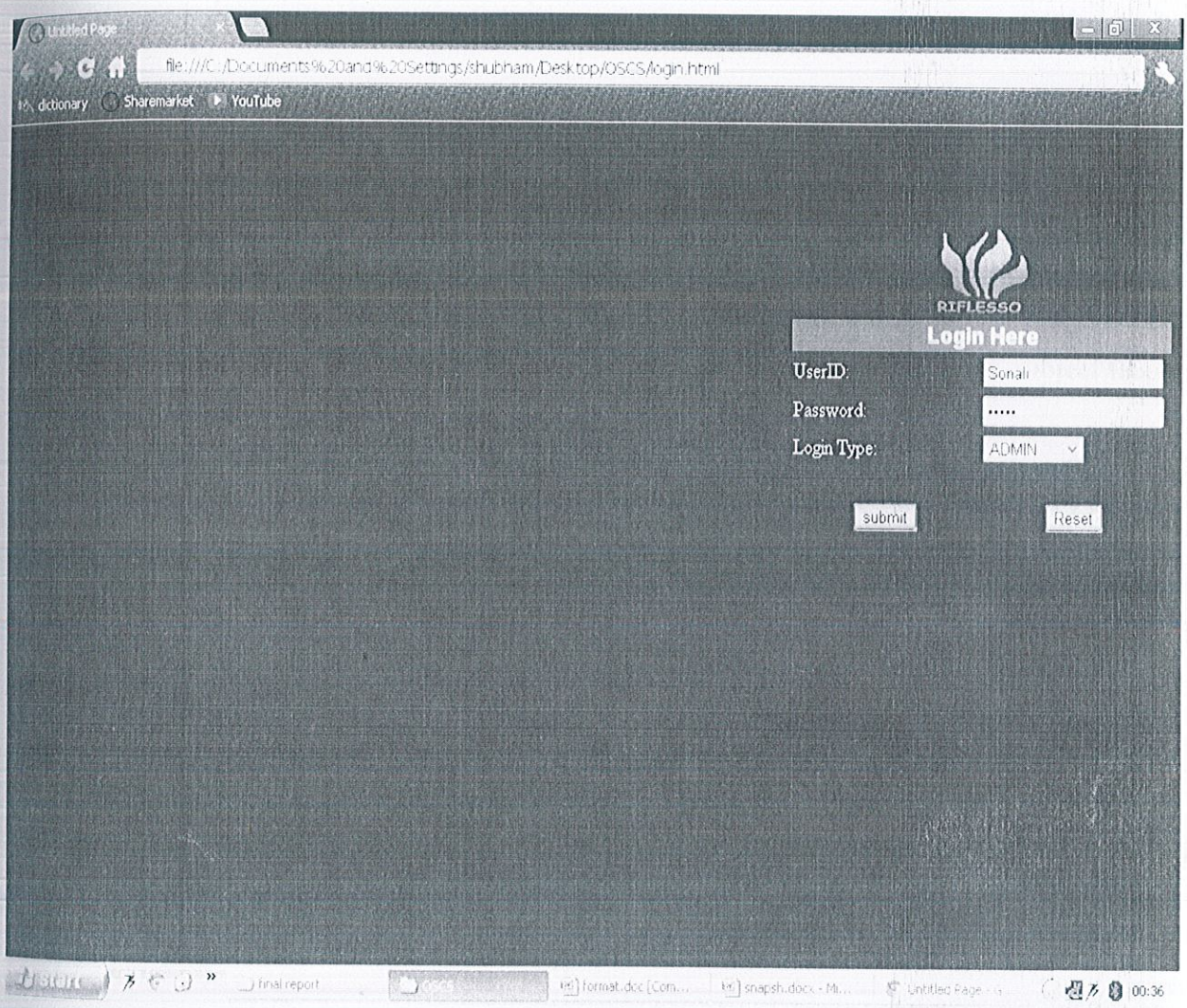


Fig. 3.6 Login as Administrator

CHAPTER 4

4. COLLEGE AND DEPARTMENT DETAILS

This module is designed to access and manipulate the data regarding the college academic built-up. The functions of this module are

- Register a faculty member.
- Add/ Delete the different departments in the college.
- Register, modify or delete subjects in a particular department.

Register a faculty member: The faculty member is registered in order to obtain the details of the teacher such as name, department, subject (he/she is teaching), e-mail id (for notifications) and other related details. This information is also necessary when registering the subject taught by a faculty member.

Registering the faculty member also provide him/her with a user-id in order to access the system.

As in case of student, faculty can also edit some of his/her information stored in the faculty table.

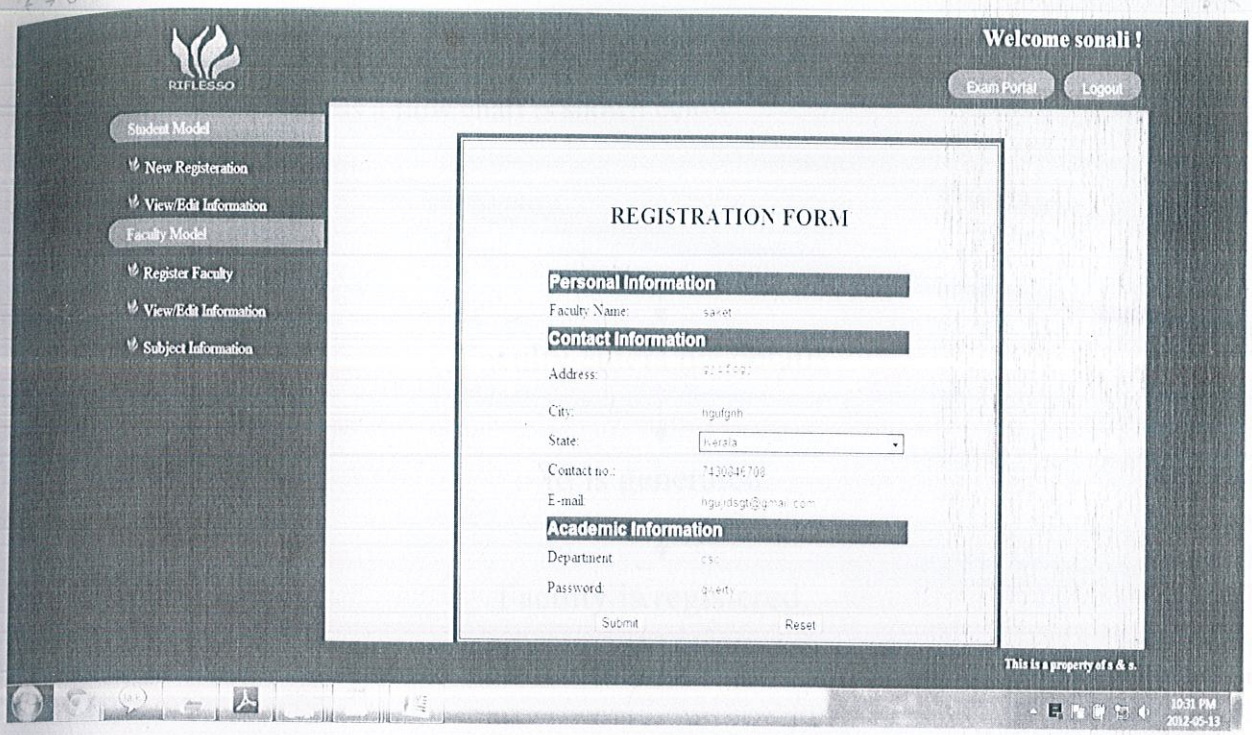
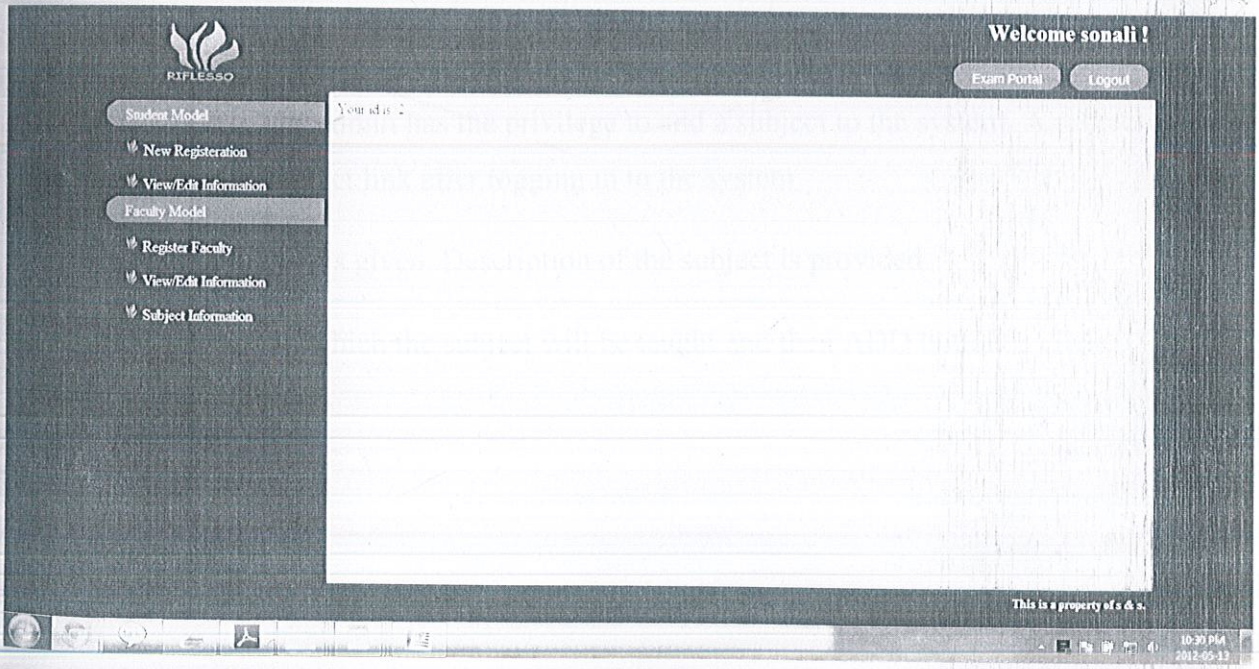


Fig. 4.1 Regiseration of Faculty



.Fig. 4.2 Generation of facult- id

Registration of faculty member is also done by the administrator. The registration process of a faculty member does not require the use of biometric device.

To clarify the whole process a flow chart is shown below

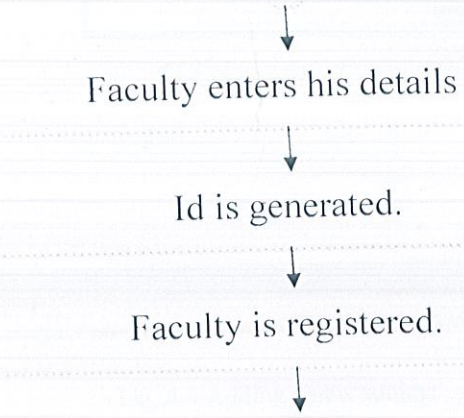


Fig. 4.3 Flowchart of faculty registration

Adding a subject: The admin has the privilege to add a subject to the system. A subject is added by going to the add subject link after logging in to the system.

The name of the subject is given. Description of the subject is provided.

Semester is selected in which the subject will be taught and then ADD button is clicked.

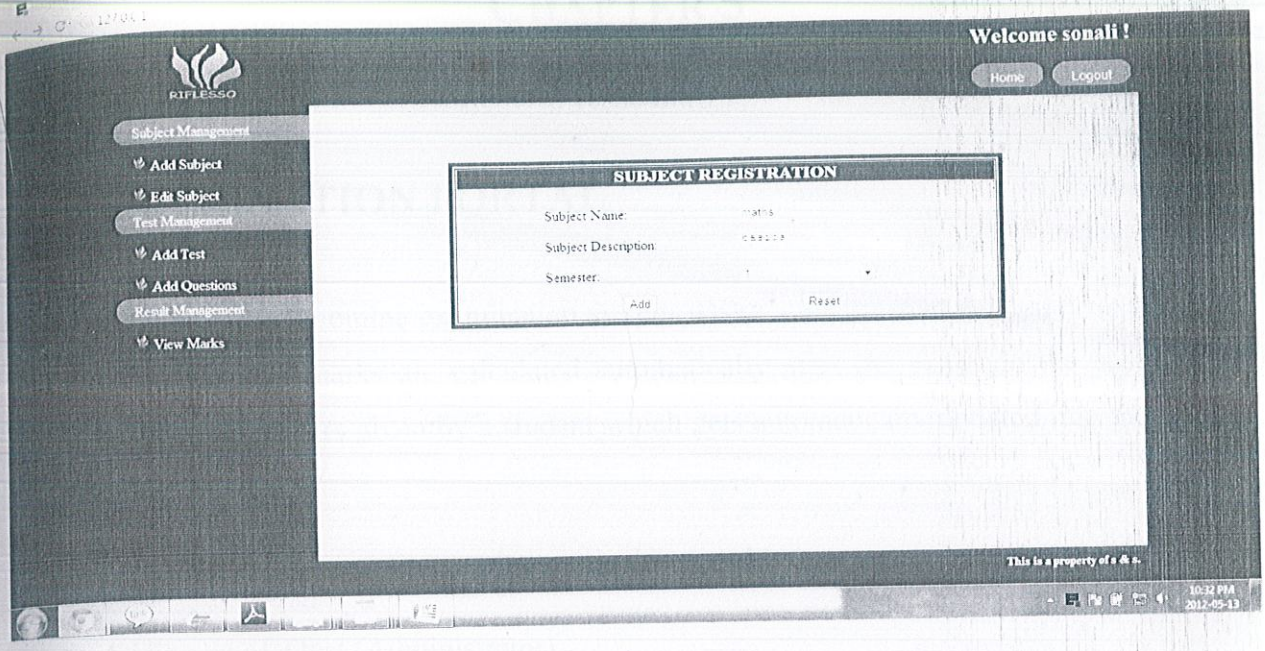


Fig. 4.4 Adding a new subject

On successful addition of the subject, a message is displayed on the screen.

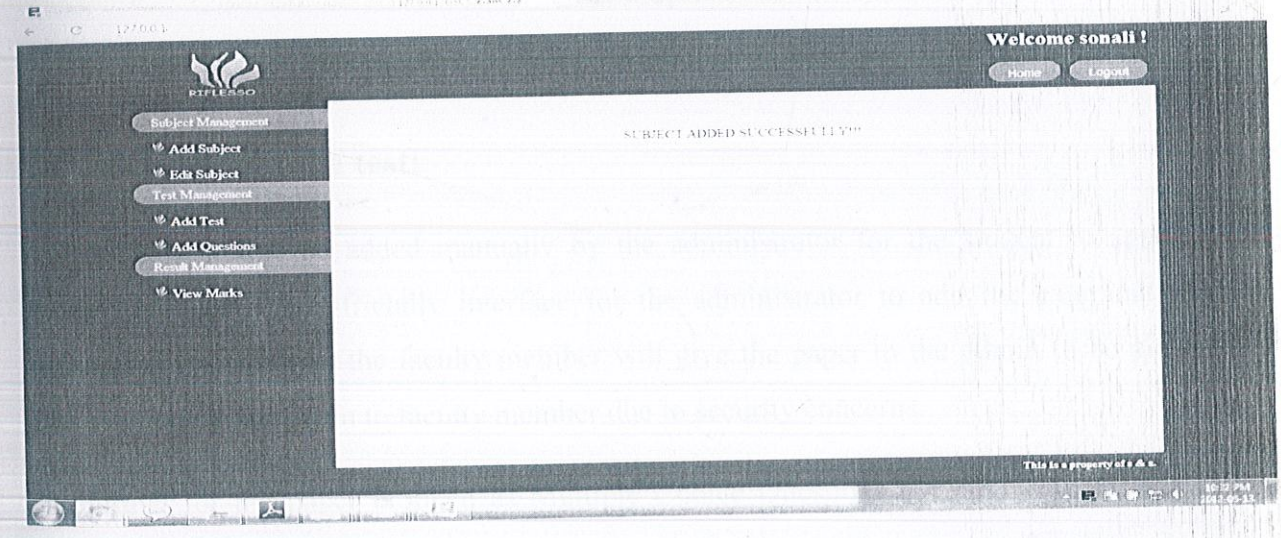


Fig. 4.5 Message displayed on successful adding of a new subject

The subject registered also gets a subject id that helps in linking the subject with the faculty id of the faculty member who is teaching the subject.

This subject id will also be used when an assignment is uploaded for a particular subject.

CHAPTER 5

5. EXAMINATION PORTAL

Examination portal is an online examination system where students give the exam for a subject on the computers and marks are calculated automatically after the exam on the basis of the correct number of answers given by a student which gets automatically updated into the student database.

Following are the processes being carried out in this portal

- Adding up of a test (Administrator)
- Giving the test (Student (Biometric Device provides the authentication to the student's id)
- Uploading of an assignment for a subject (Faculty)

5.1 Adding the test:

The test will be added manually by the administrator for the student to appear. The system provides a user-friendly interface for the administrator to add the examination. The assumption is made that the faculty member will give the paper to the admin to be put online. This privilege is not given to faculty member due to security concerns.

The format for the paper is taken as Multiple Choice Questions (MCQs) with each question having 4 options. Out of these four options, only one is the correct answer.

Subsequent series of images show the process:-

- The subject for which the test is made is selected.

Here, it is shown as JAVA

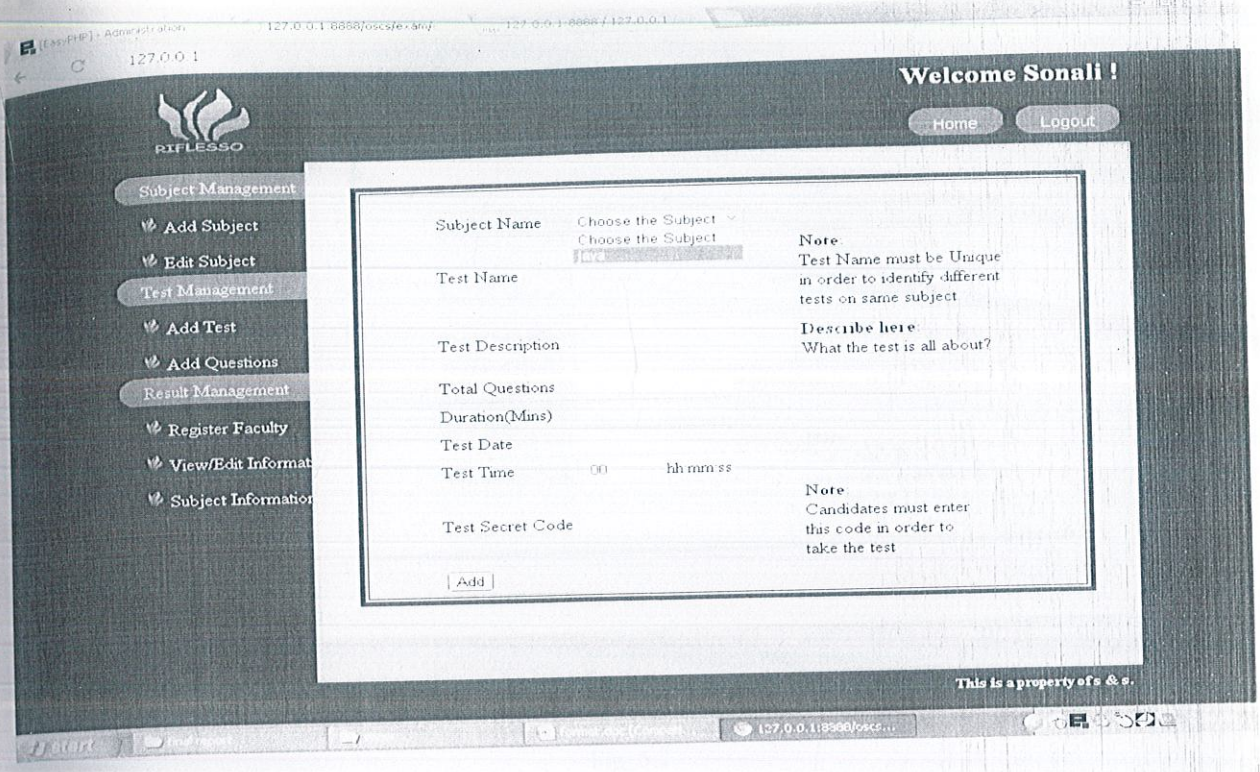


Fig. 5.1

- The name of the test is entered and a brief description of the test is provided.

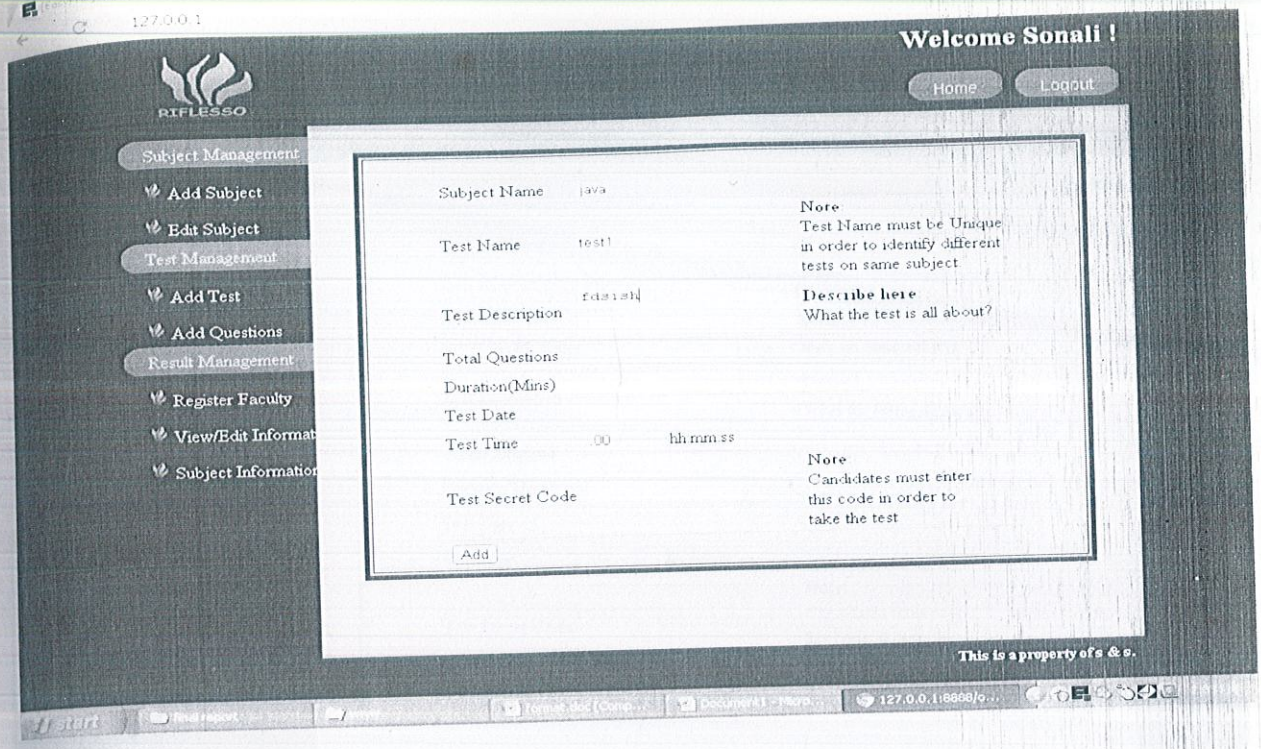


Fig. 5.2

- The number of questions in the exam is provided. The process for adding the questions to the exam checks this value and lets you enter the number of questions as specified before and no less or more than that.
- The duration of the project is also specified in minutes. The examination timer will take this value for its countdown timer.

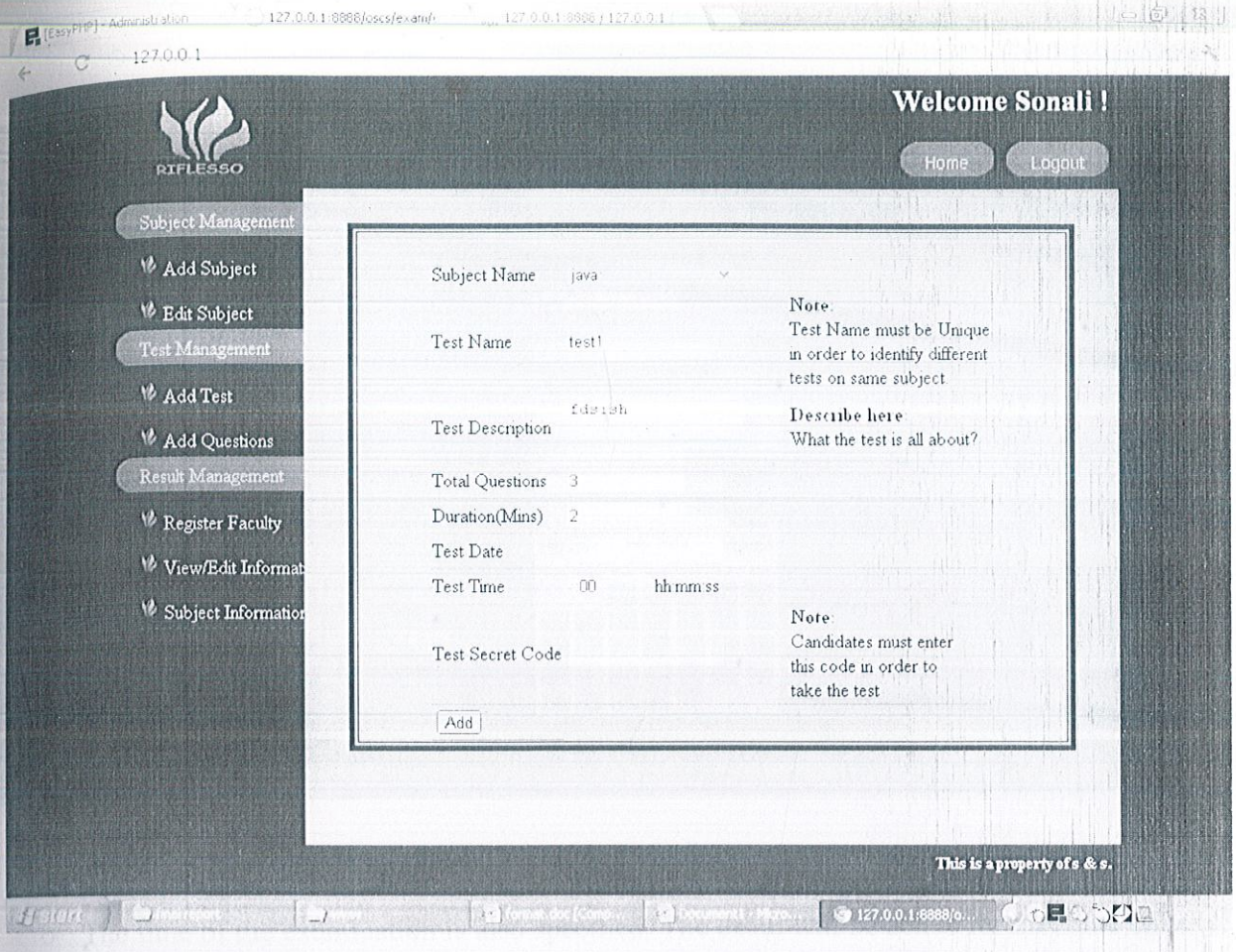


Fig. 5.3

- The date of the test is selected from the calendar. The exam will be active only on this date.

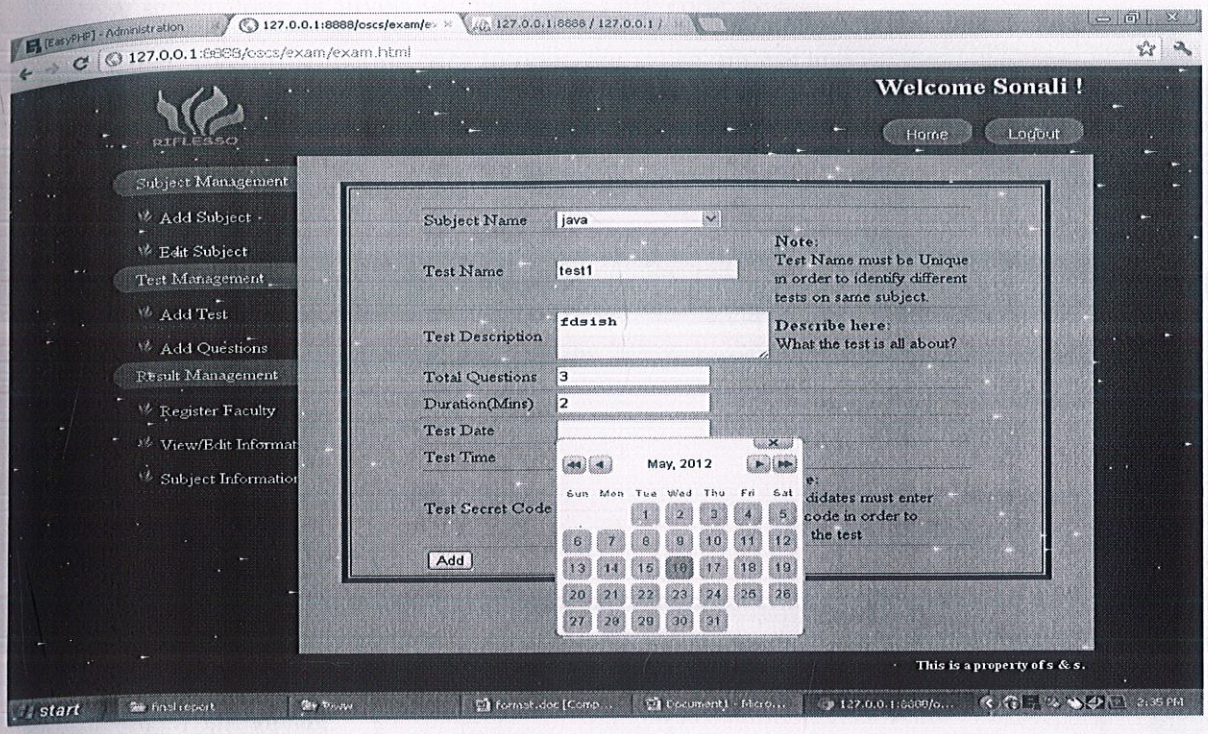


Fig. 5.4

- The time of the examination is also specified. The exam will be active for that time only. Any student who has not started the examination until that time will not be able to take the test.

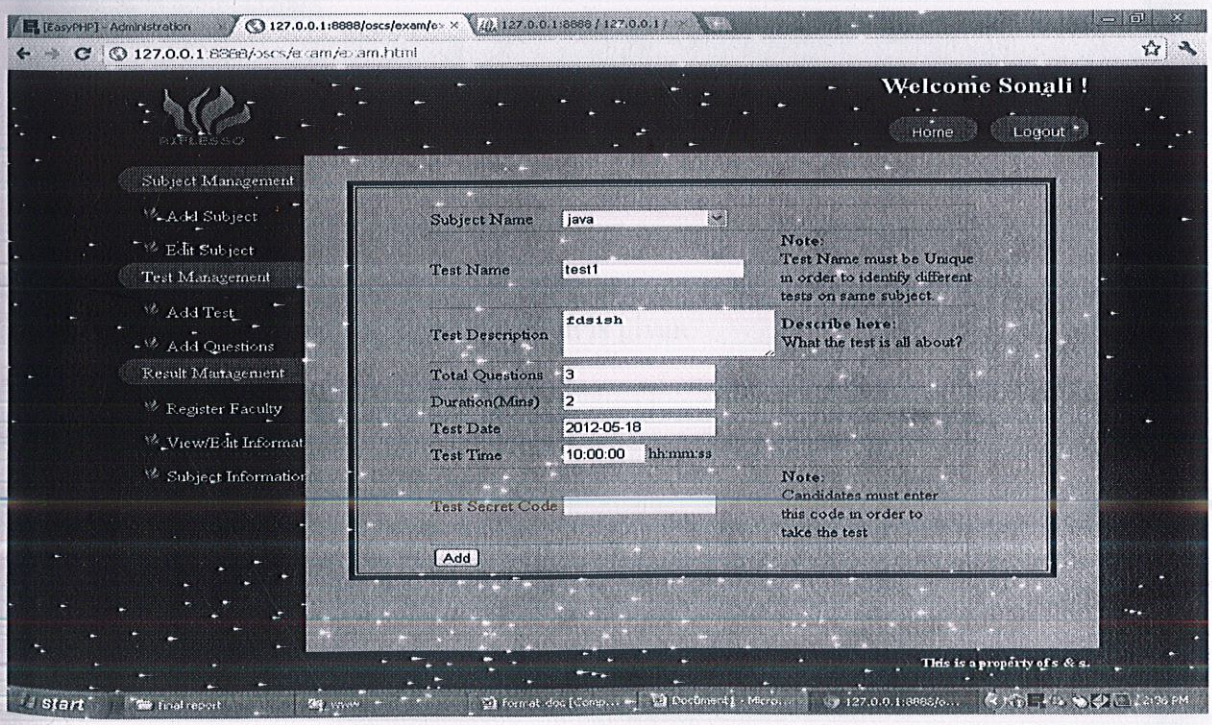


Fig. 5.5

- A secret code is given to each test. This code is given to increase the security of the paper. Even if someone tries to open the exam, unless and until the secret code is entered, the exam will not open.

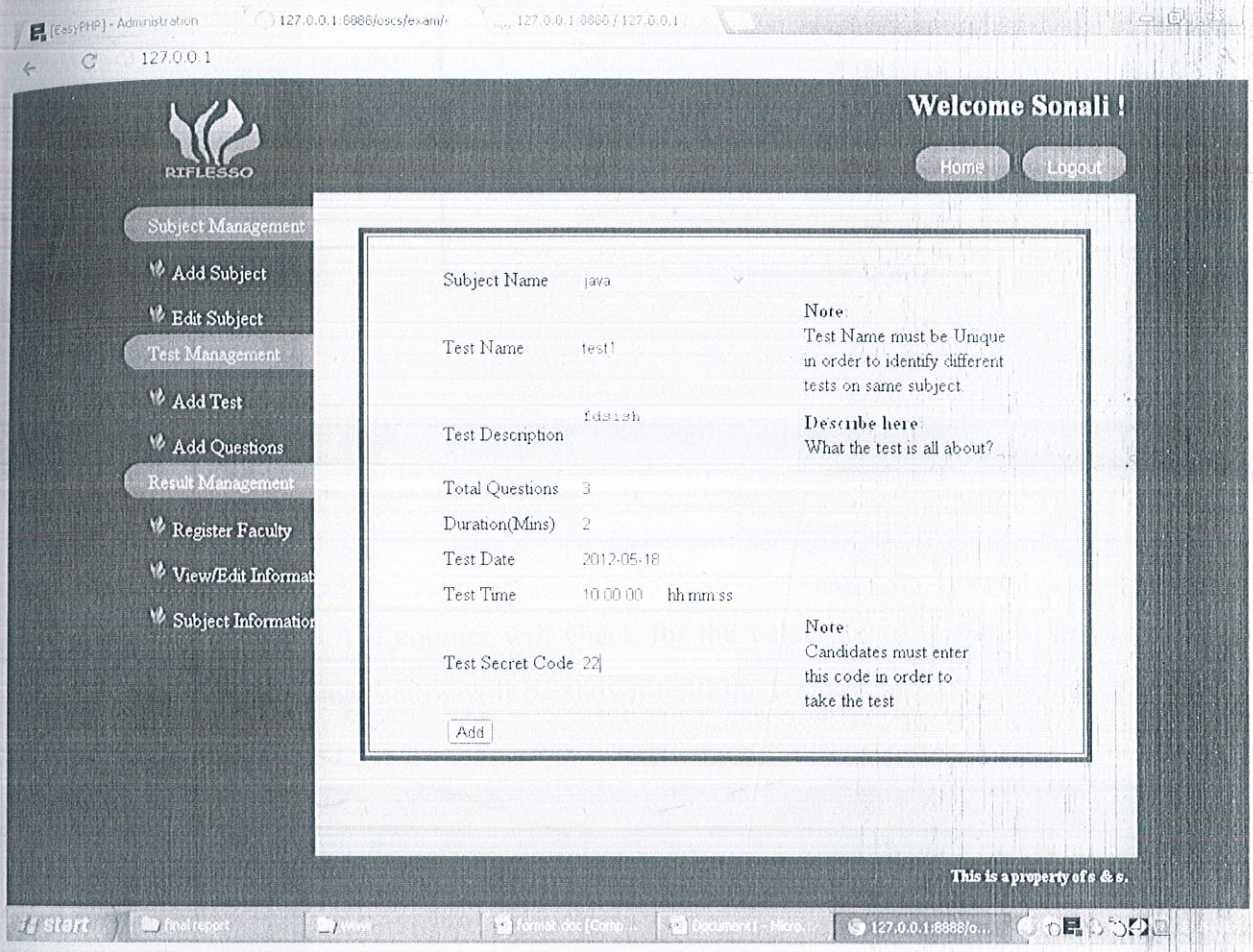


Fig. 5.6

- Now questions are added.
  - The statement of the question is given.
  - Then values for the 4 options are given.
  - The correct option is mentioned.
  - Marks allotted to each question are provided.

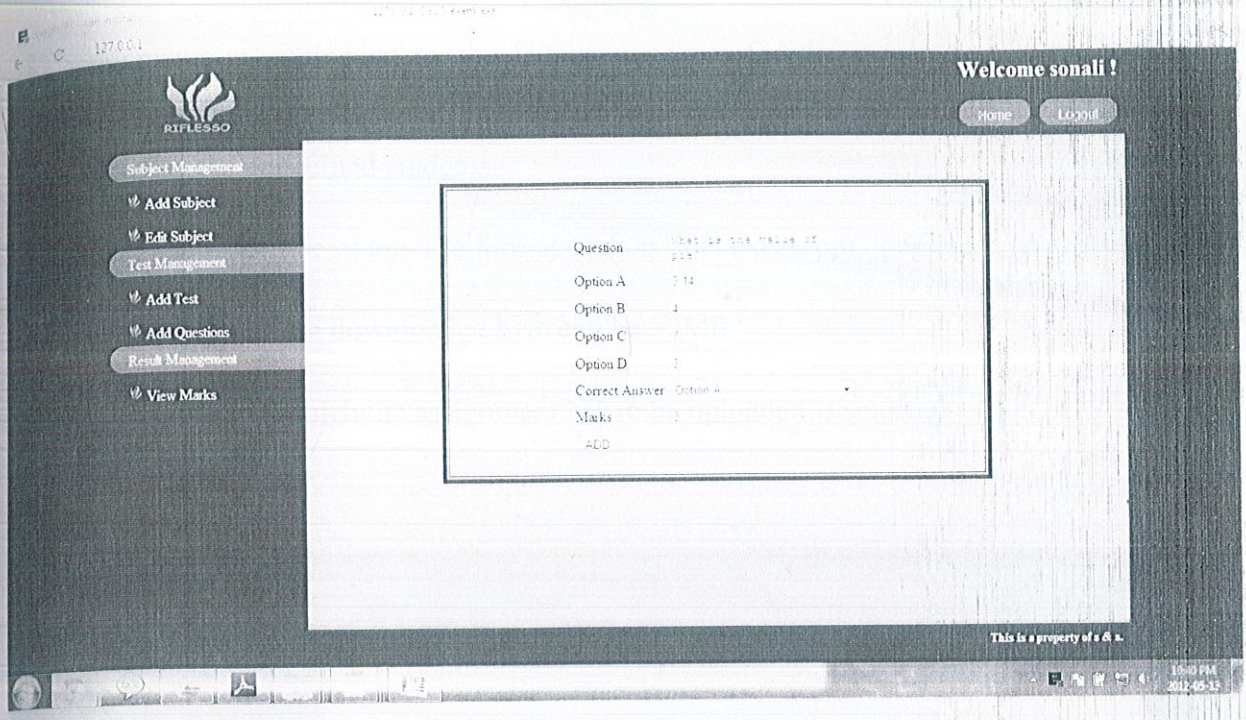


Fig. 5.7

After a question is added, the counter will check for the value no. of questions and if it's less than that value then the page below will be shown until the counter reaches its specifgied value.

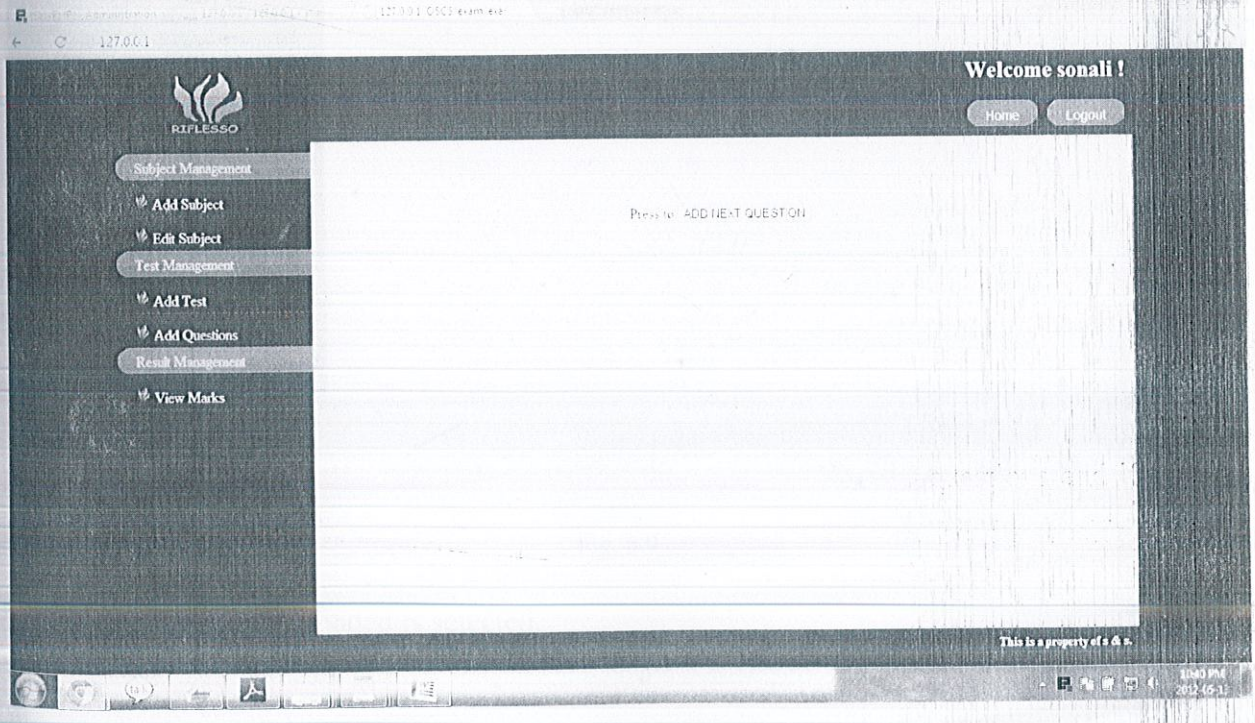


Fig. 5.8

5.2 Uploading the Assignment

The assignments are uploaded by the faculty members for the students electronically on the system. These assignments after the upload are available to the students, which can be downloaded by the concerned students.

These assignments can be of any file format such as doc, excel, ppt, jpeg etc.

The maximum size of the downloaded byte can be ~2Mb.

- The subject in which the assignment has to be uploaded is selected

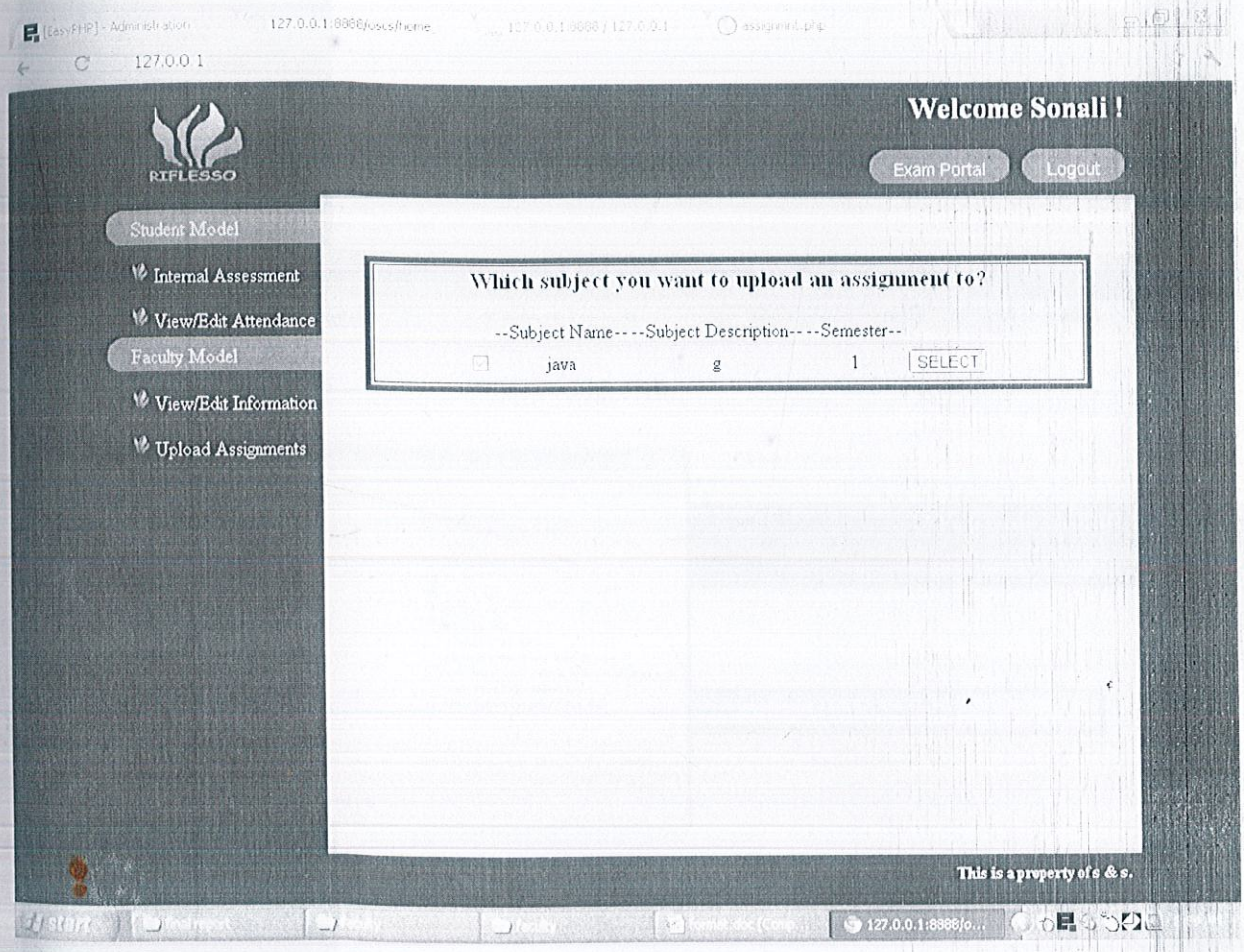


Fig. 5.9

- The file to be uploaded is selected.

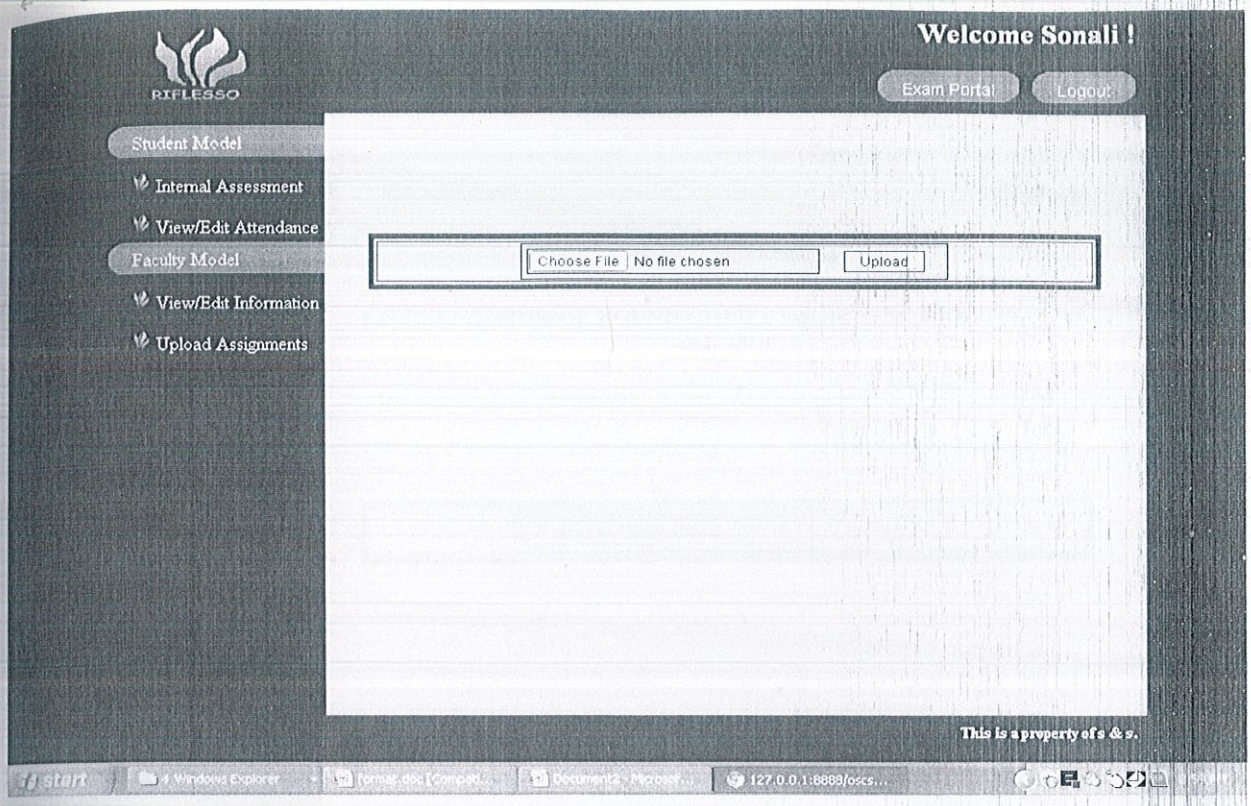


Fig. 5.10

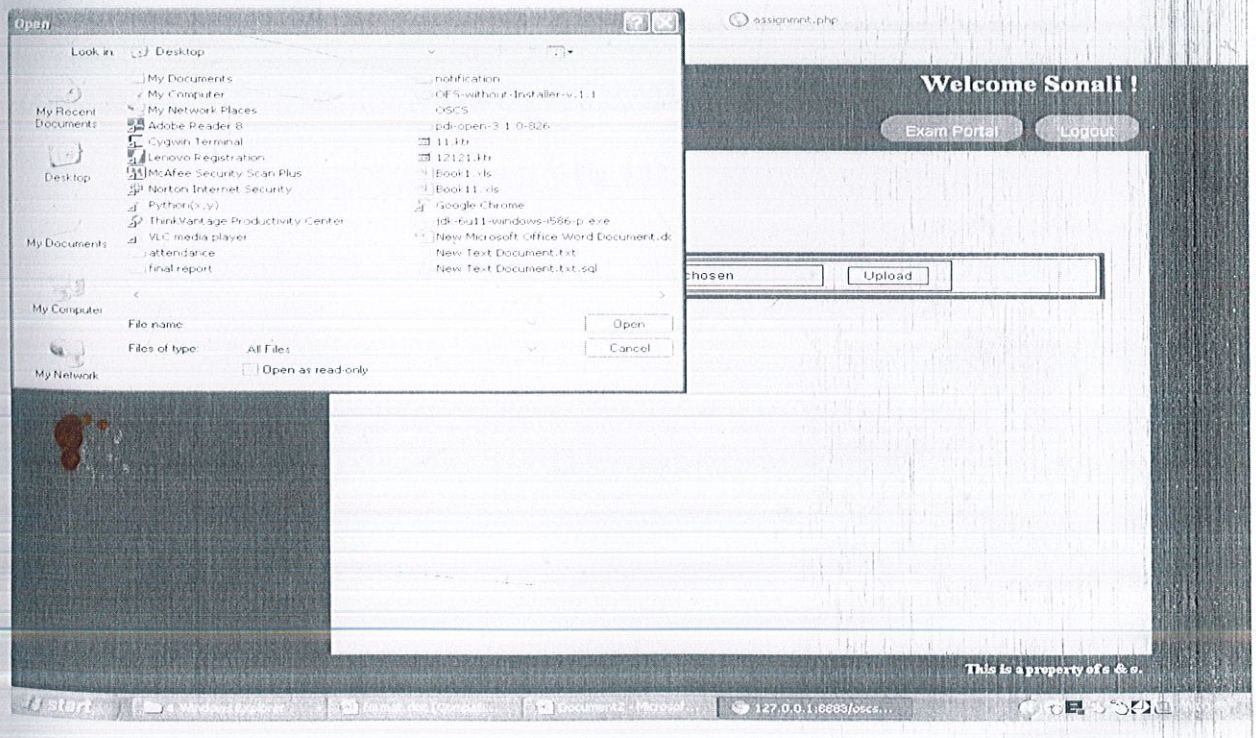


Fig. 5.11

If all the specifications of the uploaded file are within the specified parameters then a success message is displayed.

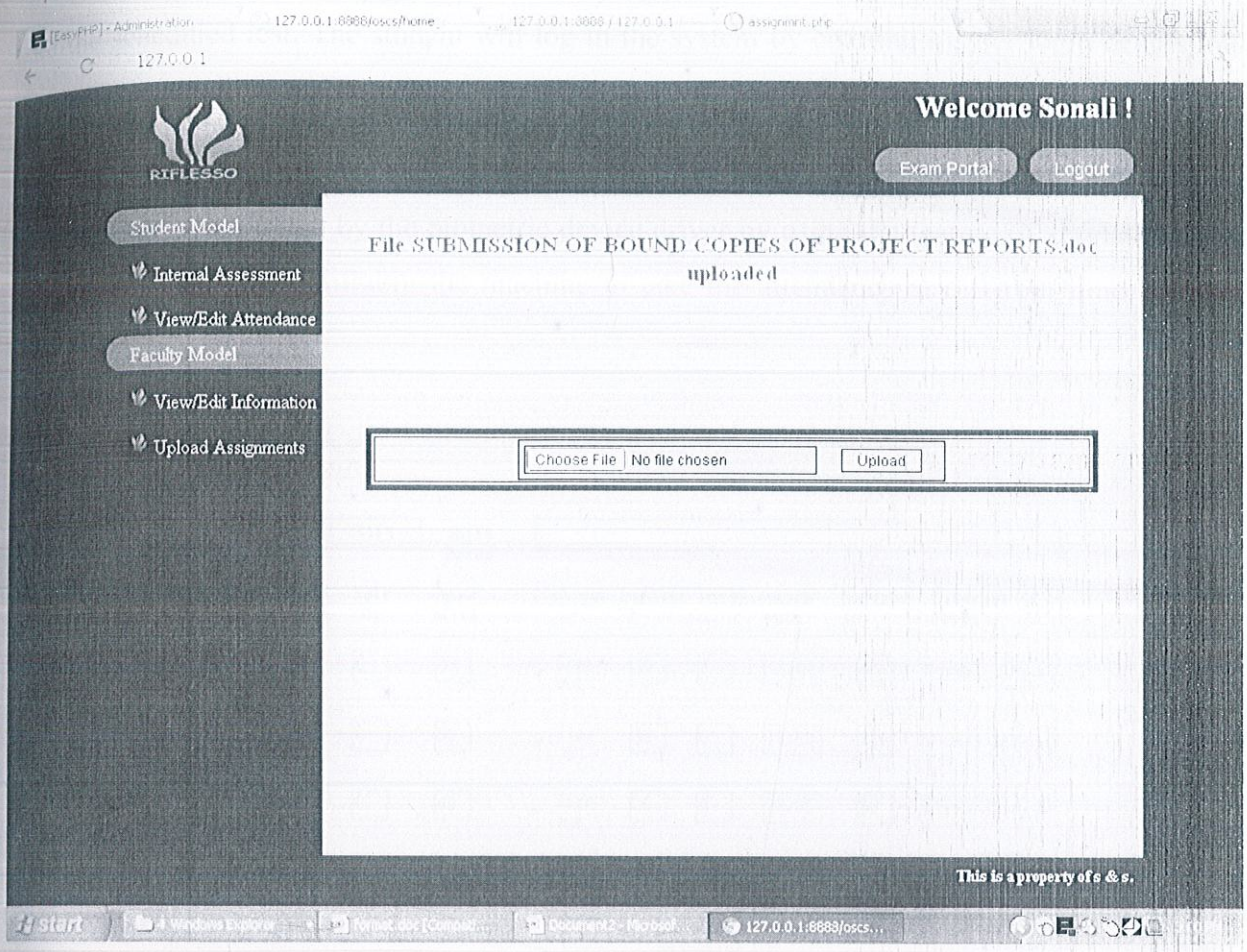


Fig. 5.12

5.3 Taking the Test

This module deals with the student logging in to the system and using the exam portal to take the scheduled test. The student will log in the system by entering his/her login details into the system and will proceed to the exam portal.

Before that the student has to register the thumbprint in order to register the attendance into the system. This is facilitated by the biometric device driver by using the feature of special shift. In this we can explicitly configure the machine to take the attendance at different time than the normal.

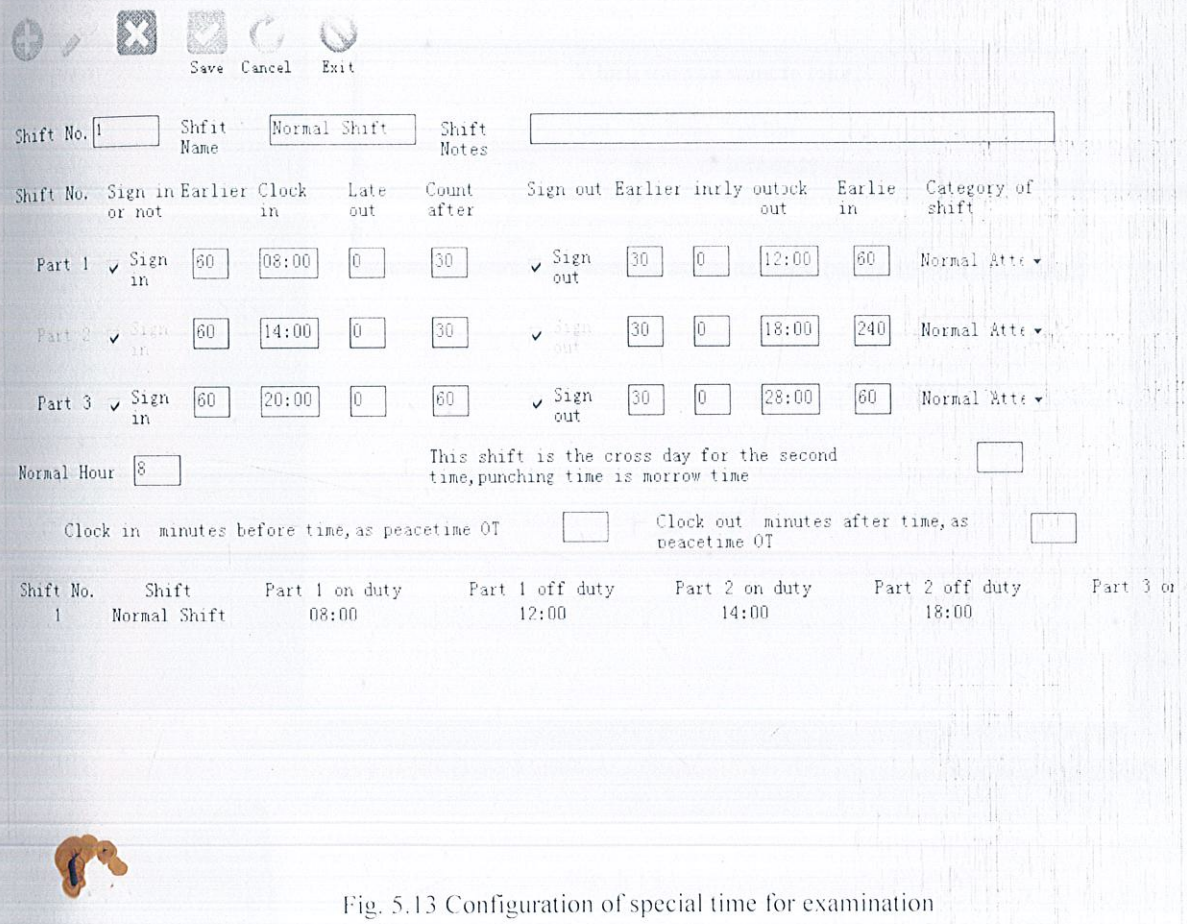


Fig. 5.13 Configuration of special time for examination

In this, we can change the time of attendance and make this shift as special shift. This shift will automatically revert back to its original value after this time is over.

If a student does not register his/her thumbprint then he/she will be marked absent in the examination and will not be able to get the marks registered even if he/she has given it.

This is how it authorizes the student to give the examination.

Also there are various techniques to ensure the integrity of the examination paper.

Now after logging into the system, the student will select the subject for which the examination has to be taken.

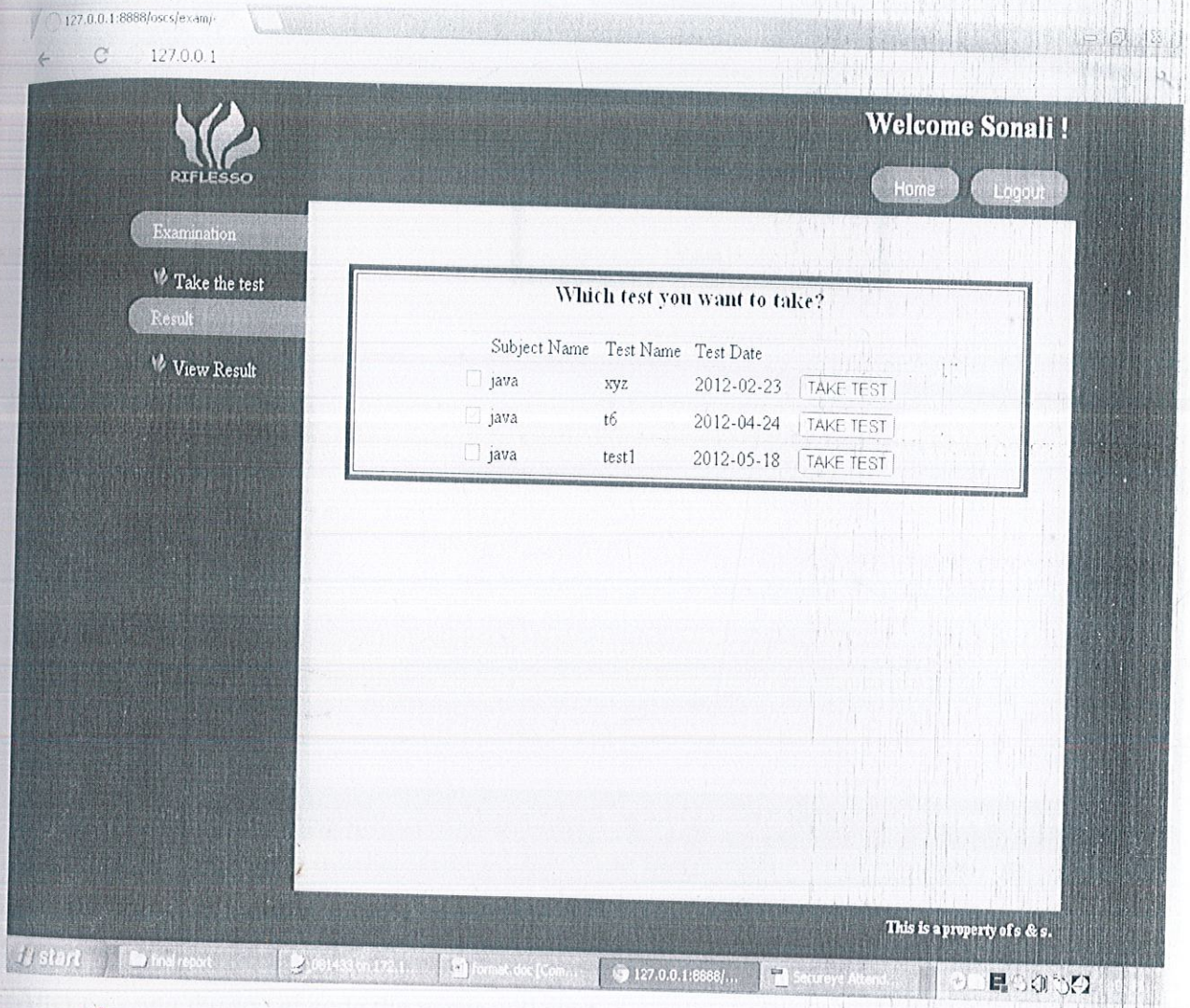


Fig. 5.14

After the selection of the test to be taken the portal will ask for a secret code to be entered. This code can will be told to the students to be entered at the time of examination by the invigilators or can be entered by the invigilator(s) as required.

The thought behind adding a secret code is to maintain the secrecy of the paper. No person can access the paper unless and until this secret code is entered. Also there are different codes for different examinations. Therefore no student can have access to the paper unless required to.

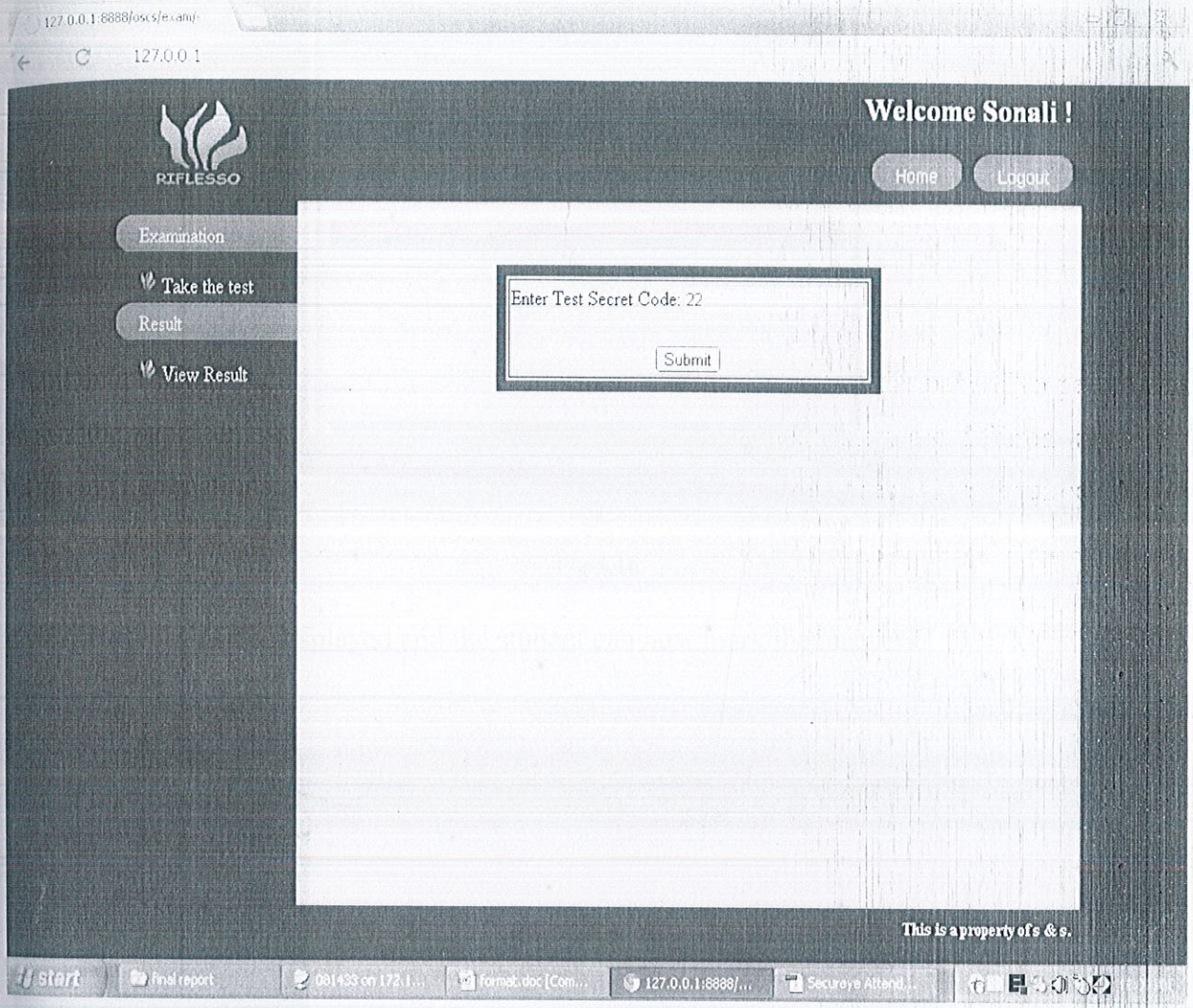


Fig. 5.15

After entering the secret code the paper will start.

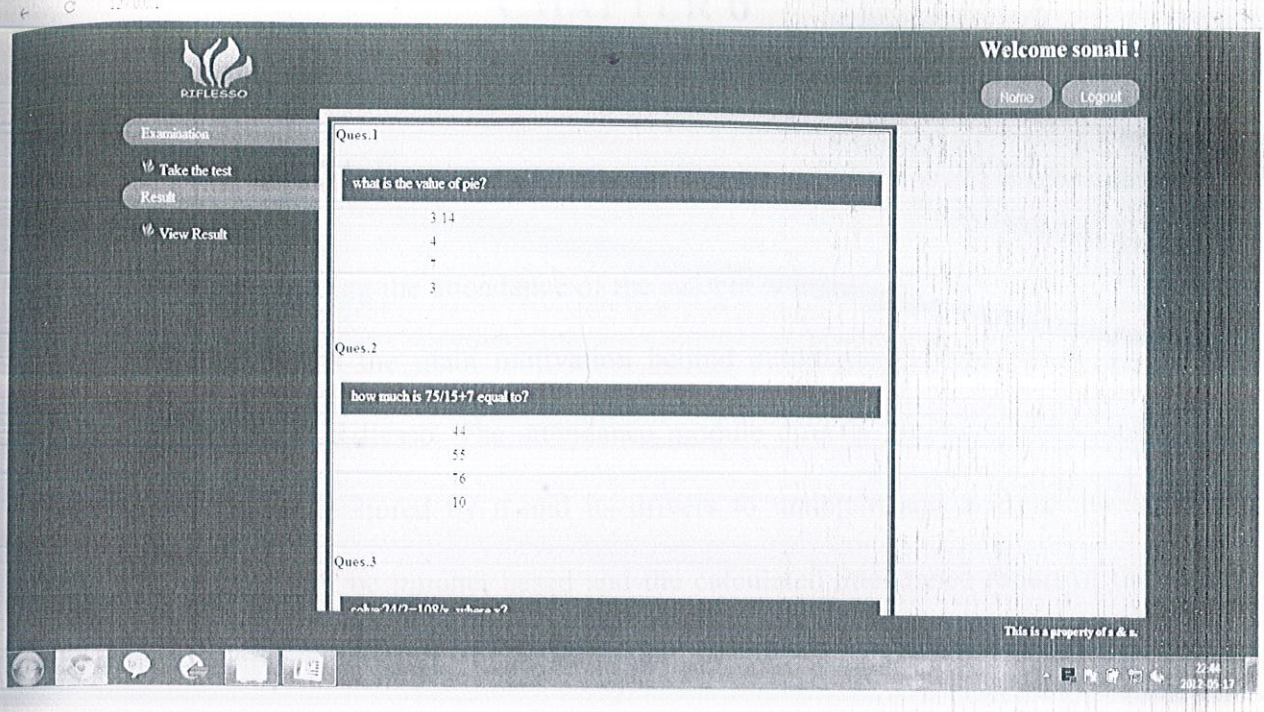


Fig 5.16

The question paper is displayed and the student can now mark the answers.

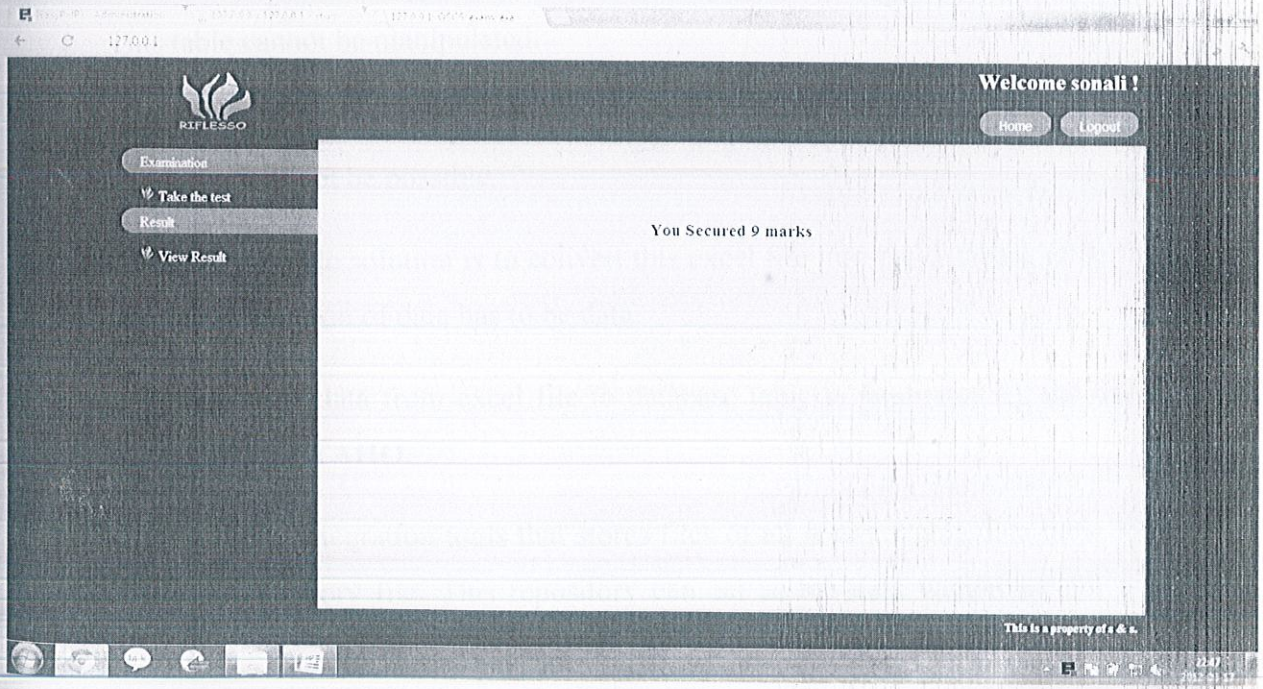


Fig 5.17

After submitting the paper the marks are displayed on the screen.

CHAPTER 6

6.1ATTENDANCE

This module is for recording the attendance of the student and maintains its record for the entire semester. This module is the main motivation behind automation of the whole system and undertaking this project Riflesso. The attendance module directly interacts with the biometric device and uses the data stored by it and its drivers to maintain and analyze its attendance records. The module will be intranet based and the calculated attendance report of the student can be made available on the web.

Issues encountered during the automation of the process.

- The data stored by the device is in the form of an excel table.
- This table cannot be manipulated.
- Excel file cannot be updated on regular basis and thus automation of the attendance recording will not be possible.
- The only possible solution is to convert this excel file into the database table format. For this transformation of data has to be data.

The transformation of data from excel file to database table is facilitated by the use of a data warehousing tool **PENTAHO.**

PENTAHO is a data integration tools that stores files of all recognizable data table formats and keeps it in one repository file. This repository can act as the data warehouse for a particular system.

This tool performs transformation operations from one file type to another in order to be used and manipulated by the languages.

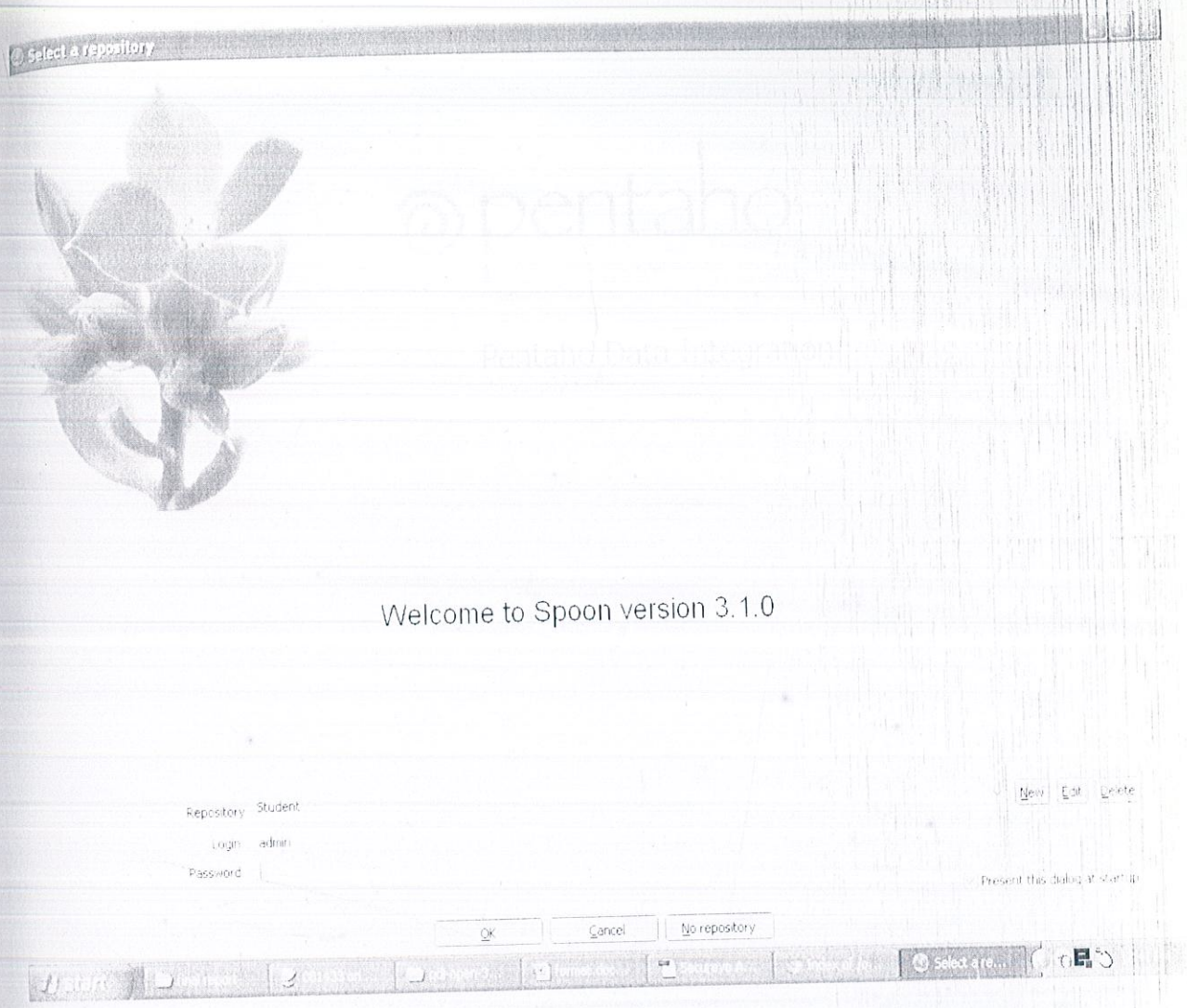


Fig. 6.1

Here we are using this tool to convert the data from the excel file into database file. This file then can be manipulated accordingly and used to generate the attendance reports of students.

- The figure on the next page shows the transformation process. The input file type as excel file is selected and is configured for taking only the relevant data.
- The output file as table output is selected and the connection to the database server is configured. The path to the table to be generated is specified and the connection is checked. Any errors in the SQL query to be executed are also checked.

Then a transformation is used which on successful execution will copy the data from excel file and convert it into the table.

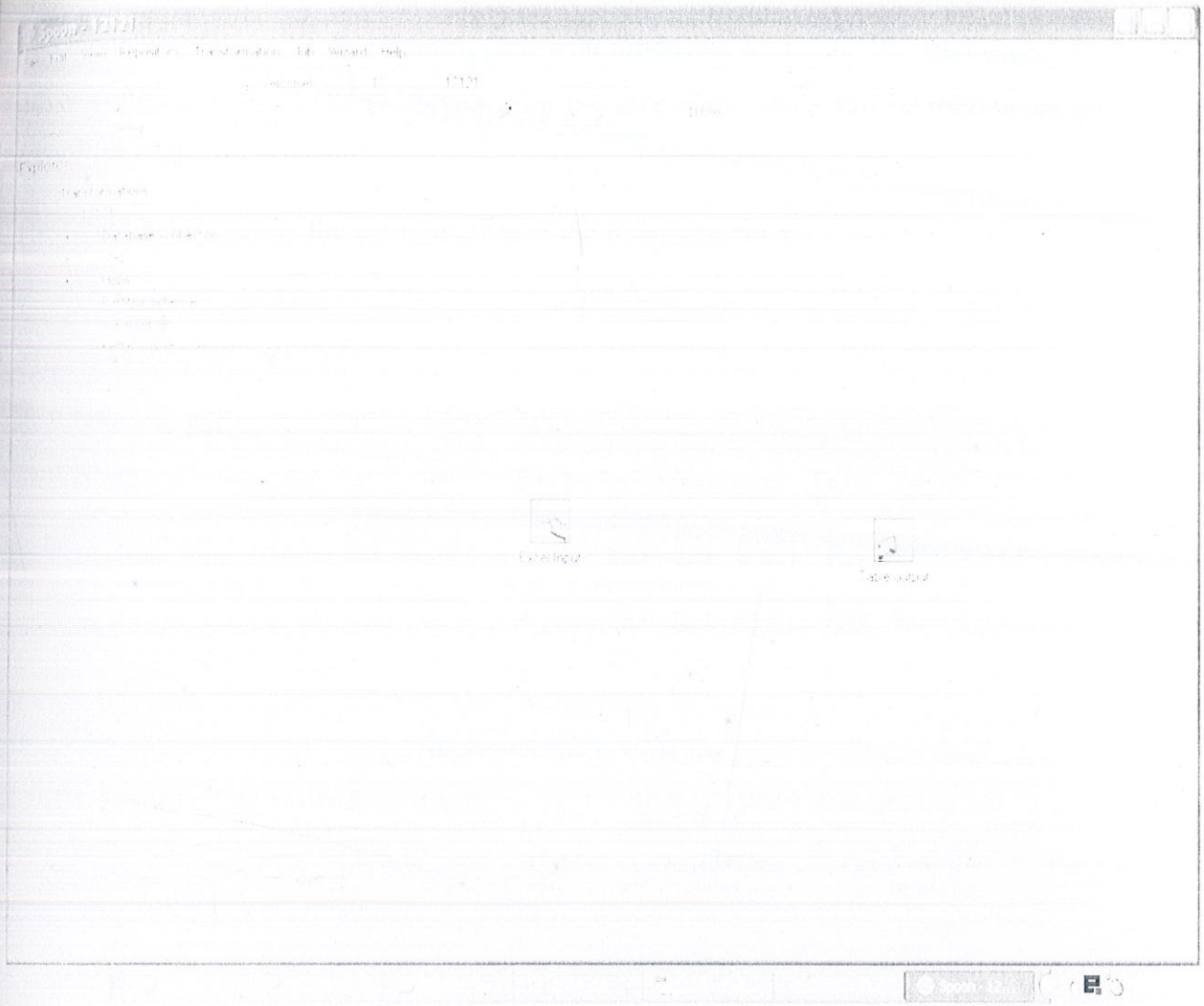


Fig. 6.2

The data in this table can now then be accessed by the system as required.

The issue regarding the automation of the data is also resolved as this transformation process can be made into an executable file and can be scheduled into the task scheduler similar to any anti-virus updating program.

At the scheduled time for transformation the process will execute itself and the transformation is carried out updating the database table with new data.

Configuration of Biometric Device

The biometric device has to be configured with time slots for taking the attendance. The device driver configures the biometric device with the time slots which can be used as an attendance record for a particular leeture.

The figure below shows the configuration of the biometric device.

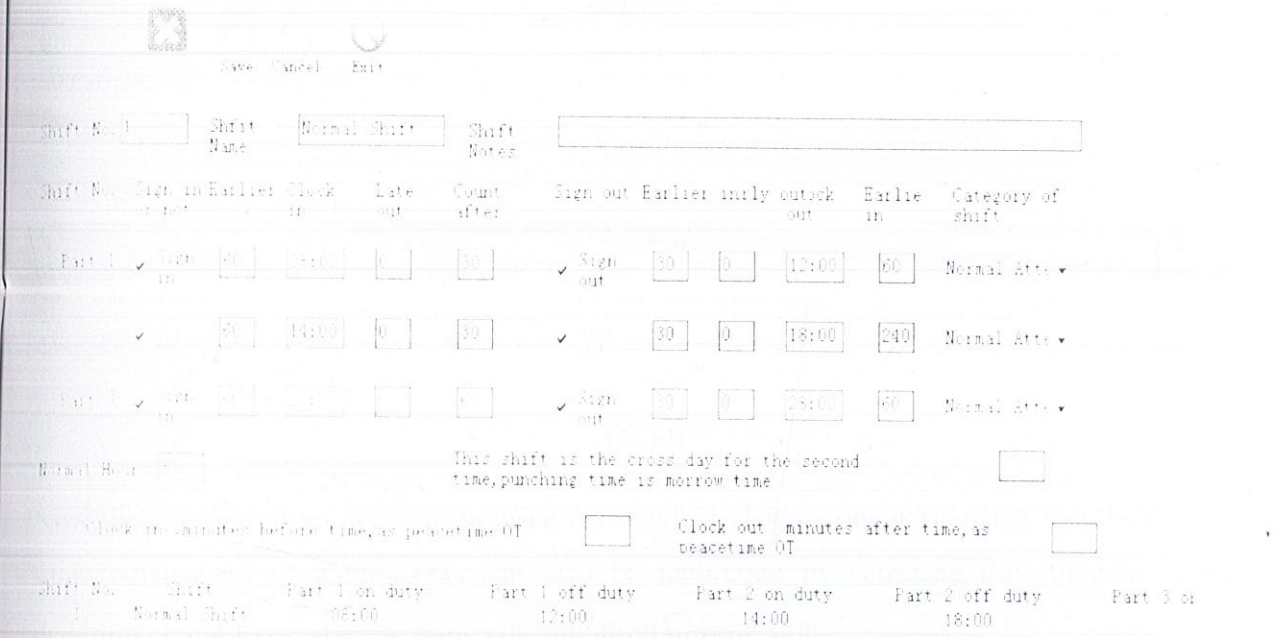


Fig. 6.3

The time for signing in to the device and the time to sign out is mentioned. A student has to register his her thumbprint at the time of signing in and also at the time of signing out. If a student does not perform any of the two things the attendance of the student for that particular class is not recorded and he she is marked absent.

Different time slots can be entered as there will be different classes.

In this way different time table formats can be created which can be used for different days of a week

Any configuration of time table can be repeated over the days in a week.

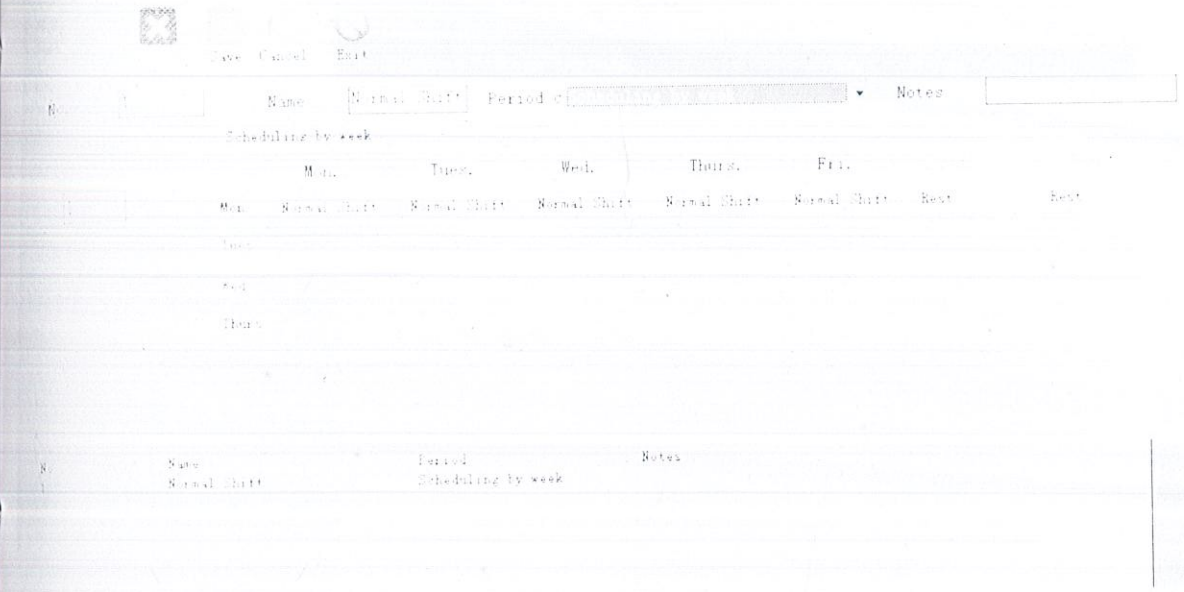


Fig. 6.4

Now this configuration needs to be used for many students, considering that they belong to the same department or batch. This can also be facilitated by selecting the students of whole department and assigning the same configuration of time slots.

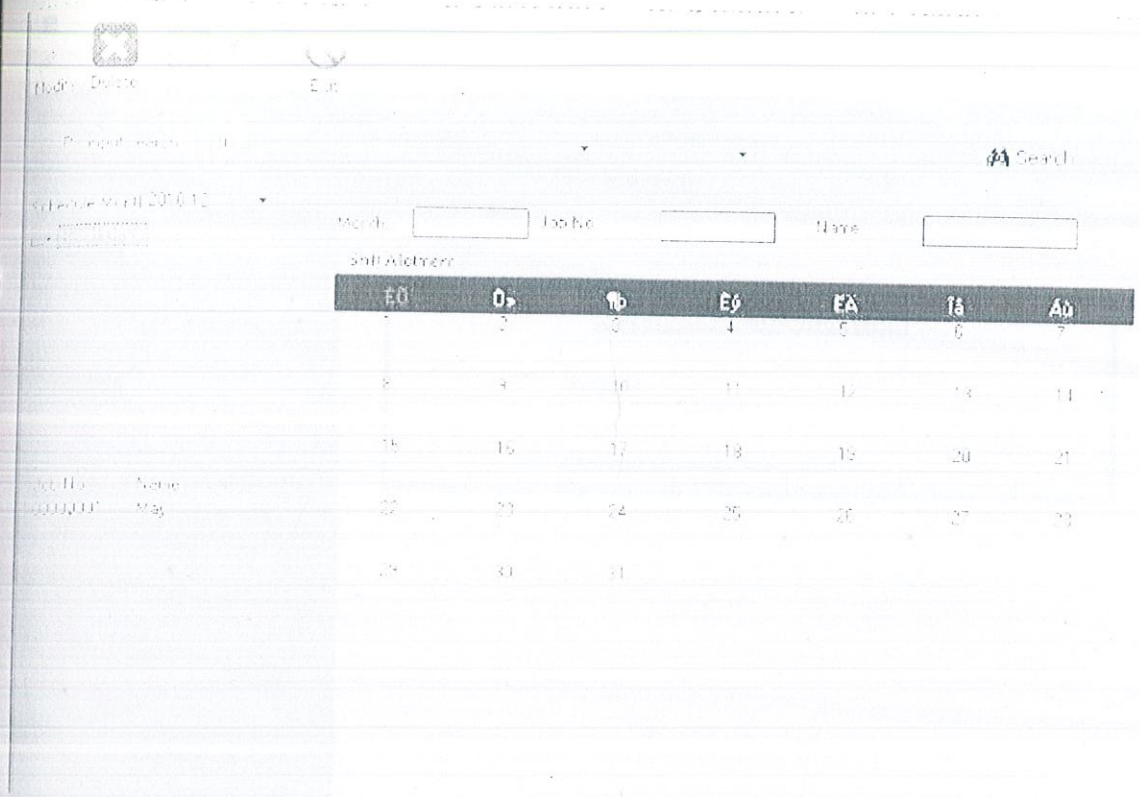


Fig. 6.5

Viewing of attendance

The attendance of a student can be viewed by him or her by logging into the system.

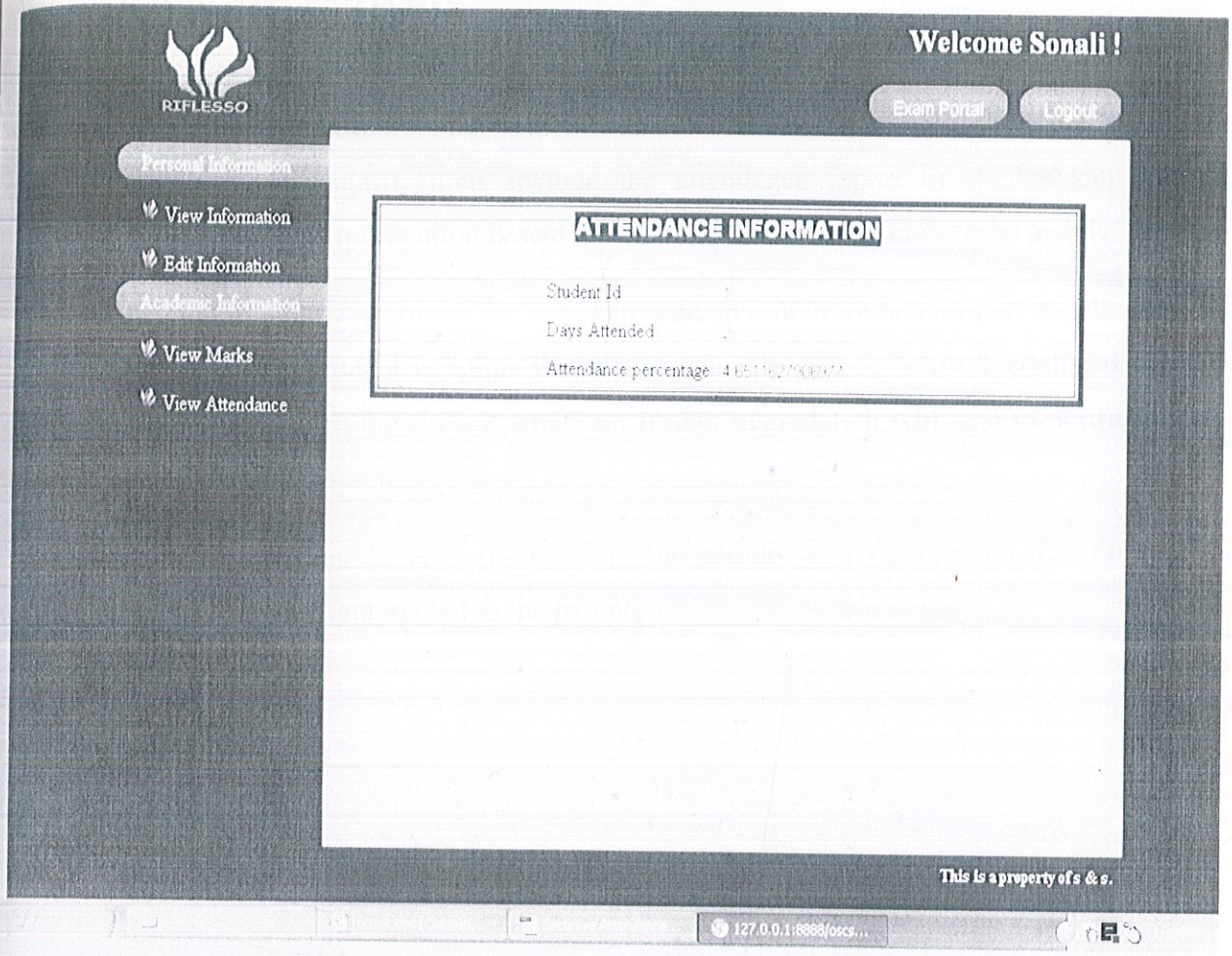

RIFLESSO
Welcome Sonali !
Exam Portal
Logout
Personal Information
View Information
Edit Information
Academic Information
View Marks
View Attendance
ATTENDANCE INFORMATION
Student Id
Days Attended
Attendance percentage
This is a property of s & s.

Fig. 6.6

6.2NOTIFICATIONS

The module deals with intimating the academic details of a student to the student, parents and the faculty members. The notifications include the attendance report of the student and the examination marks. This notification is sent through the e-mail to the concerned people.

The notification module will trigger the server to send an e-mail to the required people when the attendance report of a student is below the normal. The attendance below a certain percentage will send a notification to the student which on further degradation will lead to notifications to faculty then parents and so on.

The marks of the student are also intimated to the parents of a student and the result of the student is also compiled and mailed to the parents.

APPENDIX

// Registration:

```
<?php
$dbhost = 'localhost';
$dbuser = 'root';
$conn = mysql_connect($dbhost, $dbuser);
mysql_select_db('student');
$sql = "INSERT INTO student_info ( Student_Name ,Father_Name ,Mother_Name ,Dob
,address ,city ,state ,Phone_no ,Email ,pEmail ,Tenth ,Twelth,Password,Login_type,semester
)values(

'$_POST[a]',
'$_POST[b]',
'$_POST[c]',
'$_POST[d]',
'$_POST[e]',
'$_POST[f]',
'$_POST[g]',
'$_POST[h]',
'$_POST[i]',
'$_POST[j]','$_POST[k]',
'$_POST[l]','$_POST[m]','$_POST[student]','$_POST[sem]')";

$z = mysql_query($sql);
if(!$z){
echo mysql_error();
}

$m = mysql_query("select id from student_info WHERE phone_no = '$_POST[h]'");
if (mysql_num_rows($m) > 0)
{
while($row = mysql_fetch_row($m))
{
echo "Your id is :".$row[0];
}
}
mysql_close($conn);
?>
```

// Login:

```
<?php
session_start();
?>
<html>
<head>
<title>Untitled Page</title>
<?php
$dbhost = 'localhost';
$dbuser = 'root';
$conn = mysql_connect($dbhost, $dbuser);
mysql_select_db('student');
$x = $_POST['type'];
$a = "ADMIN";
$b = "FACULTY";
$c = "STUDENT";
if(strcmp($x,$a) == 0)
{
$flag = 3;}
if(strcmp($x,$b) == 0)
{
$flag = 2;}
if(strcmp($x,$c) == 0)
{
$flag = 1;}
if($flag == 1){
$login = mysql_query("SELECT * FROM Student_info WHERE (id = '" .
mysql_real_escape_string($_POST['id']) . "') and (password = '" .
mysql_real_escape_string($_POST['password']). "') and (Login_type = '" .
mysql_real_escape_string($_POST['type']). "')");
if (mysql_num_rows($login) == 1)
{
$_SESSION['id'] = $_POST['id'];
$_SESSION['type'] = $_POST['type'];
unset($_GLOBALS['message']);
header('Location: home_page.html');}
else
{
$_GLOBALS['message'] = "Check Your user name and Password.";
}
}
if($flag == 2){
$login = mysql_query("SELECT * FROM Faculty_info WHERE (facid = '" .
mysql_real_escape_string($_POST['id']) . "') and (fpassword = '" .
mysql_real_escape_string($_POST['password']). "')");
```

```
if(mysql_num_rows($login) == 1)
{
$_SESSION['id'] = $_POST['id'];
$_SESSION['type'] = $_POST['type'];
unset($_GLOBALS['message']);
header('Location: home_page.html');}
else
{
$_GLOBALS['message'] ="Check Your user name and Password.";
}
}
if($flag == 3){
$login = mysql_query("SELECT * FROM Login_info WHERE (id = '" .
mysql_real_escape_string($_POST['id']) . "') and (password = '" .
mysql_real_escape_string($_POST['password']). "')");
if(mysql_num_rows($login) == 1)
{
$_SESSION['id'] = $_POST['id'];
$_SESSION['type'] = $_POST['type'];
unset($_GLOBALS['message']);
header('Location: home_page.html');}
else
{
$_GLOBALS['message'] = "Check Your user name and Password.";
}
}
?>
```

// Subject insert:

```
<html><head><?php
$dbhost = 'localhost';
$dbuser = 'root';
$conn = mysql_connect($dbhost, $dbuser);
mysql_select_db('student');
$sql = "INSERT INTO Subject( subname ,subdesc) values(

'$_POST[a]',
'$_POST[b]')";
$z = mysql_query($sql);
if(! $z)
{
echo mysql_error();
}
```

```
else
?>
<head>
<body bakground "../img/bkg1.jpg">
<center>
<br><br>
<?php
$m= mysql_query("select subid from Subject WHERE subname='$_POST[a]'");
if(mysql_num_rows($m) > 0)
{
while($row = mysql_fetch_row($m))
{
echo"    SUBJECT ADDED SUCCESSFULLY!!!";}}
mysql_close($conn);
?>
</body></html>
```

// Test insert:

```
<html><head><?php
$dbhost = 'localhost';
$dbuser = 'root';
$conn = mysql_connect($dbhost, $dbuser);
mysql_select_db('student');
$sql = "INSERT INTO Test(subid ,testname ,testdesc ,totalquestions, duration,testdate ,test_hr,
testcode) values(

'$_POST[a]',
'$_POST[b]',
'$_POST[c]',
'$_POST[d]',
'$_POST[e]',
'$_POST[teston]',
'$_POST[hr]',
'$_POST[f]')";
$z= mysql_query($sql);
if(! $z)
{
echo mysql_error();
}

$m= mysql_query("select testid from Test WHERE testname='$_POST[b]'");
if(mysql_num_rows($m) > 0)
{
while($row = mysql_fetch_row($m))
{
```

```
echo "Test id is :".$row[0];
echo"      TEST ADDED SUCCESSFULLY!!!";
}
}
mysql_close($conn);

?>
</head>
<body bakground="bkg1.jpg">
<center>
<br><br>

</body></html>
```

Insert question:

```
<?php
session_start();
?>
<?php
$dbhost = 'localhost';
$dbuser = 'root';
$conn = mysql_connect($dbhost, $dbuser);
mysql_select_db('student');
$td = $_SESSION['testid'];
$sql = "INSERT INTO Question ( testid,question ,optiona,optionb,optionc,optiond,correctanswer
,marks)values(
'$td',
'$_POST[q]',
'$_POST[a]',
'$_POST[b]',
'$_POST[c]',
'$_POST[d]',
'$_POST[e]',
'$_POST[f]')";
$z= mysql_query($sql);
if(! $z){
echo mysql_error();
}


else{
echo "<br><br><br><center>";
echo" Press to<a href=\"make_ques.php\"><input type=\"button\" value=\"ADD NEXT
QUESTION\" /></a></center>";}
```

```
mysql_close($conn);

?>
<html><head></head><body background "bkg1.jpg"><center>

</body></html>
```

// Uploading Assignments:

```
<?php
if(isset($_POST['upload']))

{
   $fileName = $_FILES['userfile']['name'];
      $tmpName  = $_FILES['userfile']['tmp_name'];
      $fileSize = $_FILES['userfile']['size'];
      $fileType = $_FILES['userfile']['type'];
      $fp = fopen($tmpName, 'r');
      $content = fread($fp, $fileSize);
      $content = addslashes($content);
      fclose($fp);



      if(!get_magic_quotes_gpc())

      {          $fileName = addslashes($fileName);

      }
      $query = "INSERT INTO upload (subid, name, size, type, content ) ".

            "VALUES ('$_SESSION[subjid]', '$fileName', '$fileSize', '$fileType', '$content')";
      mysql_query($query) or die('Error, query failed');
      echo "<center><h3><br>File $fileName uploaded<br>";
}     mysql_close($conn);

?>
```

// Downloading Assignments:

```
<?php

if(isset($_GET['id']))

{
// if id is set then get the file with the id from database
$id    = $_GET['id'];

$query    "SELECT name, type, size, content " .

        "FROM upload WHERE id = '$id'";
$result = mysql_query($query) or die('Error, query failed');

list($name, $type, $size, $content) = mysql_fetch_array($result);
header("Content-length: $size");
header("Content-type: $type");
header("Content-Disposition: attachment; filename=$name");
echo $content;
exit;
}
 mysql_close($conn);

?>
```

// Taking the test:

```
<?php
session_start();
?>
<html>
<head>
<?php
$dbhost = 'localhost';
$dbuser = 'root';
$conn = mysql_connect($dbhost, $dbuser);
mysql_select_db('student');
$id = $_SESSION['testid'];?>
<style>
span.fieldset {background:#031B4B;
color:white;}
h4{color:#031B4B;}
</style>
</head>
<body background = "../img/bkg1.jpg" align = "left">
<form action = "check_ans.php" method = "post">
```

```
<table border="5" bordercolor="#031B4B"><tr><td>
<?php
$sql="SELECT * FROM Question WHERE testid='$td'";
$m=1;$i=31;
$names=array("nu","a","b","c","d","e","f","g","h","i","j","k","l","m","n","o","p","q","r","s","t","
u","v","w","x","y","z","aa","ba","ca","da");
$names1=array("nux","ax","bx","cx","dx","ex","fx","gx","hx","ix","jx","kx","lx","mx","nx","ox
","px","qx","rx","sx","tx","ux","vx","wx","xx","yx","zx","ay","by","cy","dy");
$result=mysql_query($sql);
if(!$sql)
{
echo"query failed .mysql_error()";
}
if(mysql_num_rows($result)>0)
{
        while($row=mysql_fetch_row($result))
{
echo"<table width=\"600\">";
echo"<h4>Ques.".$m."                                                                    </h4><tr><td
colspan=\"3\"><fieldset>".$row['2']."</fieldset></td></tr>";
echo"<input type=\"hidden\" name=\"".$names1[$m]."\" value=\"".$row['1']."\"></input>";

echo"<tr><td> </td><td></td><td><input      type=\"radio\"      name=\"".$names[$m]."\"
value=\"optiona\"></input>".$row['3']."</td></tr>";
echo"<tr><td> </td><td></td><td><input      type=\"radio\"      name=\"".$names[$m]."\"
value=\"optionb\"></input>".$row['4']."</td></tr>";
echo"<tr><td> </td><td></td><td><input      type=\"radio\"      name=\"".$names[$m]."\"
value=\"optione\"></input>".$row['5']."</td></tr>";
echo"<tr><td> </td><td></td><td><input      type=\"radio\"      name=\"".$names[$m]."\"
value=\"optiond\"></input>".$row['6']."</td></tr>";
echo "</table><br>";
$m=$m+1;}
$_SESSION['count']=$m;
$q=$m;
while(!($q==$i))
{
echo "<input type=\"hidden\" name=\"".$names1[$q]."\" value=\"0\"></input>";
echo "<input type=\"hidden\" name=\"".$names[$q]."\" value=\"0\"></input>";
$q=$q+1;
}
}
mysql_close($conn);
?>
<br><br><br><span>
<input type="submit" value="SUBMIT TEST"></input></span></td></tr></table>
</form>
</body>
</html>
```

//Check answers:

```
<?php
session_start();
?>
<html><head></head><body background="../img/bkg1.jpg">

<br><br><center><h3>

<?php
$dbhost = 'localhost';
$dbuser = 'root';
$conn = mysql_connect($dbhost, $dbuser);
mysql_select_db('student');

$td = $_SESSION['testid'];
$et = $_SESSION['count'];
$sd = $_SESSION['id'];
$marks=0;$i=0;
$a = $_POST['a'];
$b = $_POST['b'];
$c = $_POST['c'];
$d = $_POST['d'];
$e = $_POST['e'];
$f = $_POST['f'];
$g = $_POST['g'];
$h = $_POST['h'];
$i = $_POST['i'];
$j = $_POST['j'];
$k = $_POST['k'];
$l = $_POST['l'];
$m = $_POST['m'];
$n = $_POST['n'];
$o = $_POST['o'];
$p = $_POST['p'];
$q = $_POST['q'];
$r = $_POST['r'];
$s = $_POST['s'];
$t = $_POST['t'];
$u = $_POST['u'];
$v = $_POST['v'];
$w = $_POST['w'];
$x = $_POST['x'];
$y = $_POST['y'];
$z = $_POST['z'];
$aa = $_POST['aa'];
$ba = $_POST['ba'];
$ca = $_POST['ca'];
$da = $_POST['da'];

$ax = $_POST['ax'];
```

```
$bx = $_POST['bx'];
$cx = $_POST['cx'];
$dx = $_POST['dx'];
$ex = $_POST['ex'];
$fx = $_POST['fx'];
$gx = $_POST['gx'];
$hx = $_POST['hx'];
$ix = $_POST['ix'];
$jx = $_POST['jx'];
$kx = $_POST['kx'];
$lx = $_POST['lx'];
$mx = $_POST['mx'];
$nx = $_POST['nx'];
$ox = $_POST['ox'];
$px = $_POST['px'];
$qx = $_POST['qx'];
$rx = $_POST['rx'];
$sx = $_POST['sx'];
$tx = $_POST['tx'];
$ux = $_POST['ux'];
$vx = $_POST['vx'];
$wx = $_POST['wx'];
$xx = $_POST['xx'];
$yx = $_POST['yx'];
$zx = $_POST['zx'];
$ay = $_POST['ay'];
$by = $_POST['by'];
$cy = $_POST['cy'];
$dy = $_POST['dy'];
$nm = array($a,$b,$c,$d,$e,$f,$g,$h,$i,$j,$k,$l,$m,$n,$o,$p,$q,$r,$s,$t,$u,$v,$w,$x,$y,$z,$aa,$ba,$ca,$d
a);
$names1 = array($ax,$bx,$cx,$dx,$ex,$fx,$gx,$hx,$ix,$jx,$kx,$lx,$mx,$nx,$ox,$px,$qx,$rx,$sx,$tx,$ux,$
vx,$wx,$xx,$yx,$zx,$ay,$by,$cy,$dy);
}
while(!($i = $_SESSION['count']))
{

$check1 = $nm[$i];
$check2 = $names1[$i];
$i = $i + 1;
$sql="SELECT * FROM Question WHERE qnid = '$check2' and correctanswer='$check1'";
$result=mysql_query($sql);
if(mysql_num_rows($result) == 1)
{
$row = mysql_fetch_array($result);
$marks = $marks + $row['8'];
}

}
echo"<br>You Secured ".$marks." marks";
$sql1 = "INSERT INTO Student_Test( id,testid,finalmarks)values(
'$sd',
```

```
'$td',
'$marks')";
$za=mysql_query($sql1);
if(!$za)
{
echo mysql_error();
}

{mysql_close($conn);
?>
</body></html>
```

// Attendance:

```
<?php
session_start();
?>
<html><head></head>
<body background="img/bkg1.jpg">
<br><br>
<form><center><table border="5" bordercolor="#031B4B" width="600"><tr><td><center>
<table>
<tr><td colspan="5" align="left" style="background-color: #031B4B"><font color="white"
size="2" face="Arial, Helvetica, sans-serif"><strong>
            <font color="white" size="4">ATTENDANCE
INFORMATION</font></strong></font></td>
        </tr></table>
<?php
$dbhost='localhost';
$dbuser='root';
$conn=mysql_connect($dbhost, $dbuser);
mysql_select_db('attendance');
$z=$_SESSION['id'];$count=0;$count_days=0;
$zz=array(0,0,0,0,0,0,0,0);
$zz[7]=$z;$zz1=implode($zz);
$sql="SELECT * FROM attendance WHERE id='$zz1'";
$result=mysql_query($sql);
if(!$sql)
{
echo"query failed.mysql_error()";
}
if(mysql_num_rows($result)>0)
{
while($row=mysql_fetch_row($result))
{
if($row[0]!=NULL)
{$count=$count+1;}
```

```
if($row[3]!=NULL)
{$count_days=$count_days+1;}
$pert=$count/$count_days;
$pert1=$pert*100;
}}
echo"<br>";
echo"<table style=width: 400px; height: 122px rules=\"rows\">";
echo "<tr><td > Student Id:</td><td><input type=text name=z value=".$z;
echo " readonly></input></td></tr>";
echo " <tr><td>Days Attended:</td><td><input type=text name=b value=".$count;
echo " readonly></input></td></tr>";
echo " <tr><td>Attendance percentage:</td><td><input type=text name=c value=".$pert1;
echo " readonly></input></td></tr></table>";
mysql_close($conn);
?></td></tr></table></form>
</body>
</html>
```

REFERENCES

- Learning PHP, MySQL, and JavaScript by Robin Nixon - O'Reilly (2009)
- PHP And Mysql Web Development By Luke Welling, Laura Thomson - Addison-Wesley (2008)
- www.phpreferencebook.com
- www.w3schools.com
- www.php.net
- www.tizag.com
- www.stackoverflow.com
- www.freewebmasterhelp.com
- www.sourceforge.net