

A Project/Dissertation ETE Report

On

Web-Based Application for Automatic Timetable Generation

*Submitted in partial fulfillment of
the requirement for the award of
the*

degree of

B.Tech in Computer Science



(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

**Under The Supervision of Name of
Supervisor : Mr.S.Janarthanan**

Submitted By

Name of Student/s :Ankita Singh

**Enrollment/Admission No. 19SCSE1010411/19021011591
SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA
INDIA**

ACKNOWLEDGEMENT

We would like to express our special thanks of gratitude to our guide (Mr.S.Janarthanan) who gave us a golden opportunity to do this project and has also provided his full support to complete this project.

We would also like to extend our gratitude to our Department School of computing science and engineering for providing us with all the facilities that are required.

Sincere Regards,

From the Team

TABLE OF CONTENTS

S.No
1 .ABSTRACT
2. INTRODUCTION
3. PROBLEM STATEMENT
4. OBJECTIVE
5. CONCLUSION
6. FUTURE WOKS
7.METHODOLGY AND DISCUSSION
8.PROPOSED SYSTEM AND ARCHITECTURE OF PROPOSED SYSTEM

Project Title: Web Based Application For Automatic Time Table Generation

ABSTRACT:

Timetabling Problem is NP-Hard trouble which is very hard to clear up through the usage of conventional strategies. A lot of complicated constraints need to be addressed for improvement of an efficient set of rules to resolve this timetabling trouble. Therefore, there is a tremendous requirement for an software dispensing the path lightly and without collisions. There are numerous equipment to be had for producing timetable. This device can lessen our guide work of producing timetable but barriers of this tool is that it requires greater time, gives less accuracy and additionally blunders price is excessive. So, our purpose here is to expand a simple, without problems understandable, efficient and transportable software, which may want to mechanically generate true excellent timetable inside seconds. In the prevailing scenario, all of the university related work together with making the defaulter list of students and measuring the performance of a instructor consistent with comments given by way of students is finished manually. All those responsibilities are time eating and additionally require a variety of efforts and sources. To solve those issues our gadget uses Sentiment evaluation API for producing remarks of instructor. System generates teacher's overall performance graph consistent with remarks given by way of scholar. The gadget may even send alert message to students if their attendance is less than seventy five% .The major reason of our gadget is to reduce the work load of instructors and additionally be a value effective and a short respondent gadget. KEYWORDS: Hard constraints, tender constraints, Sentiment analysis, Mashape API , Workload, Attendance sheet.

1. INTRODUCTION

Overview

Every educational institution has a timetable of its personal to be observed by the students who take a look at there. Preparing a timetable might be smooth only when there are much less wide variety of college students in that unique organization i.E., (getting ready it manually) . Here comes the usage of Timetable generator i.E., the use of a software program to manipulate timetable. Automatic timetable generation is a java based software that makes it convenient for the designer of timetable to prepare the timetable. The system aids solving the timetabling problem even as giving significance to instructor availability. This hassle makes use of a heuristic technique to present a general solution to timetabling problem. During computerized timetable technology, there are one of a kind requirements that want to be happy and they may clutter the consumer interface. It takes the user enter of a number of subjects, range of teachers, subjects each teacher takes, range of days in a week for which the timetable wishes to be set, wide variety of time slots in a day and the maximum lectures a trainer can behavior in a week.

The difficulty of making timetables for classrooms is a scheduling algorithm with tremendous curiosity and association in the fields of artificial intelligence and operational research. This problem is being dealt in many organizations manually, i.e. timetables

are set using a trial and error procedure. The process of preparing a timetable involves beneficial employment of resources which needs to be confronted each year by every educational institute.

LITERATURE SURVEY/REVIEW OF LITERATURE

Trying to develop a software which helps to generate Timetable for an Institution automatically. By looking at the existing system we can understand that timetable generation is done manually. Manually adjust the timetable when any of the faculty is absent, and this is the big challenge for the faculty is absent. As we know all institutions/organizations have its own timetable ,managing and maintaining these will not be difficult. Considering workload with this scheduling will make it more complex. As mentioned , when Timetable generation is being done, it should consider the maximum and minimum workload that is in a college. In those cases timetable generation will become more complex. Also , it is a time consuming process.

SOFTWARE REQUIREMENT ANALYSIS

The current problem is setting up of Timetable manually. Any absent is their , then assigning that period to another Faculty is a not an easy job because at that time we should consider the Maximum workload for a Faculty (assigning extra period should not exceed the Maximum workload).

3.1 Problem Definition

Avoid the complexity of setting and managing Timetable manually. It will help you to manage all the periods automatically. Faculty can receive their periods information in their phone . Initially we will be setting the maximum workloads for a Faculty in a day, week and month. Main challenge is to manage timetable when any Faculty is absent. By using this software it will be very easy to allocate subject for different faculty.

3.2 Fesibility Analysis

A feasibility study was an evaluation of a proposal designed to determine the difficulty in carrying out a designated task. Generally, a feasibility study precedes technical development and project implementation.

3.2.1 Economic Feasibility

To develop the proposed system, it needs no extra facilities and devices. All dependencies are satisfied from the open source projects. All tools used are free, open source and the programming language is JSP and hence its development is economically .

3.2.2 Technical Feasibility

Proposed system is technically feasible because the proposed system

requires only those H/W and S/W tools that are available in the system. It requires the installation of JSP and MYSQL which can be done for free. More over expandability will be

maintained in the new system. New modules can be add later on the application, if required in

the future. Additionally the application will have Userfriendly Forms and Screens.

3.2.3 Behavioral Feasibility

Behavioral feasibility determines how much effort will go in the proposed information system, and in educating and training the users on the new system.

Since the user interface is very simple and easily understandable , no training is required for using this software.

NETBEANS IDE FOR JAVA

NetBeans IDE provides first-class comprehensive support for the newest Java technologies and latest Java enhancements before other IDEs. With its constantly improving Java Editor, many rich features and an extensive range of tools, templates and samples, NetBeans IDE sets the standard for developing with cutting edge technologies out of the box.

THE JAVA PROGRAMMING LANGUAGE

Java is a programming language originally developed by Sun Microsystems and released in 1995 as a component of Sun Microsystems platform. The language derives much of its syntax from C and C++, but has a simpler object model and fewer low level facilities. Java applications are typically compiled to byte code that can run on any Java Virtual Machine(JVM) regardless of computer architecture.

The Java programming language is a high level languages that can be characterized by all the following:

- Simple
- Object-Oriented
- Distributed
- Multi-threaded
- Dynamic
- Portable

➤ Architecture Neutral

In Java programming language all source code is written in plain text

files ending with Java extension. Those source files then compiled into class files by the Javac compiler. A class file does not contain code that is native to your processor: it instead contain byte codes- the machine language of Java

Virtual

My SQL

My SQL is the world's most used open source Relational Database Management system(RDBMS) as of 2008 that runs as a server providing multi-User access to a number of databases. It is named after co-founder Michael Widenius' daughter, My. The SQL phrase stands for Structured Query Language. The My SQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. My SQL was owned and sponsored by a single for-profit firm, the Swedish company My SQL Lab, now owned by Oracle Corporation. MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack—LAMP is an acronym for "Linux, Apache, My SQL, Perl /PHP/Python." Free-software -open source projects that require a full-featured database management system often use My SQL

Interfaces

My SQL is primarily an RDBMS and ships with no GUI tools to

Administer My SQL databases or manage data contained within the databases.

Users may use the included command lines tools, or use My SQL "front-ends",

Desktop software and web applications that create and manage My SQL databases,

build database structures, back up data, inspect status, and work with data

records .. The official set of My SQL front-end tools, My SQL workbench is

actively developed by Oracle, and is freely available for use.

Software and Hardware requirements

Platform forms the foundation on which the architecture, design, and implementation of a product is built. System specification defines the full functionality of the system. In many systems we work on, some functionality performed in hardware and some in software. System specification documents can thus be defined as the requirements documentation that formally specifies the system level requirements of an application. This application developed in Windows platform.

Software Specification > Operating System: Windows/Linux > Technology: JSP/SERVLET > Web Technologies: X Html, JavaScript, CSS > Web Server: Tomcat/Glassfish > Design Tools :Net Beans IDE > Compiler : JDK > Database : My SQL > Web Browser :Mozilla Firefox/Google Chrome Hardware Specification >Hardware : Pentium >Speed : 2 Ghz >RAM : 2GB >Hard Disk : 80 GB

3.5 Define the modules and their functionalities The proposed system consists of the following modules: 3.5.1 Subject management Module Subject management module in this project is meant for the users. In this module, subjects are allocated for users. It does the following tasks:

- ✓ Give subject , faculty and Semester
- ✓ View timetable
- ✓ View leave request

2. PROBLEM STATEMENT

During analysis of this problem, we can remark that an automated introduction of timetables has elements. First problem is in complexity of the answer and searching set of rules for the solution. This is a combinatorial problem with a massive wide variety of variables. Only small percent of them are possible timetables, and some of them can be considered as appropriate ones. The hassle is partially solvable the usage of a selection of heuristic and optimization techniques, integer linear programming, taboo search, genetic algorithms and so on. Much less explored hassle is in defining necessities of the timetable. This query is associated no longer best to computerized time desk solving, but also for usage in software program which most effective take a look at constraints , or for documentation of guide timetabling. Sometimes it is sufficient to go into a matrix with the names of instructors and instructions in the matrix header, filled with weekly number of hours that the teacher teach in the particular class.

Objective :

Normally timetable era done manually. As we understand all establishments have their own timetable, coping with and retaining these will no longer be hard. Considering workload with tis scheduling will make it more complex. As noted, whilst Timetable generation is being performed, it must keep in mind the maximum and minimal workload this is in a university. In those cases timetable generation turns into greater complicated. Also, making ready timetable manually consumes a lot of time so, the primary goal is to design an set of rules which could remedy this hassle efficiently and successfully and put into effect that set of rules through a targeted programming language like java.

Organisation of document

- The idea of timetabling has turn out to be a very complex trouble now a days for any business enterprise. So, this trouble calls for a completely green approach to make the designing of timetable an clean challenge.
- The mission as a whole calls for to shop facts and content material of working body ofworkers in that group.

So, a database is required to hold the information of these information.

- The trouble of timetabling wishes to fulfill the necessities of constraints that rely in that organization and the restrictions might not be equal as special institutes comply with exclusive timetable and rules.
- These all information maintained in database are to be blanketed to be able to misuse those info of body of workers.
- The generated timetable by way of thinking about all the above elements have to be efficient with time and made to look quite easy.
- The person interface layout ought to be easy as even a normal person need to apprehend

EXISTING SYSTEM:

Generally timetable generation is accomplished manually. Present day preparator of timetable use MS EXCEL and MSWORD . This way of preparing timetable doesn't make the machine efficient and makes it look extra complex it is also a time eating procedure.

ISSUES IN EXISTING SYSTEM:

As we know all institutions/corporation have its very own timetable dealing with and preserving those may be tough . Considering the workload of workforce will make the scheduling part of timetable extra complicated . Because of those constraints managing personnel with recognize to their workloads may be difficult.

PROJECT SCOPE:

Timetable Generation System generates timetable for every class and trainer, in line with the availability calendar of instructors, availability and potential of bodily resources consisting of lecture rooms and rules applicable at special training, semesters, instructors and subjects degree. Best of all, this Timetable Generation System quite improves resource usage and optimization.

Salient features of the system

- Automatic Timetable manager is a Java based software used to generate timetable automatically.
- It will also manage timetable when any teacher is absent late coming or early going.
- Proposed system will help to generate it automatically also helps save the time .
- There is no need for Faculty to worry about their timetable
- It is a comprehensive timetable management solutions for Colleges which help to over the challenges in current system.

CONCLUSION

The aim of the assignment is to generate a time-desk which could schedule automatically. Timetabling trouble being the difficult combinatorial problem this is might take extra than just the software of handiest one principle. The timetable trouble may additionally handiest be solved while the constraints and allocations are definitely defined and simplified thoroughly and more than one principle is applied to it (i.e. A hybrid solution – a mixture of different answer techniques). This contains a number of strategies, aimed to enhance the efficiency of scheduling. It also, addresses the vital difficult constraint of clashes between the availability of teachers. The non-rigid soft constraints i.e. Optimization objectives also are efficiently treated. Thus, thru the technique of automation of the time-table trouble, many an-hours of making an effective timetable were decreased eventually. By considering a majority of these constraints successfully and correctly a timetable may be generated which sooner or later may be used for

a selected elegance of an institution and might additionally be stored for future. This whole process of storing data of team of workers and topics and retrieving them by way of timetable technology element makes use of database and as a result generates a properly scheduled timetable by thinking about the above constraints.

SOFTWARE AND HARDWARE REQUIREMENTS

Software Specification

- Operating System : Windows/Linux
- Technology : JSP/SERVLET
- Web Technologies : XHTML, JavaScript, CSS
- Web Server : Apache Tomcat/Glassfish
- Design Tools : NetBeans IDE
- Compiler : JDK
- Database : MySQL
- Web Browser : Mozilla Firefox/Google Chrome

Hardware Specification

- Hardware : Pentium
- Speed : 2 Ghz
- RAM : 2GB
- Hard Disk : 80 GB

Define the modules and their functionalities

The proposed system consists of the following modules:

Timetable Generation Module

It is the important part of project which generate Timetable automatically. IN this module it develop module. Generation done by considering maximum and minimum workload for a Faculty (without less and without exceeding). This will be generated by admin and viewed by Principal and Faculty who's are the users of the system.

User management Module

Staff module can classified into two:

- Faculty
- Principal

Faculty does

- Register with all information
- View Timetable
- Apply for leave with specifying Substitute faculty
- View Subtitute response
- Give subject name, semester and faculty
- View timetable
- Request for leave

Principal does

- View Timetable
- View request for leave
- View substitut's response
- Approve or Reject request for leave

SOFTWARE DESIGN

The most creative and challenging phase of the life cycle is system design. The term design describes a final system and the process by which it is developed. It refers to the technical specifications that will be applied in implementations of the system. The importance of software design can be stated in a single word "Quality". Design provides us with representations of software that can be assessed for quality. Design is the only way where we can accurately translate user requirements into a complete software product or system. Without design we risk building an unstable system that might fail if small changes are made. It may as well be difficult to test, or could be one who's quality can't be tested. So it is an essential phase in the development of a software product.

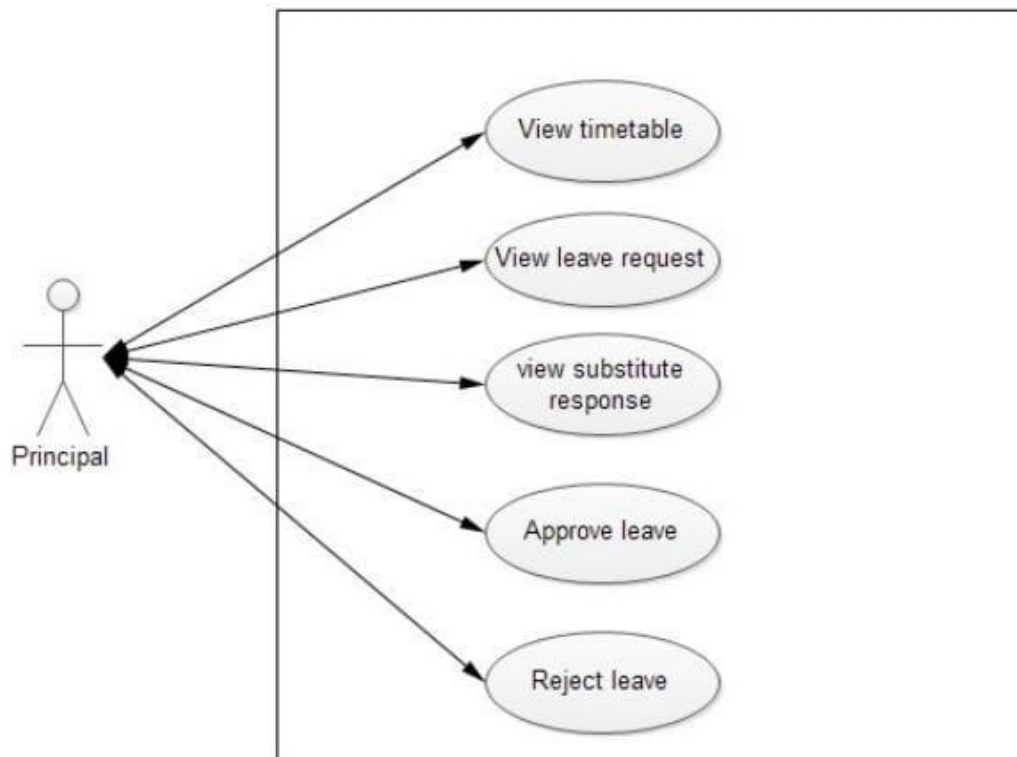
4.1 UML Diagrams

4.1.1 Use-Case diagram

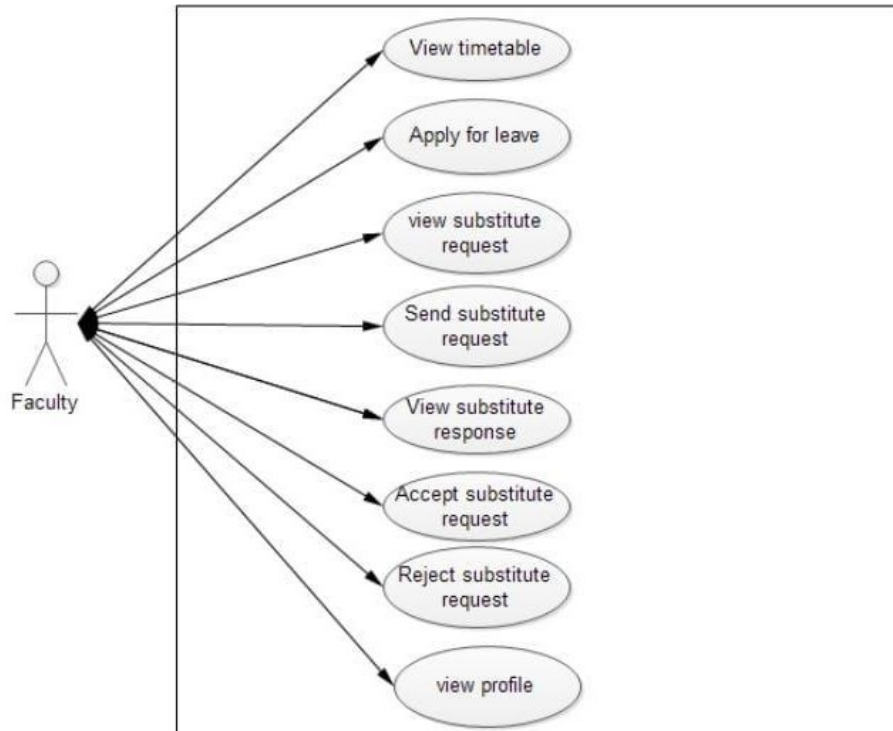
This system is used by 3 types of users. They are

- Admin , Principal ,Users(faculty)

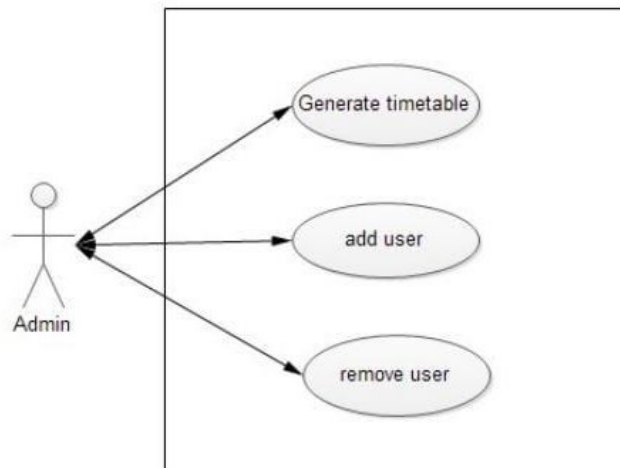
Principal



Faculty



Admin



Data Flow Diagrams

The DFD is also known as the bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of the input data to the system, various processing carried out in these data and the output data generated by the system. Starting with a set of high-level functions that a system performs, a DFD model in a hierarchical manner represents various sub-functions.

In a normal convention, a logical DFD can be completed using only four notations:

- Represents source or destination of data
- Represents Data Flow
- Represents a process that transforms incoming data into outgoing data
- Represents Data Source

Function Symbol

A function is represented using a circle. This symbol is called a process or a bubble. Bubbles are annotated with the names of corresponding functions.

External Entity Symbol

An external entity such as a librarian, library member etc is represented by a rectangle. The external entities are essentially those physical entities external to the software system, which interact with the system by inputting data to the system or by consuming the data produced by the system.

Data Flow Symbol

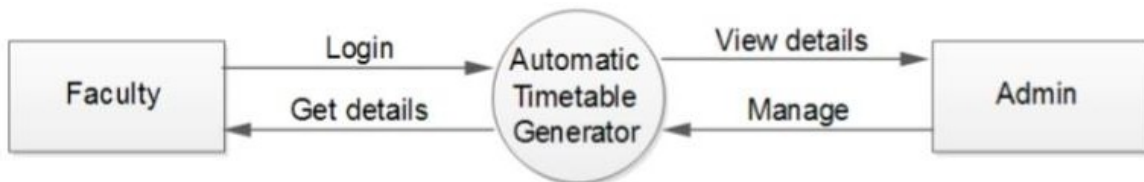
A directed arc or an arrow is used as a Data Flow Symbol. This represents the data flow occurring between two processes or between an external entity and a process in direction of the Data Flow Arrow. Data Flow symbols are annotated with corresponding data names.

Data Store Symbol

A Data Store represents a logical file; it is represented using two parallel lines. A logical file can represent either Data Store Symbol, which can represent either data structure or a physical file on disk.

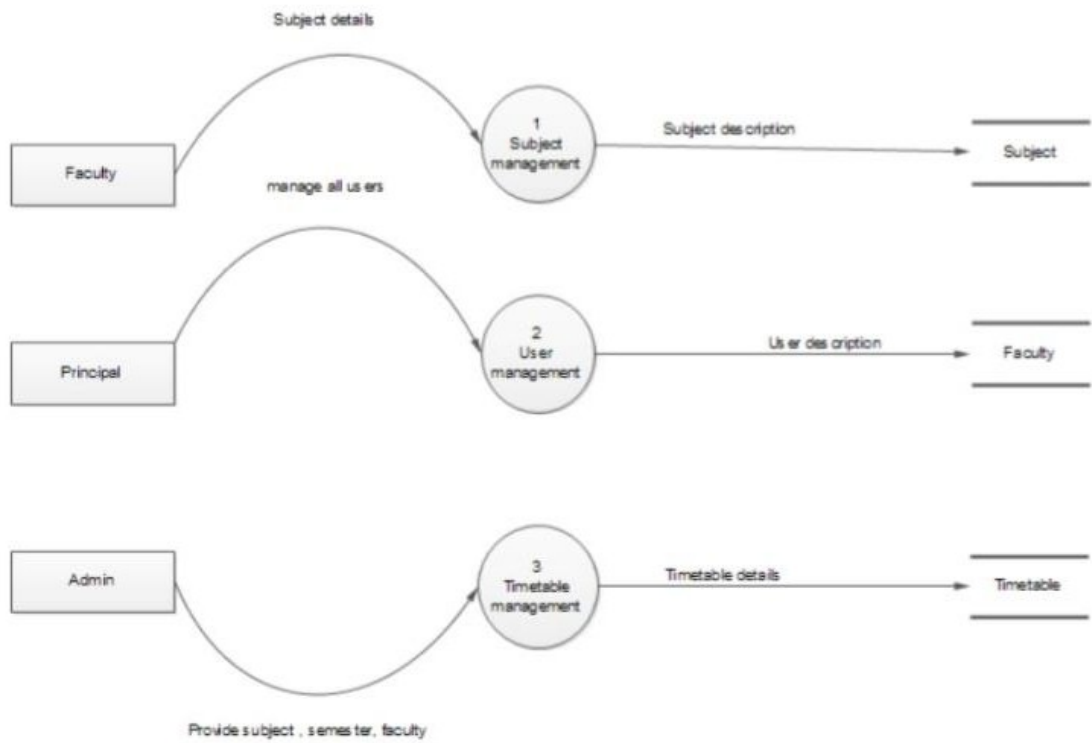
Level 0

Level 0

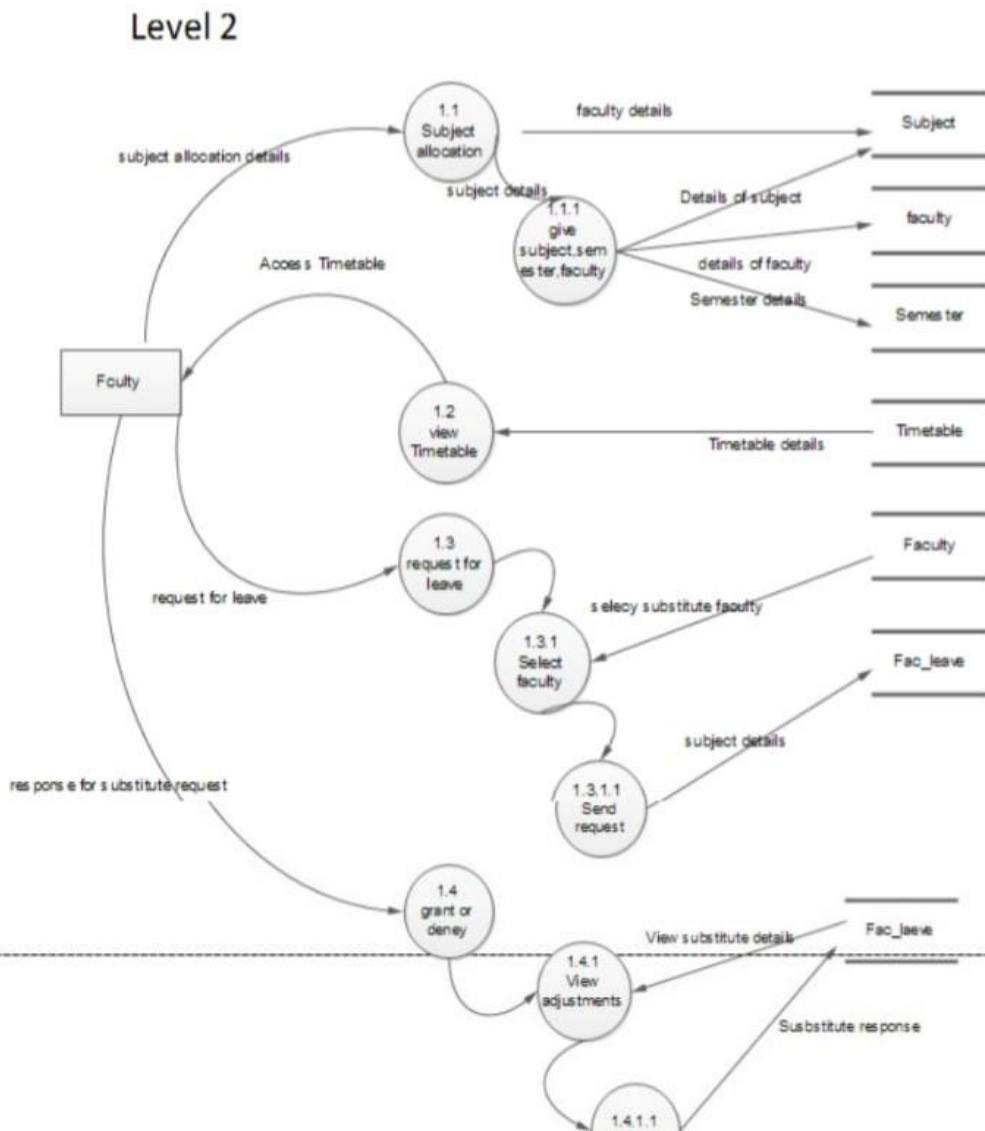


Level 1

Level 1



Level 2



Database design

The system is implemented in such a way that all the valid

information's are stored in the database. Database design converts the data model developed in logical designing to database definition. This is supported by database software. The management of data involves baoh the definition of

structure of the storage of information and provision of mechanism for the manipulation of information. In addition database system must provide safety of information handle, despite the system crashes due to attempts of unauthorized access.

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. In database design, several views of data must be considered along with the person who users them. The logical view is what the data look like, regardless of how they are stored. The physical view is the way data exist in physical storage. Tables are carefully designed aiming to achieve its main objectives ie, to make information access easy, quick, inexpensive and flexible for user.

NORMALIZATION

Normalization is the process of analyzing the given relation schemas based on their Functional Dependencies and primary keys to achieve the desirable properties of Minimizing Redundancy, Minimizing the insertion, deletion and updating anomalies.

Normalization is carried out for the following reasons:

- To structure the data so that perfect relationship between entries can be represented.
- To permit simple retrieval of data in response query and report requests.
- To reduce the need to restructure or reorganize data when new application

Normalization consists of various levels:

1. First Normal Form (1NF)

A table is in 1NF if there are no duplicate rows in the table. Each cell is single valued. Entries in a column are of the same kind.

2. Second Normal Form (2NF)

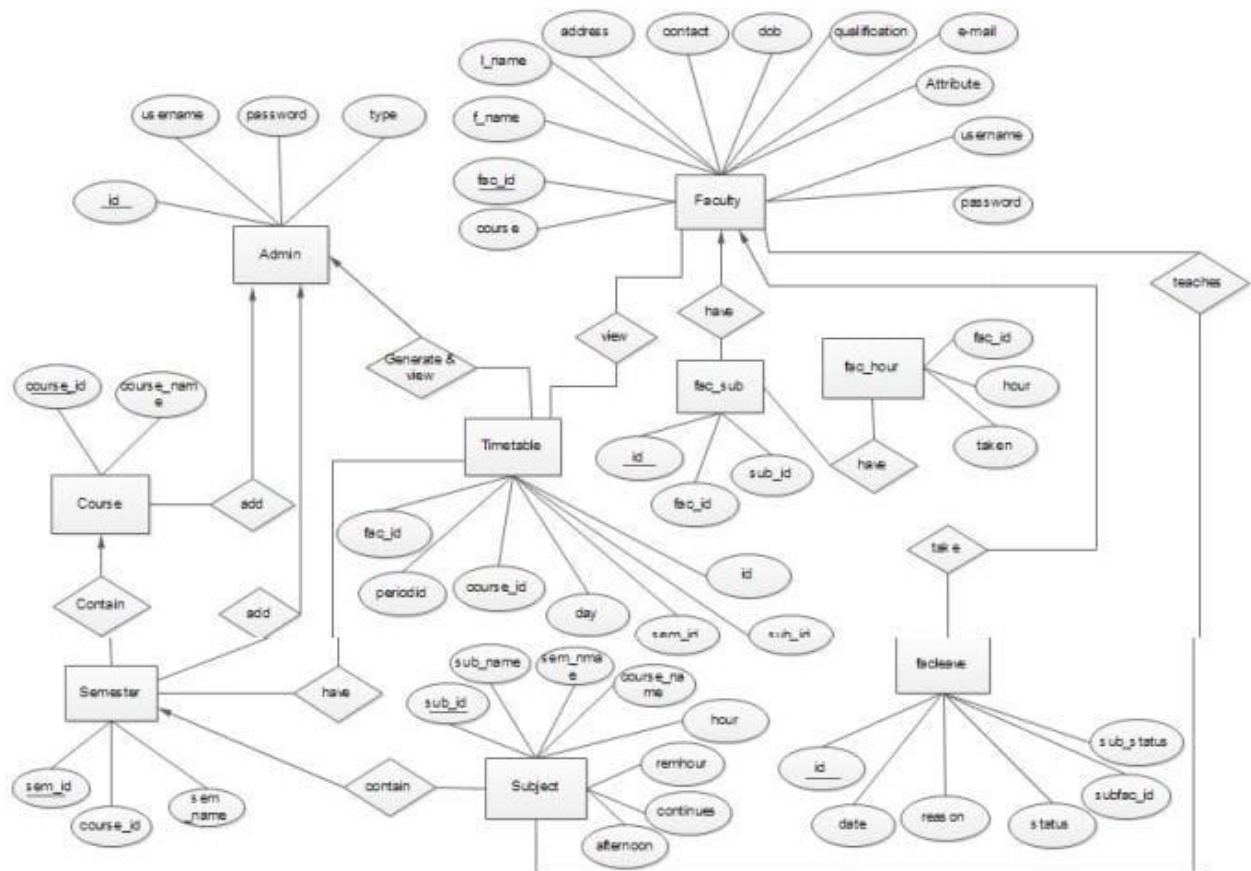
Second Normal form is based on the concept of full functional dependency. A table (relation) is in 2NF if .It is in First Normal Form and if all non-key attributes are dependent on the key. Dependent on only a part of the (composite) key, the definition of 2NF is sometimes phrased as, "A table is in 2nF if it is in 1NF and if it has no partial dependencies."

3. Third Normal Form (3NF)

Third Normal Form is based on the concept of transitive dependency.

A table (relation) is in 3NF if it is in Second Normal Form and if it has no transitive dependencies

4.2.1 E-R Diagram



FUTURE WOKS

With suitable records structures, it's far feasible to generate a hard and fast of equations that generate the timetable routinely, or can be used to check the manually generated timetable towards the constraints. Given the generality of the operation, it may further be tailored to extra particular eventualities, e.g. University, examination scheduling and in addition be better to create railway time tables. The most exciting destiny course within the development of the assignment lies in its extension to constraint propagation. Timetable problem is NP-Complete. Timetable control has been made easier with this answer. It includes an internet app to generate timetable and push to android devices.

METHODOLGY AND DISCUSSION

Timetable Generation

In our device we generate dynamic timetable. College start time and give up time isn't always fixed. Admin input start time, cease time, variety of challenge, every challenge trainer name, workload of every challenge is noted via admin. Admin can upload the challenge in database and assign teachers to situation. Timetable is generated for 2 shifts this is first and 2nd shifts. Timetable is generated with the aid of considering the elegance rooms, labs, trainer availability. Occurrences of all situation are also considered. In at some point there will not be complete lectures or practical's. Once the timetable is generated admin can import pupil and trainer contact information then timetable is des patched to all instructors and pupil through electronic mail the usage of e-mail facts that is given by way of admin.

Methodology The establishment and use of sound engineering principles in order to obtain economically developed software that is reliable and works efficiently on real machines is called software engineering. Software engineering is the discipline whose aim is: >Production of quality software >software that is delivered on time >cost within the budget >satisfies all requirements. Software process is the way in which we produce the software. Apart from hiring smart, knowledgeable engineers and buying the latest development tools, effective software development process is also needed, so that engineers can systematically use the best technical and managerial practices to successfully complete their projects. A software life cycle is the series of identifiable stages that a software product undergoes during its lifetime .A software lifecycle model is a descriptive and diagrammatic representation of the software life cycle .A life cycle model represents all the activities required to make a software product transit through its lifecycle phases .It also captures the order in which these activities are to be taken.

FEEDBACK GENERATION

Student can give the remarks for any college which can be delivered by admin. Once the comments is given through scholar the use of web page then this feedback automatically sent to sentiment API. Sentiment API will send end result to timetable and remarks era device. Feedback can be generated in three forms this is high quality, poor, neutral. Once the sentiment and confidence obtained from sentiment API admin generates performance graph for college. Teacher overall performance graph can simplest be view by using admin.

CODING AND CODE TEMPLATES

The system contains 3 users , Faculty , Principal and Admin. Each user have their own functionalities as follows .

Function : Subject allocation

Input : subject, faculty and semester

Output : timetable

Logic:

Step 1: Read subject , faculty and semester from the tables

subject,faculty and semester respectively.

Step 2:Validate the details and processing is done

Step 3:subject allocated for facultys successfully in the table named Timetable.

Function : View Timetable

Input : Subject,Semester, Faculty

Output : Timetable

Logic:

Step 1: Read subject , faculty and semester from the tables subject, faculty and semester respectively.

Step 2: Timetable generated successfully in the table called timetable.

Function : Request for leave

Input : Date,Reason, Substitute

Output : Approve / Reject request

Logic:

Step 1: Read date, reason and substitute faculty from the table fac _leave

Step 2: Get Substitute status from fac _ leave

Step 3: Request successfully sent (Approve / Reject)

If Faculty need to take a leave the he/she can apply for leave with setting Substitutes. Here Date at which leave is required , reason and substitute

Faculty should be specified.

Function : Substitute request

Input : Request

Output : Approve / Reject

Logic:

Step 1: Read the information and Apply for leave with providing date ,
reason and substitute faculty in the table fac_leave

Step 2: Substitute request successfully sent (approve/reject)

Function : Grant or deny

Input : Request

Output : Reply message send Approve / Reject

Logic:

Step 1: Access substitute request send by the faculty from
fac_leave

Step 2: Reply for Substitute request successfully sent
(approve/reject)

Function : Grant leave

Input : Request

Output : Reply message send Approve / Reject

Logic:

Step 1: Access leave request send by the faculty from fac_leave

Step 2: Reply for Request for leave successfully sent
(approve/reject)

Function : Subject allocation

Input :subject, faculty and semester

Output : timetable

Logic:

Step 1: Read subject , faculty and semester from the tables
subject,faculty and semester respectively.

Step 2:Validate the details and processing is done

Code

Apply Leave

```
package autotimetable;
import java.io.IOException;
import java.io.PrintWriter;
import java.util.ArrayList;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
public class signup extends HttpServlet {
    protected void processRequest(HttpServletRequest request,
HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    String path="index.jsp#overlay";;
    try {
    DB d=new DB();
    String uname=request.getParameter("name");
    String pass1=request.getParameter("pass1");
    String pass2=request.getParameter("pass2");
    String fname=request.getParameter("fn");
    String lname=request.getParameter("ln");
    String addr=request.getParameter("addr");
    String contactno=request.getParameter("cn");
    String dob=request.getParameter("dob");
    String doj=request.getParameter("doj");
    String qualific=request.getParameter("qu");
    String email=request.getParameter("email");
    String maddr=request.getParameter("maddr");
    String course=request.getParameter("course");
    String subjectlist[]=request.getParameterValues("subjectslist");
    String slist="";
    for(int i=0;i<subjectlist.length;i++){
    // System.err.println("subjectlist :"+subjectlist[i]);
    slist=slist+subjectlist[i]+" ";
    }
    ArrayList<String> a=new ArrayList<String>();
    a.add(uname);
    a.add(pass1);
    a.add(pass2);
    a.add(fname);
```

```

a.add(lname);
a.add(addr);
a.add(contactno);
a.add(dob);
a.add(doj);
a.add(qualific);
a.add(email);
a.add(maddr);
a.add(course);
a.add(slist);
d.addTeacher(a);
path="index.jsp?success";
} finally {

}
response.sendRedirect(path);
}
protected void doGet(HttpServletRequest request, HttpServletResponse
response)
throws ServletException, IOException {
processRequest(request, response);
}
protected void doPost(HttpServletRequest request, HttpServletResponse
response)
throws ServletException, IOException {
processRequest(request, response);
}
public String getServletInfo() {
return "Short description";
}
}
Substitute request
<%@page import="java.sql.ResultSet"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<jsp:useBean id="object" class="autotimetable.DB" />
<%
ResultSet r=object.getSubst(session.getAttribute("id")+""");
out.print("<table id='admin_menu'>");
out.print("<th>");
out.print("</th>");
out.print("<th>");
out.print("</th>");

```

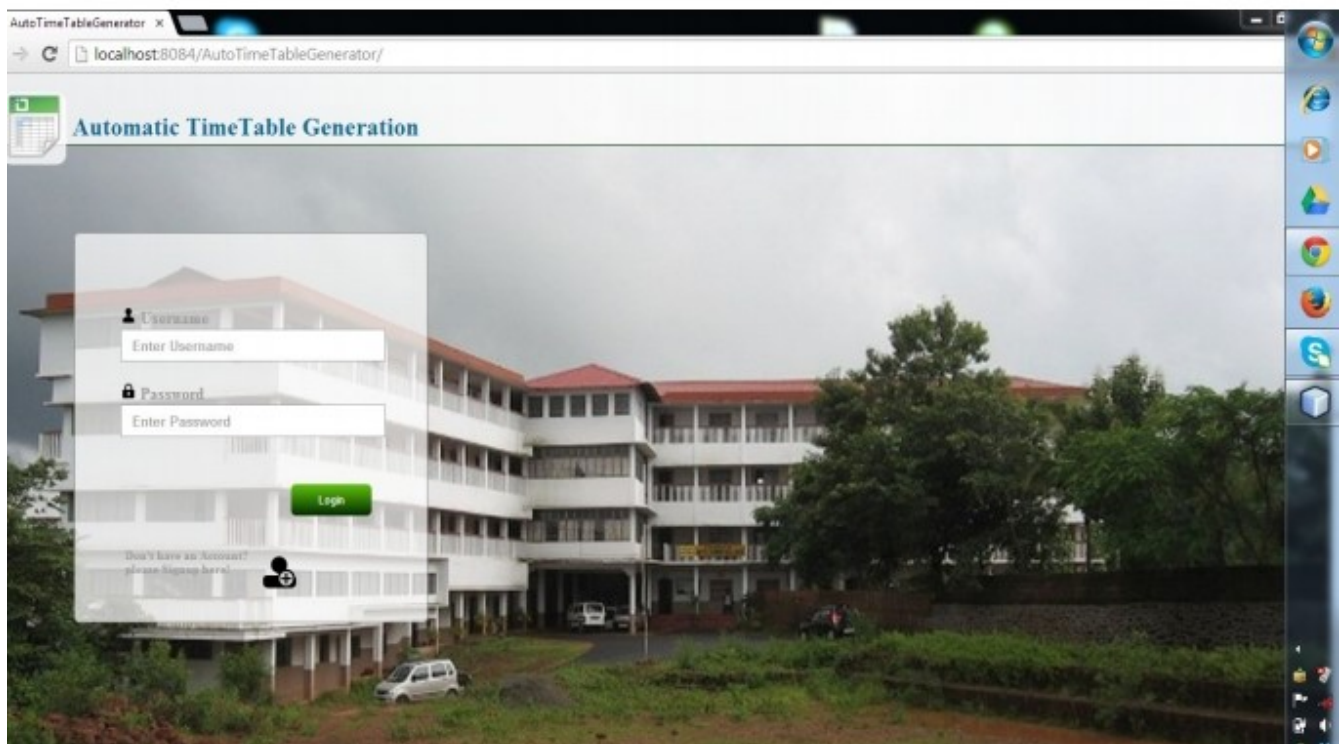
```

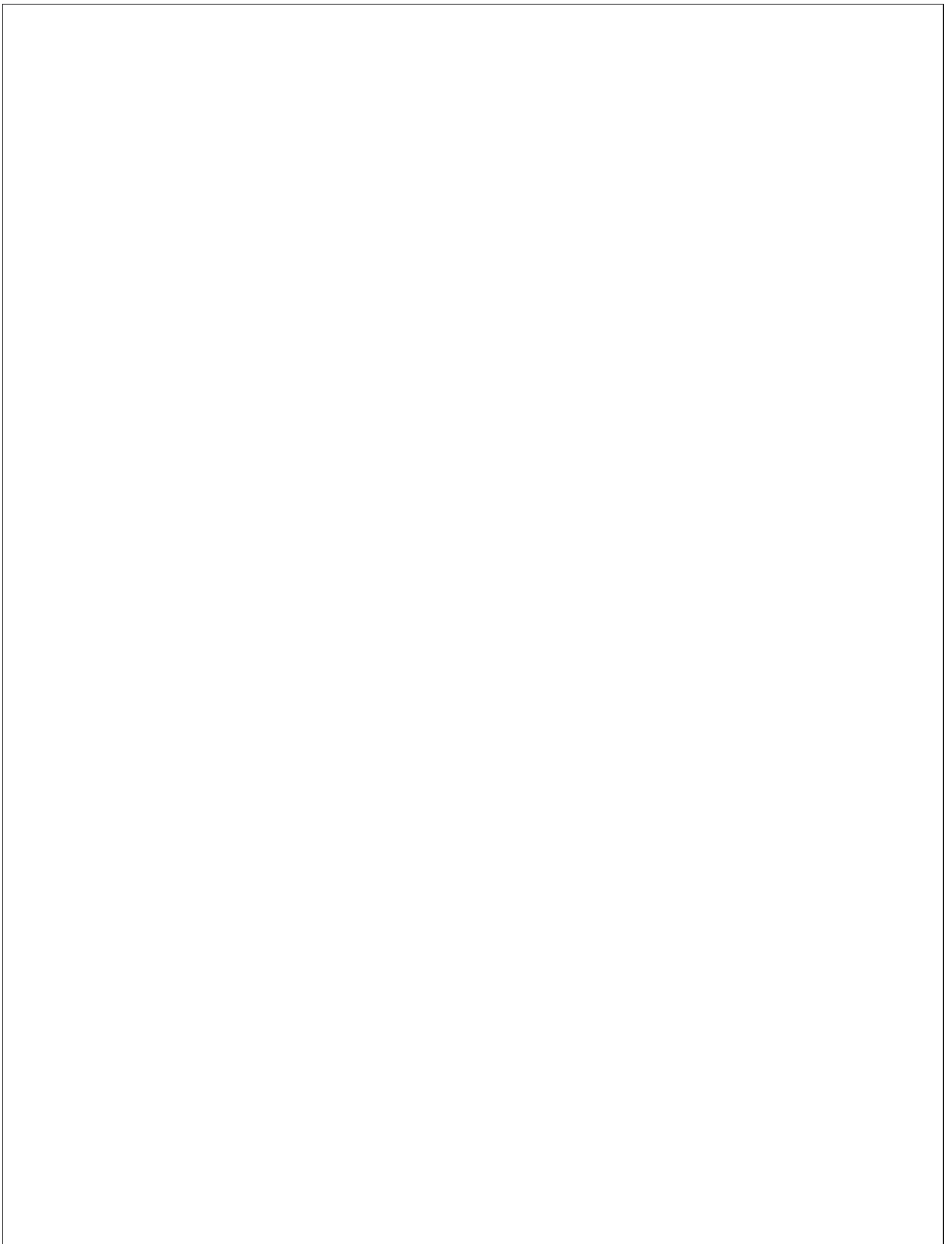
out.print("<th>");
out.print("Requested By</th>");
out.print("<th>");
out.print("Requested For</th>");
out.print("<th>");
while(r.next()){
out.print("<tr>");
out.print("<td>");
out.print("<div style='padding: 10px 10px 10px 10px; background-
color:green'><a href='user/leaveAction.jsp?id="+session.getAttribute("id")
+"&date="+r.getString(2)+"&action=approved' style='text-
decoration:none;color:white'>Approve</a></div>");
out.print("</td>");
out.print("<td>");
out.print("<div style='padding: 10px 10px 10px 10px; background-
color:Red'><a href='user/leaveAction.jsp?id="+session.getAttribute("id")
+"&date="+r.getString(2)+"&action=Rejected' style='text-
decoration:none;color:white'>Reject</a></div>");
out.print("</td>");

```

7. OUTPUT SCREENS

Login Page





Days	period 1	period 2	period 3	period 4	period 5	period 6
Monday	DS Amritha	C++ Bimal	DCN Priya	LAB2 Kavitha	LAB2 Kavitha	LAB2 Kavitha
Tuesday	POA Sandhya	DBMS Shajeer	LINUX AD: Dhanya	SEMINAR Jaya	SEMINAR Jaya	SEMINAR Jaya
Wednesday	DS Amritha	DBMS Shajeer	LINUX AD: Dhanya	DCN Priya	C++ Bimal	POA Sandhya
Thursday	POA Sandhya	DBMS Shajeer	LINUX AD: Dhanya	DCN Priya	C++ Bimal	DS Amritha
Friday	POA Sandhya	DCN Priya	C++ Bimal	DS Amritha	DBMS Shajeer	LINUX AD: Dhanya

TESTING

Validation and checks Software validation is achieved through a series of tests that demonstrate conformity with requirements. Validation succeeds when software functions in a manner that can be reasonably expected by the customer. Here line by line checking is used to find errors. Comment line facility is used for checking errors. Testing is necessary for the success of the system. During testing, program to be tested is executed with a set of test data and the output of the program for test data is evaluated to determine if the programs are performing as expected. Validation means checking the quality of software in both simulated and live environments. System validation ensures that the user can in fact match his/her claims, especially system performance. True validation is verified by having each system tested.

First the application goes through a phase often referred as alpha testing in which the errors and failures based on simulated user requirements are verified and studied. The modified software is then subjected to phase two called beta testing in the actual user's site or live

environment. After a scheduled time, failures and errors are documented for final correction and enhancements are made before the package is released.

In a software development project, errors can be injected at any stage during development. Even if error detecting and eliminating techniques were

employed in the previous analysis and design phases, errors are likely to remain undetected. Unfortunately, these errors will be reflected in the code. Since code is frequently the only product that can be executed and whose actual behavior can be observed, testing is the phase where the errors remaining from the earlier phases must be detected in addition to detecting the errors introduced during coding activity.

Having proper test cases is central to successful testing. We would like to determine a set of test cases such that successful execution of all of them implies that there are no errors in the program. Therefore, our project crew aimed at selecting the test cases such that the maximum possible numbers of errors are detected by the minimum number of test cases.

For this we have adopted both manual testing techniques and automated testing techniques. First and foremost, testing was done by means of Inspection, where participants manually examine system deliverables for occurrences of well-known errors. Inspection team consists of 5 members who are trained for

their tasks. Items for inspection include completeness of the design, and functional requirements, internal completeness and consistency in definition and usage of terminology, and correctness of the interfaces between modules. Syntax, grammar, and some other routine errors were checked by automated inspection software, so manual inspection checks are used for more subtle errors. And we have done the testing following design phase prior to implementation.

Testing objectives are :

- Testing is a process of executing a program with the intent of finding an error .

- A good test case is one that has a high probability of finding an as-yet-undiscovered error .
 - A successful test is one that uncovers an as-yet-undiscovered error.
- Automated testing is performed at four different levels

Test Types

1. Unit testing
2. Integration testing
3. System testing
4. Acceptance testing

Unit testing

In unit testing different modules are tested against the specifications produced during design phase for the modules in the project and the goal is to test the internal logic of the modules.

In order to perform the unit testing, the best approach we adopted in this project is functional testing in which inputs are given to the system for which the expected results are known, boundary values and special values. For this the module selected was advance details where the balance amount falls to negative indicating there is no more payment required. Secondly performance testing was done to determine the amount of execution time spent in various aspects like the module, program throughput, response time, and the device utilization by the program unit.

Integration testing

The primary goal of the integration testing is to see if the modules can be integrated properly. The integration testing is therefore considered as testing the design. Thus in the integration testing step, all the errors uncovered are corrected for the next testing steps.

System testing

System testing is similar to integration testing, but instead of integrating modules into programs for testing, programs are integrated into systems for testing the interfaces between programs in a system. System testing can be defined in many ways, but a simple definition is that validation succeeds when the software functions in a manner that can be reasonably expected by the customer.

Acceptance testing

Once the system tests have been satisfactory completed, the system is ready for acceptance testing. Acceptance testing is the process whereby actual users test a completed information system in the environment where it will eventually be used, the end result of which is the user's acceptance or rejection. The admin and staff at Company accepted proposed system after testing

DEFAULTER LIST GENERATION

For Defaulter listing era admin will import the attendance sheet that is in csv layout. Once the attendance sheet imported with the aid of admin system routinely despatched alert mail to defaulter student. Student who have attendance much less than seventy five% the ones students are taken into consideration as defaulter pupil. Alert mail can also be sent to defaulter students parent's.

SCOPE

1. Automatic timetable generation by considering hard and soft constraints.
2. Feedback generation by using Mashape API.
3. Generate defaulter list from attendance sheet.

REFERENCES

- [1] M. Doulaty, M. R. FeiziDerakhshi, and M. Abdi, "Timetabling: A State-of-the-Art Evolutionary Approach", International Journal of Machine Learning and Computing, Vol. Three, No. Three, June 2013.
- [2] Anirudha Nanda, Manisha P. Pai, and AbhijeetGole, "An Algorithm to Automatically Generate Schedule for School Lectures Using a Heuristic Approach", International Journal of Machine Learning and Computing, Vol. 2, No. Four, August 2012
- [3] DilipDatta, Kalyanmoy Deb, Carlos M. Fonseca, "Solving Class Timetabling Problem of IIT Kanpur the usage of Multi-Objective Evolutionary Algorithm".KanGAL 2005.
- [4] AnujaChowdhary, PriyankaKakde, ShrutiDhoke, SonaliIngle, RupalRushiya, Dinesh Gawande, "Time table Generation System", International Journal of Computer Science and Mobile Computing, Vol.3 Issue.2, February- 2014.