Contents

1 INTRODUCTION

1.1 Project Profile

Currently exam cell activity mostly includes a lot of manual calculations and is mostly paper based. The project aims to bring in a centralized system that will ensure the activities in the context of an examination that can be effectively managed. There will be total four semesters where each semester contains maximum of seven subjects. First, admin login into the system. Then student details is added.semester number and batch entry is done. Exam time table is obtained from the ktu website and the admin is added it. Staff details is entered by the admin. Duty list is assigned by the admin according to the exam time table. Then the room arrangement is done and admin can select the arrangement pattern. In each room only 30 students are allocated. For each room, one staff is assigned. The seat arrangement is done according to the register number and it can be changed daily by admin. Student attendance and staff attendance is calculated and absent report is generated. Remuneration is calculated according to the staff attendance.Admin has full ontrol over the sytem. It contain only one admin module.

2 ABOUT THE DEVELOPING TOOLS

2.1 Introduction to JSP

JSP technology is used to create dynamic web applications. JSP pages are easier to maintain then a Servlet. JSP pages are opposite of Servlets as a servlet adds HTML code inside Java code, while JSP adds Java code inside HTML using JSP tags. Everything a Servlet can do, a JSP page can also do it. JSP enables us to write HTML pages containing tags, inside which we can include powerful Java programs. Using JSP, one can easily separate Presentation and Business logic as a web designer can design and update JSP pages creating the presentation layer and java developer can write server side complex computational code without concerning the web design. And both the layers can easily interact over HTTP requests.

Advantage of JSP over Servlet

1. Extension to Servlet
   JSP technology is the extension to servlet technology. We can use all the features of servlet in JSP.

2. Easy to maintain
   JSP can be easily managed because we can easily separate our business logic with presentation logic

3. Fast Development
   If JSP page is modified, we don't need to recompile and redeploy the project.

4. Less code than Servlet
   In JSP, we can use a lot of tags such as action tags, jstl, custom tags etc. that reduces the code.

2.2 HeidiSQL

HeidiSQL is a useful and reliable tool designed for web developers using the popular MySQL server, Microsoft SQL databases and PostgreSQL. It enables you to browse and edit data, create and edit tables, views, procedures, triggers and scheduled events. Also, you can export structure and data either to SQL file, clipboard or to other servers. Its codebase was originally taken from Ansgar Becker's own MySQL-Front 2.5 software. Due to having sold the MySQL-Front branding to an unrelated party, Becker chose "HeidiSQL" as a replacement. The name was suggested by a friend as a tribute to Heidi Klum, and was further reinforced by Becker's own nostalgia for Heidi, Girl of the Alps. HeidiSQL is a useful and reliable tool designed for web developers using the popular MySQL server, Microsoft SQL databases and PostgreSQL. It enables you to browse and edit data, create and edit tables, views, procedures, triggers and scheduled events. Also, you can export structure and data either to SQL file, clipboard or to other servers. HeidiSQL is a free and open-source administration tool for MySQL and its forks, as well as Microsoft SQL Server and PostgreSQL. Its codebase was originally taken from Ansgar Becker's own MySQL-Front 2.5 software. Due to having sold the MySQL-Front branding to an unrelated party, Becker chose "HeidiSQL" as a replacement. The name was suggested by a friend as a tribute to Heidi Klum, and was further reinforced by Becker's own nostalgia for Heidi, Girl of the Alps.A version written in Java, jHeidi, was designed to work on Mac and Linux computers. It was discontinued in March 2010 in favor of Wine support.

2.3 Netbeans

NetBeans IDE is a free, open source, integrated development environment (IDE) that enables you to develop desktop, mobile and web applications. The IDE supports application development in various languages, including Java, HTML5, PHP and C++. The IDE provides integrated support for the complete development cycle, from project creation through debugging, profiling and deployment. The IDE runs on Windows, Linux, Mac OS X, and other UNIX-based system.The IDE provides comprehensive support for JDK 7 technologies and the most recent Java enhancements. It is the first IDE that provides support for JDK 7, Java EE 7, and JavaFX 2. The IDE fully supports Java EE using the latest standards for Java, XML, Web services, and SQL and fully supports the GlassFish Server, the reference implementation of Java EE.

2.4 GitHub

Version control is a system that manages changes to a file or files.These changes are kept as logs in a history, with detailed information on what file(s) was changed, what was changed within the file, who changed it, and a message on why the change was made. This is extremely useful, especially when working in teams.To understand how incredibly powerful version control is How many files of different versions of a manuscript or thesis do you have laying around after getting feedback from your supervisor or co-authors? Have you ever wanted to experiment with your code or your manuscript and need to make a new file so that the original is not touched ?Have you ever deleted something and wish you hadnt ? Have you ever forgotten what you were doing on a project ? All these problems are fixed by using version control (git).

HOW TO ENTER THE PROJECT VERSION TO GIT

1.Create a new repository on GitHub. To avoid errors, do not initialize the new repository with README, license, or gitignore files. You can add these files after your project has been pushed to GitHub.
2.Open Git Bash
3.Change the current working directory to your local project.
4.Initialize the local directory as a Git repository

```
$ git init
```

5.Add the files in your new local repository. This stages them for the first commit.

```
$ git add
```

6. Commit the files that you've staged in your local repository

```
$ git commit -m "First commit"
```

7.At the top of your GitHub repository's Quick Setup page, click to copy the remote repository URL.
8.In the Command prompt, add the URL for the remote repository where your local repository will be pushed

```
$git remote add origin remote repository URL
$git remote -v
```

9.Push the changes in your local repository to GitHub

```
$git push origin master
```

2.5 GlassFish

GlassFish is an open-source application server project started by Sun Microsystems for the Java EE platform and now sponsored by Oracle Corporation. The supported version is called Oracle GlassFish Server. GlassFish is free software, dual-licensed under two free software licences the Common Development and Distribution License (CDDL) and the GNU General Public License (GPL). GlassFish is the reference implementation of Java EE .

2.6 HTML

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript it forms a triad of cornerstone technologies for the World Wide Web.[3] Web browsers receive HTML documents from a web server or from local storage and render them into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects, such as interactive forms, may be embedded into the rendered page. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as ¡img /¿ and ¡input /¿ introduce content into the page directly. Others such as ¡p¿...¡/p¿ surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML can embed programs written in a scripting language such as JavaScript which affect the behavior and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), maintainer of both the HTML and the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.[4] In 1980, physicist Tim Berners-Lee, a contractor at CERN, proposed and prototyped ENQUIRE, a system for CERN researchers to use and share documents. In 1989, Berners-Lee wrote a memo proposing an Internet-based hypertext system.[5] Berners-Lee specified HTML and wrote the browser and server software in late 1990. That year, Berners-Lee and CERN data systems engineer Robert Cailliau collaborated on a joint request for funding, but the project was not formally adopted by CERN. In his personal notes[6] from 1990 he listed[7] "some of the many areas in which hypertext is used" and put an encyclopedia first.

The first publicly available description of HTML was a document called "HTML Tags", first mentioned on the Internet by Tim Berners-Lee in late 1991.[8][9] It describes 18 elements comprising the initial, relatively simple design of HTML. Except for the hyperlink tag, these were strongly influenced by SGMLguid, an in-house Standard Generalized Markup Language (SGML)-based documentation format at CERN. Eleven of these elements still exist in HTML 4.

HTML is a markup language that web browsers use to interpret and compose text, images, and other material into visual or audible web pages. Default characteristics for every item of HTML markup are defined in the browser, and these characteristics can be altered or enhanced by the web page designer's additional use of CSS. Many of the text elements are found in the 1988 ISO technical report TR 9537 Techniques for using SGML, which in turn covers the features of early text formatting languages such as that used by the RUNOFF command developed in the early 1960s for the CTSS (Compatible Time-Sharing System) operating system: these formatting commands were derived from the commands used by typesetters to manually format documents. However, the SGML concept of generalized markup is based on elements (nested annotated ranges with attributes) rather than merely print effects, with also the separation of structure and markup; HTML has been progressively moved in this direction with CSS.

2.7 Java

Java is a set of computer software and specifications developed by Sun Microsystems, which was later acquired by the Oracle Corporation, that provides a system for developing application software and deploying it in a cross-platform computing environment. Java is used in a wide variety of computing platforms from embedded devices and mobile phones to enterprise servers and supercomputers. Java applets, which are less common than standalone Java applications, run in secure, sandboxed environments to provide many features of native applications and can be embedded in HTML pages.

Writing in the Java programming language is the primary way to produce code that will be deployed as byte code in a Java virtual machine (JVM); byte code compilers are also available for other languages, including Ada, JavaScript, Python, and Ruby. In addition, several languages have been designed to run natively on the JVM, including Scala, Clojure and Apache Groovy. Java syntax borrows heavily from C and C++, but object-oriented features are modeled after Smalltalk and Objective-C.[10] Java eschews certain low-level constructs such as pointers and has a very simple memory model where every object is allocated on the heap and all variables of object types are references. Memory management is handled through integrated automatic garbage collection performed by the JVM.

On November 13, 2006, Sun Microsystems made the bulk of its implementation of Java available under the GNU General Public License (GPL). The latest version is Java 9, the second of the two supported (with e.g. security updates) versions as of 2017. Oracle (and others) has announced that using older versions (other than Java 8) of their JVM implementation presents serious risks, due to unresolved security issues.

The heart of the Java platform is the concept of a "virtual machine" that executes Java bytecode programs. This bytecode is the same no matter what hardware or operating system the program is running under. There is a JIT (Just In Time) compiler within the Java Virtual Machine, or JVM. The JIT compiler translates the Java bytecode into native processor instructions at run-time and caches the native code in memory during execution.

The use of bytecode as an intermediate language permits Java programs to run on any platform that has a virtual machine available. The use of a JIT compiler means that Java applications, after a short delay during loading and once they have "warmed up" by being all or mostly JIT-compiled, tend to run about as fast as native programs.[16][17][18] Since JRE version 1.2, Sun's JVM implementation has included a just-in-time compiler instead of an interpreter. Although Java programs are cross-platform or platform independent, the code of the Java Virtual Machines (JVM) that execute these programs is not. Every supported operating platform has its own JVM.

2.8 Java script

JavaScript (/dvskrpt/[6]), often abbreviated as JS, is a high-level, dynamic, weakly typed, prototype-based, multi-paradigm, and interpreted programming language. Alongside HTML and CSS, JavaScript is one of the three core technologies of World Wide Web content production. It is used to make webpages interactive and provide online programs, including video games. The majority of websites employ it, and all modern web browsers support it without the need for plug-ins by means of a built-in JavaScript engine. Each of the many JavaScript engines represent a different implementation of JavaScript, all based on the ECMAScript specification, with some engines not supporting the spec fully, and with many engines supporting additional features beyond ECMA.

As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative (including object-oriented and prototype-based) programming styles. It has an API for working with text, arrays, dates, regular expressions, and basic manipulation of the DOM, but the language itself does not include any I/O, such as networking, storage, or graphics facilities, relying for these upon the host environment in which it is embedded.

Initially only implemented client-side in web browsers, JavaScript engines are now embedded in many other types of host software, including server-side in web servers and databases, and in non-web programs such as word processors and PDF software, and in runtime environments that make JavaScript available for writing mobile and desktop applications, including desktop widgets.

Although there are strong outward similarities between JavaScript and Java, including language name, syntax, and respective standard libraries, the two languages are distinct and differ greatly in design; JavaScript was influenced by programming languages such as Self and Scheme.

3 SYSTEM ANALYSIS

3.1 Introduction

System Analysis works with users to identify goals and build system to achieve them. System Analysis is an important phase of any system development process. System analysis is a step-by-step process used to identify and develop or acquire the software need to control the processing of specific application. System analysis is a continuing activity the stages of the systems development. The system is studied to the minutes details and analyzed. In analysis, a detailed study of these operation performed by a system and their relationships within and outside of the system is done The aim of the

proposed system is to develop a system with improved facilities. The proposed system can overcome all the limitation of the existing system, such as students information is maintained in the database, it gives more security to data, ensures data accuracy, reduces paper work and save time, only eligible students get chance, it makes information flow efficient and paves way for easy report generation, reduce the space. proposed system is cost effective.

3.2 Existing System

Currently exam cell activity mostly includes a lot of manual calculations and is mostly paper based. It is a time consuming process.

3.3 Feasibility Study

A feasibility study is needed to determine if a project or end result of a project is feasible and beneficial. The main objective of feasibility study is to test the technical, social and economic feasibility of developing a new computer system. Investigating the existing system in the areas under investigation and generating ideas about a new system does this. The key considerations involved in the feasibility analysis are the following: 1. Economic feasibility 2. Technical feasibility 3. Operational feasibility.

3.3.1 Economic Feasibility

Economic feasibility is a method for evaluating the effectiveness of a candidate system. This study is mainly concerned with cost-benefit analysis that is how much money the user is investing in any system and how much he is getting as a benefit in output. Our project is Economical Feasible because anyone uses this software would need only to buy the machine. Our hardware requirement is not too expensive. The money and human effort needed for the existing system is high .In the new system benefits outweigh costs. So as compare to cost the project is economically feasible. We conduct an economic feasibility study for this exam seat mapping system and it also uses minimum hardware requirements that are already used in the existing system .In existing system manual records are used for storing details. The system is cost effective because of its compatibility and effort saving nature. The cost benefit ratio is very small and hence the proposed system is feasible.

3.3.2 Technical Feasibility

Technical feasibility includes whether the technology is available in the market for the development and its availability. The assessment of technical feasibility must be based on an outline design of the system requirements in terms of input, output, files, programs and procedures. This study checks the technical aspects of system. Minimum requirements of the proposed system are a computer and internet connectivity, which will not add any additional expense in implementing the system. This software is simple to use and manage.Online Freelancer system also uses the minimum technologies for the creation of the web based application. The existing system has also required minimum technical requirements. So the proposed system is said to be technically feasible

3.3.3 Operational Feasibility

The new system is very much easier and user friendly than the existing system. It satisfies the requirements identified in the requirements analysis phase of system development. It reduces the operational time considerably. Operational cost is very less. The maintenance and modification of the new system needs very less human effort. Using command buttons throughout the application programs enhances the operational feasibility. The new system is operationally feasible and makes the operations simpler and quite easier. The proposed system exam seat mapping system does not produce any problem to existing customers, organization etc. It reduces the drawback of existing system. All these reasons make the new system operationally feasible.

3.4 Proposed system

The project aims to bring in a centralized system that will ensure the activities in the context of an examination that can be effectively managed. Admin has the full control over the system.

We all know the importance of computerization. The world is moving ahead at lightning speed and everyone is running short of time. One always wants to get the information and perform a task he/she/they desire(s) within a short period of time and too with amount of efficiency and accuracy. The application areas for the computerization have been selected on the basis of following factors:

4 FACT FINDING TECHNIQUES

The success of any project depends upon the accuracy of available data. Accurate information can be collected with the help of certain methods / techniques. These specific methods for finding information of the system are termed as fact finding techniques. Interview, Questionnaire, Record View and Observations are the different fact finding techniques used in this project.

4.1 Interview

This method is used to collect the information from groups or individuals. We select the people who are related with the system for the interview. In this method, we sit face to face with people and record their responses.

4.2 Record View

The information related to the system is available in the source like companys documents, websites and other records. This record review helped me to get valuable information about the system.

4.3 Onsite observation

Unlike the other fact finding techniques, in this method we visit the organization and observe and understand the working of the existing system, flow of the system, the users of the system etc.

5 SYSTEM SPECIFICATION

5.1 Hardware Specification

- Processor : Pentium III
- Speed : 1.1 Ghz
- RAM : 256 MB(min)
- Hard Disk : 20 GB
- Floppy Drive : 1.44 MB
- Key Board : Standard Windows Keyboard
- Mouse : Two or Three Button Mouse

5.2 Software Specification

- Operating System : Windows 7
- Application Server : GlassFish Server
- Front End : HTML, Java
- Scripts : JavaScript
- Server side Script : Java Server Pages
- Database : HeidiSQL
- Database Connectivity : JDBC

6 SYSTEM DESIGN

6.1 Introduction of System Design

In this project design technique used is top-down, object- oriented dynamic modeling technique. A top-down design approach starts by identifying the major components and iterating until the desired level of details is achieved. In object oriented design technique, the modules in the design represent data abstraction. A dynamic model aim to specify new the state of various objects changes as events occur

6.2 Input Design

Input design is a part of overall system design, which requires very careful attention. Input design features can ensure the reliability of the system and produce result from accurate data, or they can result in the production or erroneous information. The input design also determines whether the user can interact efficiently with the system. Admin who was a person which they can add student to the system. .semester number and batch entry is done. Exam time table is obtained from the ktu website and the admin is added it. Staff details is entered by the admin. Duty list is assigned by the admin according to the exam time table.

6.3 Output Design

One of the important features of an information system for users is the output produces. Output is the information delivered to users through the information system. Output design is very important phase because the output will be interactive manner. In order to create the most useful output possible. To make a user friendly output and for better communication the programmer can use the features of a window. admin can view the student details,faculty details,subject details,exam duty list,seat arrangement and remuneration details

6.4 Normalization

Normalization is the process of efficiently organizing data in a database. There are two goals of the normalization process: eliminating redundant data (for example, storing the same data in more than one table) and ensuring data dependencies make sense (only storing related data in a table). Both of these are worthy goals as they reduce the amount of space a database consumes and ensure that data is logically stored.

The database community has developed a series of guidelines for ensuring that databases are normalized. These are referred to as normal forms and are numbered from one (the lowest form of normalization, referred to as first normal form or 1NF) through five (fifth normal form or 5NF). In practical applications, you'll often see 1NF, 2NF, and 3NF along with the occasional 4NF. Fifth normal form is very rarely seen.

6.4.1 First Normal Form(1NF)

First normal form (1NF) sets the very basic rules for an organized database.Create separate tables for each group of related data and identify each row with the primary key.

6.4.2 Second Normal Form(2NF)

Second normal form (2NF) further addresses the concept of removing duplicative data.

6.4.3 Third Normal Form(3NF)

Third normal form (3NF) goes one large step further .Meet all the requirements of the second normal form and remove columns that are not dependent upon the primary key.

6.4.4 Boyce-Codd Normal Form (BCNF or 3.5NF)

The Boyce-Codd Normal Form, also referred to as the "third and half (3.5) normal form.

6.4.5 Fourth Normal Form(4NF)

Finally, fourth normal form (4NF) has one additional requirement.A relation is in 4NF if it has no multi-valued dependencies.

6.5 Database Design

Database design is the process of producing a detailed data model of database. This data model contains all the needed logical and physical design choices and physical storage The process of doing database design generally consists of a number of steps which will be carried out by the database designer. Usually, the designer must:

- Determine the data to be stored in the database.
- Determine the relationships between the different data elements.
- Superimpose a logical structure upon the data on the basis of these relationships.

In this project database design generally the data is to be stored in the database whether it can more relation for each modules. And it provides the logical relation between them.

6.6 Architectural Design

Architectural design is of crucial importance in software engineering during which the essential requirements like reliability, cost, and performance are dealt with. Architectural design is the responsibility of developers, some other people like user representatives, systems engineers, hardware engineers, and operations personnel are also involved. All these stakeholders must also be consulted while reviewing the architectural design in order to minimize the risks and errors.

In Ilahia Exam cell automation,it is fully managed by the admin and all activities are done by admin

6.7 System Modules

1. ADMIN MODULE

   Admin has the full control over the system.First, admin login into the system. Then student details is added .semester number andbatch entry is done. Exam time table is obtained from the ktu website and the admin is added it. Staff details is entered by the admin. Duty list is assigned by the admin according to the exam time table.

   Room arrangement is done and admin can select the seat arrangement pattern. Admin assigns staffs for each room. The seat arrangement is done according to the register number and it can be changed daily by admin. Student attendance and staff attendance is calculated and absent report is generated. Remuneration is calculated according to the staff attendance.

6.8 Form Design

A form designing means deciding the contents and layout of forms for the purpose of collecting and processing the required information economically and efficiently. The importance of forms designing can be understood because of the following points:

1. Forms are used to collect record and communicate the required information according to the expectations of the needy persons. Therefore, forms are treated as tools of office work. If the forms are badly designed, it reduces the speed of operation of office work.

2. The forms create psychological impact on the people who use it. The people may be frustrated and get tired if the forms are not designed properly.

3. The badly designed forms results in more number of mistakes in clerical work. Hence, there is a need of well-designed forms to avoid mistakes in clerical work.

4. Sometimes, the designed form may project a poor image in the minds of the customers. This may adversely affect the good will of the company.

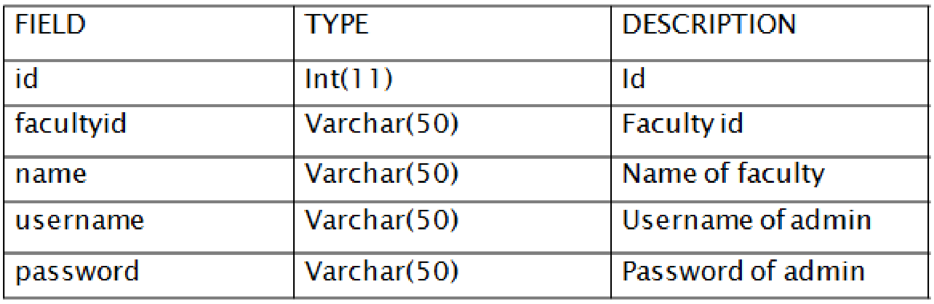
5. System is the basis for form design. Hence, forms are designed according to the needs of the system. If forms are badly designed, they can ruin a whole system.

6. The well-designed forms contribute much to the efficiency of employees of an organization and efficiency of the system.

7. The cost of forms is less than the cost of completing office forms, transporting and filling of office forms. The ratio will be greater if the forms are badly designed.

6.9 Table Structure

Table No:1
Table name: Tbl_teacher
Primary key:id



FIELD	TYPE	DESCRIPTION
id	Int(11)	Id
facultyid	Varchar(50)	Faculty id
name	Varchar(50)	Name of faculty
username	Varchar(50)	Username of admin
password	Varchar(50)	Password of admin

Table No:2

Table name: Tbl_subject

Primary key:id

FIELD	TYPE	DESCRIPTION
id	Int(11)	Subject id
subjectname	Varchar(50)	Subject name
semester	Int(11)	Semester number

Table No:3

Table name: Tbl_semester

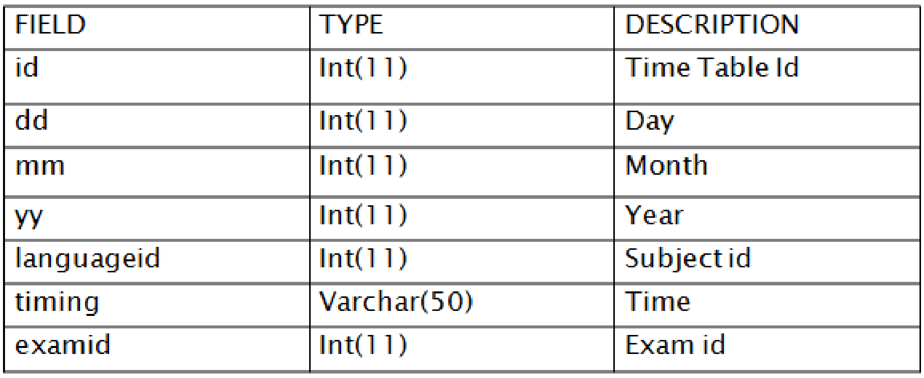
Primary key:id

FIELD	TYPE	DESCRIPTION
id	Int(11)	Semester id
sem	Varchar(10)	Semester number

Table No:4

Table name: Tbl_examtimetable

Primary key:id

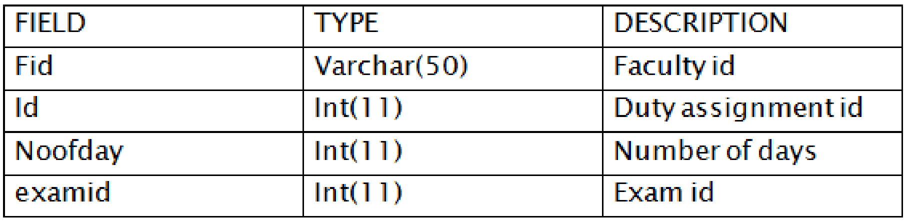


FIELD	TYPE	DESCRIPTION
id	Int(11)	Time Table Id
dd	Int(11)	Day
mm	Int(11)	Month
yy	Int(11)	Year
languageid	Int(11)	Subject id
timing	Varchar(50)	Time
examid	Int(11)	Exam id

Table No:5

Table name: Tbl_dutyassign

Primary key:id



FIELD	TYPE	DESCRIPTION
Fid	Varchar(50)	Faculty id
Id	Int(11)	Duty assignment id
Noofday	Int(11)	Number of days
examid	Int(11)	Exam id

Table No:6

Table name: Tbl_student

Primary key:id

FIELD	TYPE	DESCRIPTION
id	Int(11)	Id
studid	Varchar(50)	Student id
rollno	Varchar(50)	Rollnumber
ktuid	Varchar(50)	Ktu id
name	Varchar(50)	Student name
classid	Varchar(50)	Class id
batchid	Varchar(50)	Foreign Key

Table No:7

Table name: Tbl_batch

Primary key:id

FIELD	TYPE	DESCRIPTION
id	Int(11)	Batch id
batchname	Varchar(50)	Batch name
duration	Varchar(50)	Duration of course

Table No:8

Table name: Tbl_examduty

Primary key:id

FIELD	TYPE	DESCRIPTION
id	Int(11)	Exam duty id
dateof	Date	Date
class	Int(11)	Class
fid	Varchar(50)	Foreign Key
batchid	Varchar(50)	Foreign Key
examid	Varchar(50)	Foreign Key

Table No:9

Table name: Tbl_exam

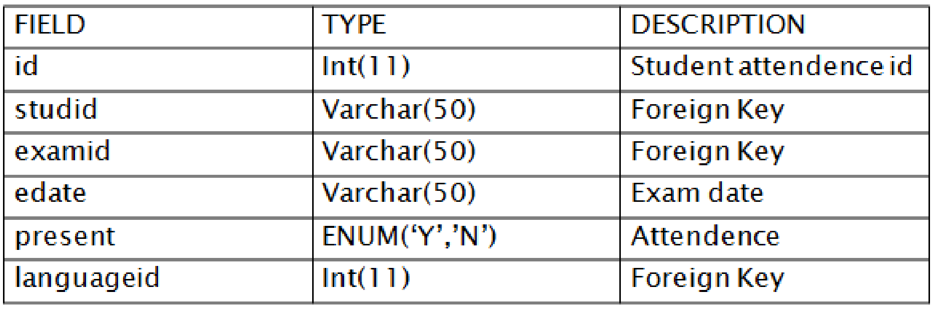
Primary key:id

FIELD	TYPE	DESCRIPTION
id	Int(11)	Exam id
examname	Varchar(50)	Exam name
batchid	Int(11)	Foreign Key
semid	Int(11)	Foreign Key

Table No:10

Table name: Tbl_studentattendence

Primary key:id

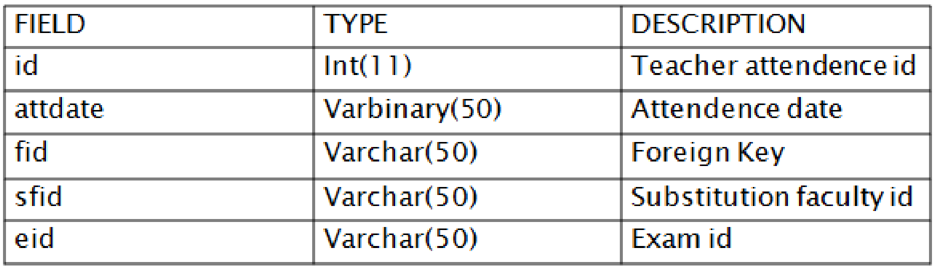


FIELD	TYPE	DESCRIPTION
id	Int(11)	Student attendence id
studid	Varchar(50)	Foreign Key
examid	Varchar(50)	Foreign Key
edate	Varchar(50)	Exam date
present	ENUM('Y','N')	Attendence
languageid	Int(11)	Foreign Key

Table No:11

Table name: Tbl_teacherattendence

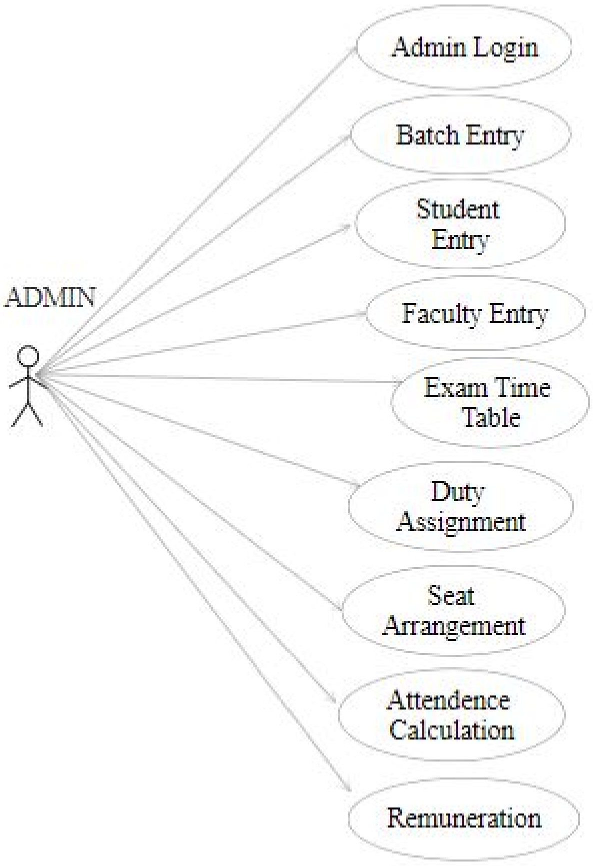
Primary key:id



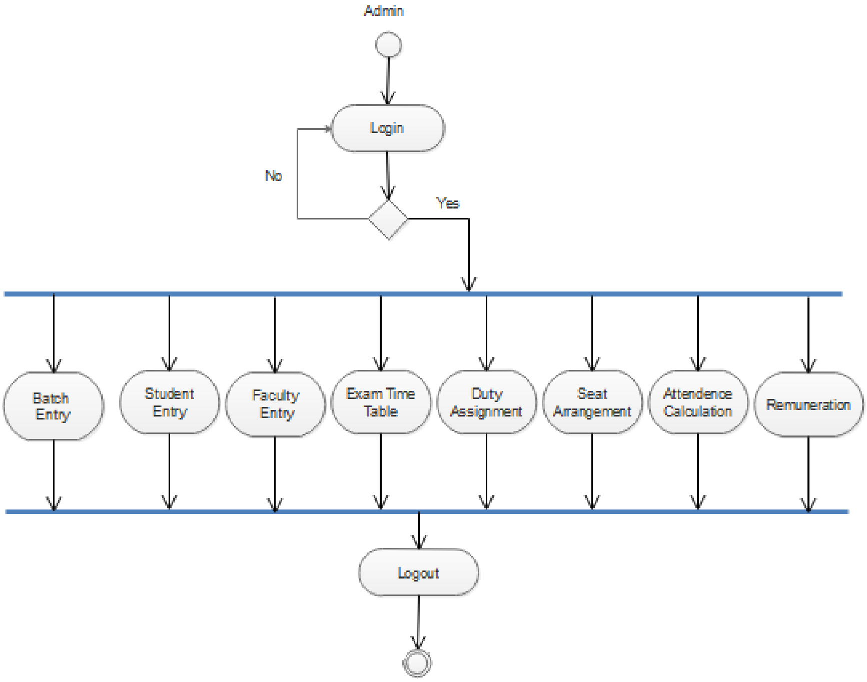
FIELD	TYPE	DESCRIPTION
id	Int(11)	Teacher attendence id
attdate	Varbinary(50)	Attendence date
fid	Varchar(50)	Foreign Key
sfid	Varchar(50)	Substitution faculty id
eid	Varchar(50)	Exam id

6.10 UML

6.10.1 Usecase Diagram



6.10.2 Activity Diagram



7 SYSTEM TESTING

7.1 Introduction to System Testing

Testing is the process of examining the software to compare the actual behavior with that of the excepted behavior. The major goal of software testing is to demonstrate that faults are not present. In order to achieve this goal the tester executes the program with the intent of finding errors. Though testing cannot show absence of errors but by not showing their presence it is considered that these are not present.

System testing is defined as the process by which one detects the defects in the software. Any software development organization or team has to perform several processes. Software testing is one among them. It is the final opportunity of any programmer to detect and rectify any defects that may have appeared during the software development stage. Testing is a process of testing a program with the explicit intention of finding errors that makes the program fail. In short system testing and quality assurance is a review in software products and related documentation for completion, correctness, reliability and maintainability.

System testing is the first stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences. Testing is vital to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct and the goal will be successfully achieved. A series of testing are performed for the proposed system before the proposed system is ready for user acceptance testing.

7.2 Unit Testing

This method of testing test the smallest unit of software called modules. It will test all the important path to find errors within the boundary of module. This has enabled the detection of errors in coding and logic. Various test cases are prepared. For each module these test cases are implemented and it is checked whether the module is executed as per the requirements and outputs the desired result. In this test each service input and output parameters are checked. In unit testing, All independent paths through the control structures are executed to ensure that all statements in the modules have been executed at least once. Error handling paths are also tested.

7.3 Integration Testing

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. In this testing, all individual modules were combined and module wise shifting was verified to be alright The integration testing is performed in the Ilahia Exam Cell Automation by combining the activities in admin module and found all activities are running without any error.

7.4 Validation Testing

Validation testing can be defined in many ways, but a simple definition is that validation succeeds when the software functions in manner that is reasonably accepted by user. Software validation is achieved through a series of tests that demonstrate conformity with requirement. Deviation or error discovered at this step in this project is corrected prior to completion of the project with the help of the user. In Ilahia Exam Cell Automation verifications are done correctly. So there is no chance for users to enter incorrect values. It will give error messages by using different validations. The validation testing is done very clearly and found it is error free.

7.5 Alpha Testing

Alpha testing is one of the most common software testing strategies used in software development. Its specially used by product development organizations.

1. This test takes place at the developers site. Developers observe the users and note problems.

2. Alpha testing is testing of an application when development is about to complete. Minor design changes can still be made as a result of alpha testing.

3. Alpha testing is typically performed by a group that is independent of the design team, but still within the company, e.g. in-house software test engineers, or software QA engineers.

4. Alpha testing is final testing before the software is released to the general public. It has two phases:

5. In the first phase of alpha testing, the software is tested by in-house developers. They use either debugger software, or hardware-assisted debuggers. The goal is to catch bugs quickly.

6. In the second phase of alpha testing, the software is handed over to the software QA staff, for additional testing in an environment that is similar to the intended use.

7. Alpha testing is simulated or actual operational testing by potential users/customers or an independent test team at the developers site. Alpha testing is often employed for off-the-shelf software as a form of internal acceptance testing, before the software goes to beta testing.

7.6 Beta Testing

Beta Testing is also known as field testing. It takes place at customers site. It sends the system/software to users who install it and use it under real-world working conditions.

- A beta test is the second phase of software testing in which a sampling of the intended audience tries the product out. (Beta is the second letter of the Greek alphabet.) Originally, the term alpha testing meant the first phase of testing in a software development process. The first phase includes unit testing, component testing, and system testing. Beta testing can be considered pre-release testing.

- The goal of beta testing is to place your application in the hands of real users outside of your own engineering team to discover any flaws or issues from the users perspective that you would not want to have in your final, released version of the application. Example: Microsoft and many other organizations release beta versions of their products to be tested by users.

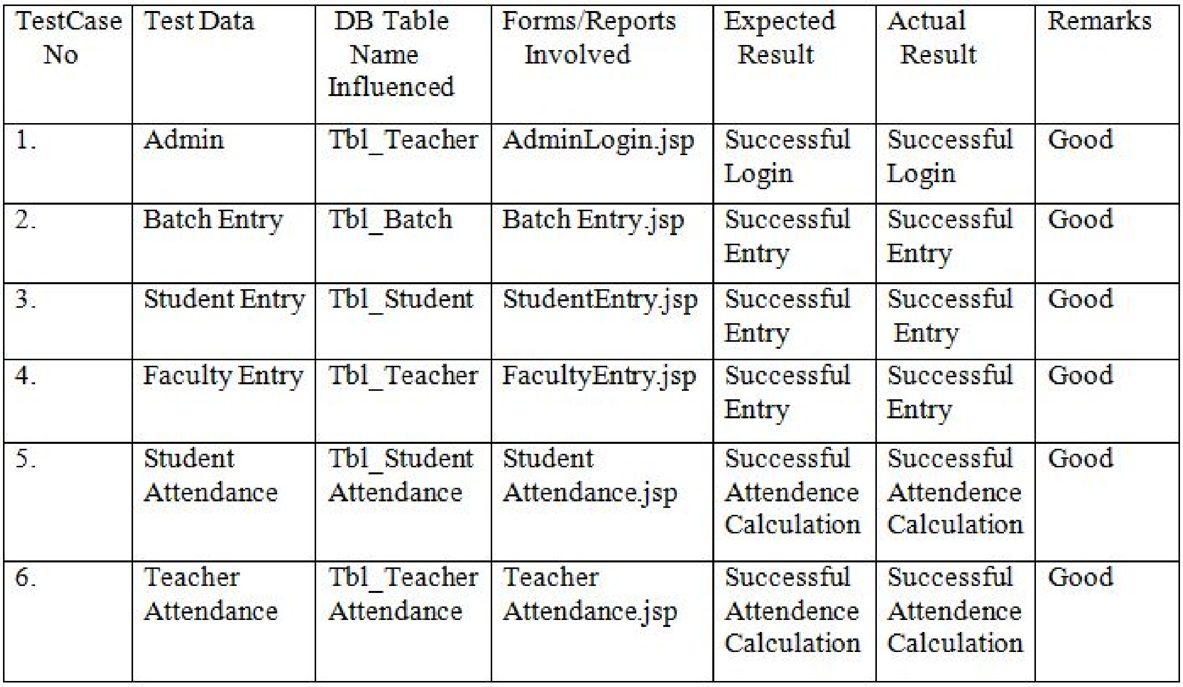
7.7 Bugzilla

Bugzilla is an open-source issue/bug tracking system that allows developers effectively to keep track of outstanding problems with their product. It is written in Perl and uses MYSQL database.Bugzilla is a Defect tracking tool, however it can be used as a test management tool as such it can be easily linked with other Test Case management tools like Quality Center, Testlink etc. This open bug-tracker enables users to stay connected with their clients or employees, to communicate about problems effectively throughout the data-management chain Key features of Bugzilla includes

1. Advanced search capabilities
2. E-mail Notifications
3. Modify/file Bugs by e-mail
4. Time tracking
5. Strong security
6. Customization
7. Localization

7.8 Test Cases

A Test Case is a script, program, or other mechanism that exercises a software component to ascertain that a specific correctness assertion is true. In general, it creates a specified initial state, invokes the tested component in a specified way, observes its behavior, and checks to ensure that the behavior was correct.



TestCase No	Test Data	DB Table Name Influenced	Forms/Reports Involved	Expected Result	Actual Result	Remarks
1.	Admin	Tbl_Teacher	AdminLogin.jsp	Successful Login	Successful Login	Good
2.	Batch Entry	Tbl_Batch	Batch Entry.jsp	Successful Entry	Successful Entry	Good
3.	Student Entry	Tbl_Student	StudentEntry.jsp	Successful Entry	Successful Entry	Good
4.	Faculty Entry	Tbl_Teacher	FacultyEntry.jsp	Successful Entry	Successful Entry	Good
5.	Student Attendance	Tbl_Student Attendance	Student Attendance.jsp	Successful Attendence Calculation	Successful Attendence Calculation	Good
6.	Teacher Attendance	Tbl_Teacher Attendance	Teacher Attendance.jsp	Successful Attendence Calculation	Successful Attendence Calculation	Good

8 SYSTEM IMPLEMENTATION

8.1 Introduction to System Implementation

The implementation is the final state and it is an important phase. It involves the individual programming; system testing, user training and the operational running of developed proposed system that constitutes the application subsystems. A major task of preparing for implementation is education of users, which should really have been taken place much earlier in the project when they were being involved in the investigation and design work. During the implementation phase system actually takes physical shape. In order to develop a system implemented planning is very essential. A software implementation method is a systematically structured approach to effectively integrate software based service or component into the workflow of an organizational structure or an individual end-user. This entry focuses on the process modeling (Process Modeling), a process model is a description of a process at the type level, side of the implementation of large product software, using the implementation of Enterprise Resource Planning systems as the main example to elaborate on. A product software implementation method is a blueprint to get users and/or organizations running with a specific software product. The method is a set of rules and views to cope with the most common issues that occur when implementing a software product: business alignment from the organizational view and acceptance from the human view. The implementation of product software, as the final link in the deployment chain of software production, is in a financial perspective of a major issue. The implementation phase of the software development is concerned with translating design specification into source code. The user tests the developed system and changes are made according to their needs. Our system has been successfully implemented. Before implementation several tests have been conducted to ensure that no errors are encountered during the operation. The implementation phase ends with an evaluation of the system after placing into the operation for a period of time. Implementation is the third phase of the system processe.

8.2 Training

An analysis of user training focuses on two factors:

1. User capabilities

2. Nature of the system to be installed.

Users range from the native to the highly sophisticated. They approach it as concrete learners, learning how to use the system without trying to understand which abstract principles determine which function. The distinction between concrete and formal (student type) learning says about what one can expect from trainees in general.These project also sophisticated the user capabilities and the corresponding nature of the system to be installed.

8.3 Conversion

Conversion refers to changing from one design to another system. The main objective of conversion is to put tested system into operation while holding costs, risks, and personal irritation to minimum. The various tasks involved in conversion are:

1. Creating computer compatible files.
2. Training the operating staffs.
3. Installing terminals and hardware.

The project entitled Ilahia Exam Cell Automation agreed the conversion phases that begin with a review of the project plan, the system test documentation and the implementation plan. And also conversion portion of the implementation plan is finalized and approved. Files are converted.

8.4 Post Implementation Review

Every system requires periodic evaluation after implementation. A post implementation review measures the systems performance against predefined requirements. Unlike system testing, which determines where the system fails so that the necessary adjustments can be made, a post-implementation review determines how well the system continues to meet performances specifications. It is done after design and conversion are complete. It also provides information to determine whether major redesign is necessary.

8.5 System Maintenance

Software maintenance is the modification of a software product after delivery to correct faults, to improve performance or other attributes. This section describes the six software maintenance processes as:

1. The implementation processes contains software preparation and transition activities, such as the conception and creation of the maintenance plan, the preparation for handling problems identified during development, and the follow-up on product configuration management.

2. The problem and modification analysis process, which is executed once the application has become the responsibility of the maintenance group. The maintenance programmer must analyze each request, confirm it (by reproducing the situation) and check its validity, investigate it and propose a solution, document the request and the solution proposal, and, finally, obtain all the required authorizations to apply the modifications.

3. The process considering the implementation of the modification itself.

4. The process acceptance of the modification, by confirming the modified work with the individual who submitted the request in order to make sure the modification provided a solution.

5. The migration process is exceptional, and is not part of daily maintenance tasks. If the software must be ported to another platform without any change in functionality, this process will be used and a maintenance project team is likely to be assigned to this task.

6. Finally, the last maintenance process, also an event which does not occur on a daily basis, is the retirement of a piece of software.

9 SYSTEM EVALUATION

Although system evaluation is an ongoing process throughout the performance testing effort, it offers greater value when conducted early in the test project. The intent of system evaluation is to collect information about the project as a whole, the functions of the system, the expected user activities, the system architecture, and any other details that are helpful in guiding performance testing to achieve the specific needs of the project.

1. Your need to evaluate and select software that meets your business requirements.

2. Your need to evaluate and select a partner that is capable of delivering the most benefit to your business from your software investment, as well as managing the risks inherent in system implementation projects.

3. Your time and ours is valuable; at each step along the way we will each decide whether or not it is beneficial to proceed.

To help you with your selection, this evaluation process is designed to give us both a clear understanding of the systems to be implemented and the corresponding benefits of the partnership.This information provides a foundation for collecting the performance goals and requirements, characterizing the workload, creating performance-testing strategies and plans, and assessing project and system risks. A thorough understanding of the system under test is critical to a successful performance-testing effort. The measurements gathered during later stages are only as accurate as the models that are developed and validated in this stage. The evaluation provides a foundation for determining acceptable performance; specifying performance requirements of the software, system, or component(s); and identifying any risks to the effort before testing even begins. System evaluation providing in these project is needed to evaluate and select the requirements and managing the risk in system implementation on project. Also it is valuable in time so that way it is beneficial in each step.

10 CONCLUSION

The project was successfully completed within the time span allotted .The drawbacks of the existing system as listed before are fully evacuated. All the existing inconsistencies are fully solved as this system is implemented. This reduced the burden of the administration of the system. The module is tested and put together to form the main system. Finally the system is tested with real data and it worked successfully. Thus the system has fulfilled the entire objective defined. The system has been developed in an interactive manner and the reports generated by the system are clear. The system is flexible, user friendly and has its own full data security and all data recovery facility. The developed system only one module,that is admin. And the admin has full control over the system.

10.1 Scope for Future Enhancements

In future, we can expect the modified version of ILAHIA EXAM CELL AUTOMATION system. Ilahia Exam Cell Automation is currently used by MCA department. The scope of the project in the future will be,it can extend its application to other departments like MBA and BTech.

11 APPENDIX

11.1 APPENDIX A

11.1.1 Sample Source Code / Pseudo Code

- LOGIN

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<%@page import="java.sql.ResultSet"%>
<jsp:useBean class="db.dbConnection" id="obj"></jsp:useBean>
<%
    String AdnUser="",AdnPass="";
    AdnUser=request.getParameter("txt_name");
    AdnPass=request.getParameter("txt_pass");
    if(request.getParameter("Btn_save")!=null)
    {
        ResultSet rsAdminLogin=obj.selectData("select * from tbl_teacher th wher
    while(rsAdminLogin.next())
    {
        response.sendRedirect("Homepage.jsp");
    }
    }
 %>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Admin Login</title>
        <style>
            body
               {
                 background:#d4d4d4 url("Images/Screenshot_20171213_203021.png")
                margin:0px; padding:0px;
                font-size:12px; font-family:Arial; color:#363636;
               }
               .MalyaliMarriage_logo
```

```
                {
                 padding:10px 15px;
                 }
                 .Propic
                 {
                     width:600px;
                     height: 300px;
                     border:solid #363636;
                     border-radius: 10px;
                     margin-left: 400px;
                     margin-top: 50px;
                    //background-color: #FFFFFF;
                   background: transparent;
box-shadow: #bbb 0px 8px 8px 0px;
                 }

  .regHead
  {
                       border-style:2px solid;
                       border-color: #363636;
                       border-radius:10px;
                       padding: 5px 40px 5px 40px;
                       background-color:#ff0033;

  }
    .registerhead
                       {
                           font-family:sans-serif ;font-size:24px;color:#ffffff;

                       }

                       .MySaveBtn
                       {
                           margin-left: 150px;
                           margin-top: 20px;
                           border-radius: 10px;
                           color:#FFFFFF;
                           background-color: #ff0033;
                           width: 100px;
                           height: 30px;
                       }
                       .MySaveBtn:hover
                       {
                            margin-left: 140px;
                           margin-top: 10px;
                           border-radius: 10px;
                           color:#FFFFFF;
                           background-color: #ff0033;
                           width: 120px;
                           height: 35px;
                           //border: solid #363636;
                           box-shadow: #363636 5px 5px;
                           font-size: 20px;
                       }
                       .MytxtBox
                       {
                          margin-left: 20px;margin-top: 10px;border: solid #3636
                       }
                       .MytxtBox:hover{
                           box-shadow: #363636 5px 5px;
                       }
                       .Mylabel
                       {
                          margin-left: 40px;
                          margin-top: 10px;
                          color: #ffffff;
                          font-size: 25px;
                          font-family: cursive;
                       }
        </style>
    </head>
    <body>
        <form method="post">
         <div style="width: 600px;height: 120px; background-color: #363636;margi
              <div class="MalyaliMarriage_logo" style="float:left;">


                  <img src="logo/download (12).jpg" alt="Ilahia Exam Cell Automa

    </div>
            <div    style="color:#ff0033; font-size:26px; margin:0px auto;float:
            </div>
              </div>

           <div class="Propic">

               <br><label class="registerhead" style="margin-left: 40px;"><span
               <table style="margin-top: 50px;">
                   <tr>
                       <td>
                           <label class="Mylabel" >UserName </label></td><td>
                           <input class="MytxtBox" type="text" placeholder="User
               </tr>
               <tr>
                   <td>
               <label class="Mylabel">Password </label>
                   </td>
                   <td>               <input  class="MytxtBox" type="password" p
</td>
               </tr>
               </table>
               <button class="MySaveBtn" name="Btn_save" >Login</button>

         </div>
        </form>
    </body>
</html>
```

- BATCH ENTRY

```
 <%@page import="java.sql.ResultSet"%>
<jsp:useBean class="db.dbConnection" id="obb"></jsp:useBean>

<!DOCTYPE html>
<%
    String Batch="",BatchId="",Duration="";
    String s="";
    if(request.getParameter("btn_submit")!=null)
    {
        String Bname=request.getParameter("txt_Batch");
        if(!request.getParameter("hiddenid").equals(""))
        {
            int bid=Integer.parseInt(request.getParameter("hiddenid"));
            obb.executeCommand("update tbl_batch set BatchName='"+Bname+"'and Du
        s="update tbl_batch set BatchName='"+Bname+"'and Duration='"+request.get
        }
        else
        {
        obb.executeCommand("insert into tbl_batch (BatchName,Duration)  values('
                }

    }
    if(request.getParameter("del")!=null)
    {
      int bid=Integer.parseInt(request.getParameter("del"));
      obb.executeCommand("delete from tbl_batch where id='"+bid+"'");
    }

    if(request.getParameter("edit")!=null)
    {
       String bid1=request.getParameter("edit");
       ResultSet rs1=obb.selectData("select * from tbl_batch where id='"+bid1+"'
       while(rs1.next())
       {
       Batch=rs1.getString("BatchName");
       BatchId=rs1.getString("id");
       Duration=rs1.getString("Duration");

    }
    }
   %>
<html>
    <head>

        <link href="AdminCss/AdminCss.css" rel="stylesheet" type="text/css" />
        <script src="Js/AdminCommonSearchJs.js" type="text/javascript"></script>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Body Type page</title>
    </head>
    <body>
```

```
        <div style="width: 1400px;height: 120px; background-color: #363636;margi
              <div class="Happywedding_logo" style="float:left;">


<img src="logo/download (12).jpg" alt="Ilahia Exam Cell" title="Ilahia Exam Cell

    </div>
                                        <div style="color:#ff0033; font-size:26p

            </div>

                                   </div>



        <form class="Malayali_campaignrightthree" style="margin-top: 150px">

            <input type="hidden" name="hiddenid" value="<%=BatchId%>">




            <div class="clear"></div>
            <div class="ReligionForm_details">
                <div class="registerhead"><label style="color: #ff0033">Batch En
                <table>
                    <tr>
                        <td><div class="Religion_innerregisterform_new" ><label
                    </tr>
                     <tr>
                        <td><div class="Religion_innerregisterform_new" ><label
                    </tr>
                    <tr><td></td><td><div class="Religion_innerregisterform_new"
                                  <input type="submit" value="Cancel" name="btn_
                </tr>
                </table>
                <!--table searching start here-->




                 <div class="Religion_innerregisterform_table">
                 <input type="text" id="myInput" onkeyup="myFunction()" placehol
```

```
     <table id="myTable" >

             <tr class="header" >
              <th style="width:60%;">Body Type</th>
               <th style="width:40%;">Duration</th>
                <th style="width:40%;">Edit</th>
               <th style="width:40%;">Delete</th>
             </tr>
             <% ResultSet rs=obb.selectData("select * from tbl_b
             while(rs.next())
             {



             %>
             <tr>
                <td>
                    <%=rs.getString("BatchName") %>
                </td>
                <td>
                    <%=rs.getString("Duration") %>
                </td>

                <td>
                    <a href="BatchEntry.jsp?edit=<%=rs.getString
                    </a>
                </td>
                <td>
                    <a href="BatchEntry.jsp?del=<%=rs.getString(
                </td>

            </tr>

            <%
             }

            %>

        </table>
</div>
```

```
                                        <!--table searching end here-->

            </div>
                            <%

                             out.print(s);
                             %>

          </form>


      </body>

  </html>
```

- FACULTY ENTRY

```
<%@page import="java.util.Random"%>
<%@page import="java.sql.ResultSet"%>
<jsp:useBean class="db.dbConnection" id="obb"></jsp:useBean>


<!DOCTYPE html>
 <%
      String FacultyName="",FacultyId="";

                                        if(request.getParameter("btn_submit")!=nul
                                        {
                                            if(!request.getParameter("hiddenid").
        {
            int bid=Integer.parseInt(request.getParameter("hiddenid"));
            obb.executeCommand("update tbl_teacher set Name='"+request.getParame
        }
                                            else
                                            {
                                            Random rand = new Random();
                                          int n = rand.nextInt(90000) + 10000;
                                          String FId="F"+n;
                                          obb.executeCommand("insert into tbl_tea
                                           //s="insert into tbl_teacher (FacultyI
                                      }
                                      }

                                        if(request.getParameter("del")!=null)
    {
      int bid=Integer.parseInt(request.getParameter("del"));
      obb.executeCommand("delete from tbl_teacher where id='"+bid+"'");
    }

    if(request.getParameter("edit")!=null)
    {
       String bid1=request.getParameter("edit");
       ResultSet rs1=obb.selectData("select * from tbl_teacher where id='"+bid1+
       while(rs1.next())
       {
       FacultyName=rs1.getString("Name");
       FacultyId=rs1.getString("id");

    }
    }
                                      %>
<html>
    <head>

        <link href="AdminCss/AdminCss.css" rel="stylesheet" type="text/css" />
        <script src="Js/AdminCommonSearchJs.js" type="text/javascript"></script>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Teacher Entry</title>
        <style>
             body
               {
                 background:#d4d4d4 url("Images/Screenshot_20180219_221915.png")
                margin:0px; padding:0px;
                font-size:12px; font-family:Arial; color:#363636;
               }
        </style>
    </head>
    <body>

        <div style="width: 1400px;height: 120px; background-color: #363636;margi
              <div class="Happywedding_logo" style="float:left;">
```

```
<img src="logo/download (12).jpg" alt="Ilahia Exam Cell" title="Ilahia Exam Cell

    </div>
                                    <div style="color:#ff0033; font-size:26p

            </div>

                                 </div>

        <form class="Malayali_campaignrightthree" style="margin-top: 150px">
            <input type="hidden" name="hiddenid" value="<%=FacultyId%>">




            <div class="clear"></div>
            <div class="ReligionForm_details">
                <div class="registerhead"><label style="color: #ff0033">Faculty 
                <table>
                    <tr>
                        <td><div class="Religion_innerregisterform_new" ><label
                    </tr>
                    <tr><td></td><td><div class="Religion_innerregisterform_new"
                                <input type="submit" value="Cancel" name="btn_

                </tr>
                </table>
                <!--table searching start here-->





                 <div class="Religion_innerregisterform_table">
                 <input type="text" id="myInput" onkeyup="myFunction()" placehol
                          <table id="myTable" >

                             <tr class="header" >
                              <th style="width:60%;">Teacher Name</th>
                                <th style="width:40%;">Edit</th>
```

```
                           <th style="width:40%;">Delete</th>
                         </tr>
                         <% ResultSet rs=obb.selectData("select * from tbl_t
                         while(rs.next())
                         {

                         %>
                         <tr>
                            <td>
                                <%=rs.getString("Name") %>
                            </td>

                            <td>
                                <a href="FacultyEntry.jsp?edit=<%=rs.getStri
                                </a>
                            </td>
                            <td>
                                <a href="FacultyEntry.jsp?del=<%=rs.getStrin
                            </td>

                        </tr>

                        <%
                         }



                         %>

                    </table>

            </div>
                                    <!--table searching end here-->

        </div>


    </form>


</body>
```

```
</html>
```

- TIME TABLE ENTRY

```
<%@page import="java.sql.ResultSet"%>
<%@ page  language="java" import="java.util.*,java.text.*"%>
<%@page import="java.util.Random"%>
<jsp:useBean class="db.dbConnection" id="obj"></jsp:useBean>

<%!
public int nullIntconv(String inv)
{
    int conv=0;

    try{
        conv=Integer.parseInt(inv);
    }
    catch(Exception e)
    {}
    return conv;
}
%>
<%
 int iYear=nullIntconv(request.getParameter("iYear"));
 int iMonth=nullIntconv(request.getParameter("iMonth"));

 Calendar ca = new GregorianCalendar();
 int iTDay=ca.get(Calendar.DATE);
 int iTYear=ca.get(Calendar.YEAR);
 int iTMonth=ca.get(Calendar.MONTH);

 if(iYear==0)
 {
      iYear=iTYear;
      iMonth=iTMonth;
 }

 GregorianCalendar cal = new GregorianCalendar (iYear, iMonth, 1);

 int days=cal.getActualMaximum(Calendar.DAY_OF_MONTH);
 int weekStartDay=cal.get(Calendar.DAY_OF_WEEK);

 cal = new GregorianCalendar (iYear, iMonth, days);
 int iTotalweeks=cal.get(Calendar.WEEK_OF_MONTH);

%>

 <%
        if(request.getParameter("btn_print")!=null)
        {
            response.sendRedirect("PrintTimeTable.jsp");
        }
        %>
<html>
    <style>
body
{
 font-family:cursive;
}
.dsb
{
 background-color:#EEEEEE
}
ul {
  margin: 0;
  padding: 0;
  list-style: none;
  widows: 100%;
}
ul li {
  cursor: pointer;
  position: relative;
  padding: 12px 8px 12px 40px;
  background: #eee;
  font-size: 18px;
  transition: 0.2s;


}

/* Set all odd list items to a different color (zebra-stripes) */
```

```
ul li:nth-child(odd) {
  background: #f9f9f9;
}

/* Darker background-color on hover */
ul li:hover {
  background: #ddd;
}

/* When clicked on, add a background color and strike out text */
ul li.checked {
  background: #888;
  color: #fff;
  text-decoration: line-through;
}

/* Add a "checked" mark when clicked on */

/* Style the close button */


/* Style the header */
.header {
  background-color: #f44336;
  padding: 30px 40px;
  color: white;
  text-align: center;

}
/* Clear floats after the header */
.header:after {
  content: "";
  display: table;
  clear: both;
}

/* Style the input */
input {
  border: none;
  width: 75%;
  padding: 10px;
  float: left;
  font-size: 16px;
}

/* Style the "Add" button */
.addBtn {
  padding: 10px;
  width: 25%;
  background: #d9d9d9;
  color: #555;
  float: left;
  text-align: center;
  font-size: 16px;
  cursor: pointer;
  transition: 0.3s;
}

.addBtn:hover {
  background-color: #bbb;
}
.hiddenmodal
{
 width: 100%;
  display: none;
}

 #myTable {
  border-collapse: collapse;
  width: 100%;
  border: 1px solid #ddd;
  font-size: 14px;

}

#myTable th, #myTable td {
  text-align: left;
  padding: 12px;
}

#myTable tr {
  border-bottom: 1px solid #ddd;
```

```
}

#myTable tr.header, #myTable td:hover {
    background-color: #e0e0e0;
    color: #000;
}

</style>
</head>
<title>Time Table Entry</title>
<script>
function goTo()
{
  document.frm.submit()
}
</script>
<style>
body
{
 font-family:Verdana, Arial, Helvetica, sans-serif
}
.dsb
{
 background-color:#EEEEEE
}
</style>
</head>

<body>
<form name="frm" method="post" >
     <div style="width: 100%;height: 120px; background-color: #363636;margin-bot
              <div class="MalyaliMarriage_logo" style="float:left;">


                  <img src="logo/download (12).jpg" alt="Ilahia Exam Cell Automa

    </div>
            <div    style="color:#ff0033; font-size:26px; margin:0px auto;float:

            </div>
              </div>
```

```
   <div id="divcalender">
   <table width="100%" border="0" cellspacing="0" cellpadding="0" >
 <tr>
   <td width="25%">&nbsp;</td>
   <td width="45%">&nbsp;</td>
   <td width="30%">&nbsp;</td>
 </tr>
 <tr>
   <td>&nbsp;</td>
   <td><table width="100%" border="0" cellspacing="0" cellpadding="0">
       </tr>
 <tr>
   <td>

       <table width="100%" border="0" cellspacing="0" cellpadding="0">
     <tr>
       <td width="6%">Year</td>
       <td width="7%">
       <select name="iYear" onChange="goTo()" id=’yearOfCalender’>
       <%
 int curretyear=0;
int currentmonth=0;
       // start year and end year in combo box to change year in calendar
       for(int iy=iTYear-70;iy<=iTYear+70;iy++)
       {
         if(iy==iYear)
         {
             curretyear=iy;
           %>
         <option value="<%=iy%>" selected="selected"><%=iy%></option>
         <%
         }
         else
         {
           %>
         <option value="<%=iy%>"><%=iy%></option>
         <%
         }
```

```
      }
     %>
      </select></td>
      <td width="73%" align="center" style="color:#4E96DE"><h3><%=new SimpleDa
      <td width="6%">Month</td>
      <td width="8%">
          <select name="iMonth" onChange="goTo()" id="MonthOfCalender">
      <%
      // print month in combo box to change month in calendar
      for(int im=0;im<=11;im++)
      {
        if(im==iMonth)
        {
            currentmonth=im;
       %>
        <option value="<%=im%>" selected="selected"><%=new SimpleDateFormat("M
        <%
        }
        else
        {
          %>
        <option value="<%=im%>"><%=new SimpleDateFormat("MMMM").format(new Dat
        <%
        }
      }
     %>
      </select></td>
    </tr>
  </table></td>
</tr>
<tr>
  <td><table align="center" border="1" cellpadding="3" cellspacing="0" width="
    <tbody>
      <tr>
        <th>Sun</th>
        <th>Mon</th>
        <th>Tue</th>
        <th>Wed</th>
        <th>Thu</th>
        <th>Fri</th>
        <th>Sat</th>
```

```
</tr>
<%
int cnt =1;
for(int i=1;i<=iTotalweeks;i++)
{
%>
<tr>
  <%
    for(int j=1;j<=7;j++)
    {
        if(cnt<weekStartDay || (cnt-weekStartDay+1)>days)
        {
         %>
        <td align="center" height="35" class="dsb">&nbsp;</td>

        <%
        }
        else
        {
             int day=cnt-weekStartDay+1;
         %>
         <td align="center" height="35" id="day_<%=(cnt-weekStartDay+1)%
              <%    ResultSet rsa=obj.selectData("select * from tbl_exam
                         while(rsa.next())
                         {
                             %>
             <div style="border: 1px solid #555;border-radius: 10px; pad
                 <label style="color: #f44336; margin-top: 5px;" >
                     <span><%=rsa.getString("subjectname")%>:<%=rsa.getS
             <%=rsa.getString("timing")%>
                 </label></div><br>
             <%
                         }
                         %>

       <%
        }
        cnt++;
      }
    %>
         </td>
```

```
          </tr>
          <%
          }
          %>
        </tbody>
      </table></td>
    </tr>
  </table></td>
      <td>&nbsp;</td>
    </tr>
    <tr>
      <td>&nbsp;</td>
      <td>&nbsp;</td>
      <td>&nbsp;</td>
    </tr>
  </table>

        <div class="Happywedding_registerform_details" style="margin-top: 0px;marg
           <table >


                 <tr><td><div class="bengali_innerregisterform_new" id="lname"><la


                                </td><td>

                                    <div class="bengali_innerregisterform_new">
                                        <select class="regtxtbox" onchange="AjaxSem

                                            <option>--Select--</option>
                                            <%

   ResultSet rs= obj.selectData("select * from tbl_exam ");
   while(rs.next())
   {
       %>
       <option value="<%=rs.getString("id")%>"><%=rs.getString("ExamName")%></opti
       <%
   }
   %>
```

```
                                     </select>
                                 </div>

                             </td>
                         </tr>

                         <tr><td><div class="bengali_innerregisterform_new" id="


                             </td><td>

                                 <div class="bengali_innerregisterform_new">
                                     <select class="regtxtbox" id="language">




                                     </select>
                                 </div>

                             </td>
                         </tr>

                         <tr>
                             <td>
                                 <div class="bengali_innerregisterform_new">Timi
                         </tr>
                         <tr><td><div class="bengali_innerregisterform_new"></di
                         </tr>

                         <!--religion -->

                         <!--cast/division-->

         </table>
        </div>
                                     <input type="submit" value="Print Time Tabl
    </div>

</form>
```

```
</body>
<script src="MyJavaScript/jquery.min.js" type="text/javascript"></script>
<script type="text/javascript">
    var day="";
    var year="";
    var month="";

    function showAlert(d)
    {
       day=d;
       alert(day);
        year=document.getElementById("yearOfCalender").value;
         alert(year);
         month=(document.getElementById("MonthOfCalender").value);

         alert(month);

         hiddenDiv.style.display="block";
    }
    function addTime()
    {
       var l=language.value;
       alert(l);
       var t=timing.value;
       alert(t);
        var ExamId=examSel.value;
        alert(ExamId);

        $.ajax({url:"ajaxDate.jsp?d="+day+"&m="+month+"&y="+year+"&l="+l+"&t="+t
         success: function(result){
         alert(result);
      $("#"+day).html(result);
    }});
    }
     function visibleTime(d)
     {
       divcalender.style.display="block" ;
       alert(d);

     }
    function semdiv()
    {
        semdivition.style.display="block";
    }
    function AjaxSem(d)
    {
        alert(d);
         $.ajax({url:"ajaxSem.jsp?d="+d,
         success: function(result){
         alert(result);
      $("#language").html(result);
    }});
    }
</script>
</html>
```

- DUTY LIST

```
<%@page import="java.sql.ResultSet"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<jsp:useBean class="db.dbConnection" id="obj"></jsp:useBean>
<%@include file="header.jsp" %>


<html>
    <head>

        <title>Time Table</title>
          <style>
            #myTable {
  border-collapse: collapse;
  width: 80%;
  border: 1px solid #ddd;
  font-size: 14px;

}

#myTable th, #myTable td {
  text-align: left;
  padding: 12px;
```

```
}

#myTable tr {
  border-bottom: 1px solid #ddd;
}
        </style>
    </head>
    <body>
        <form>
        <h1>Exam Duty List</h1>
        <table id="hidetable">
            <tr>   <td>Exam Name :</td><td><select onchange="chEck(this.value)"
                        <option value="0">--select--</option>
                <%
                    ResultSet rs=obj.selectData("select * from tbl_examtimetable
                    while(rs.next()){
                        %>
                        <option value="<%=rs.getString("Examid")%>"><%=rs.getStr

                        <%
                    }
                    %>

                    </select></td><tr>
            <tr>

        </table>



                    <div id="TimeTableDiv">
                    </div>
        </form>
                     <script src="MyJavaScript/jquery.min.js" type="text/javascr
    <script>


   function chEck(d)
   {
       alert(d);
        $.ajax({url:"AjaxDutyList.jsp?Eid="+d,
         success: function(result){
```

```
         alert(result);
      $("#TimeTableDiv").html(result);
    }});
   }
    function MyPrint()
  {
      hidetable.style.display="none";
      prntbtn.style.display="none";
      window.print();
  }

    </script>

    </body>

</html>
```

- SEAT ARRANGEMENT

```
<%@page import="java.sql.ResultSet"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<jsp:useBean class="db.dbConnection" id="obj"></jsp:useBean>
<%@include file="header.jsp" %>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Ilahia Exam Cell Automation</title>
        <style>
            #myTable {
  border-collapse: collapse;
  width: 80%;
  border: 1px solid #ddd;
  font-size: 14px;

}

#myTable th, #myTable td {
  text-align: left;
  padding: 12px;
}
```

```
#myTable tr {
 border-bottom: 1px solid #ddd;
}
    </style>
  </head>
  <body><form >
      <div id="TohidePrintDiv" style="display: block">
        <label>Exam :</label><select id="batchExam" onchange="Attendence
    <option value="0">--select--</option>
    <%
    ResultSet rsbatch=obj.selectData("select * from tbl_exam");
    while(rsbatch.next())
    {
      %>
      <option value="<%=rsbatch.getString("id")%>"><%=rsbatch.getString("E
      <%
    }
    %>
  </select>
  <br>






  </div>

          <div id="ExamDutyGenerationDiv">


          </div>
    </form>
  </body>
  <script src="MyJavaScript/jquery.min.js" type="text/javascript"></script>
  <script>
    function semdiv()
  {
    semdivition.style.display="block";
  }
  function Attendence(d)
  {


    alert(d);
     $.ajax({url:"Ajax_ClassList.jsp?EID="+d,
     success: function(result){
     alert(result);
  $("#ExamDutyGenerationDiv").html(result);
  }});

  }
  function MyPrint()
  {
    TohidePrintDiv.style.display="none";
    prntbtn.style.display="none";
window.print();
// myTable.print();
  }
  function getSeat()
  {
   var e=batchExam.value;
  var date=dateExm.value;
  var noclass=noclassof.value;
  var k=ktuiddd.value;
  alert(e);
  alert(date);
   alert(noclass);
alert(k);
     $.ajax({url:"AjaxSeat.jsp?EID="+e+"&KtuId="+k+"&date="+date+"&noOfclass
     success: function(result){
     alert(result);
$("#ExamDutyGenerationDiv").html(result);
  }});
  }

  function class1()
  {
  var divtopr=document.getElementById("class1");
  newWin=window.open("");
  newWin.document.write(divtopr.outerHTML);
```

```
      newWin.print();
      newWin.close();
    }
    function class2()
    {
        var divtopr=document.getElementById("class2");
        newWin=window.open("");
      newWin.document.write(divtopr.outerHTML);

      newWin.print();
      newWin.close();
    }
    function class3()
    {
        var divtopr=document.getElementById("class3");
         newWin=window.open("");
      newWin.document.write(divtopr.outerHTML);

      newWin.print();
      newWin.close();

    }
    function class4()
    {
         var divtopr=document.getElementsByName("class4");
         newWin=window.open("");
      newWin.document.write(divtopr.outerHTML);

      newWin.print();
      newWin.close();

    }
</script>
</html>
```

- REMUNERATION

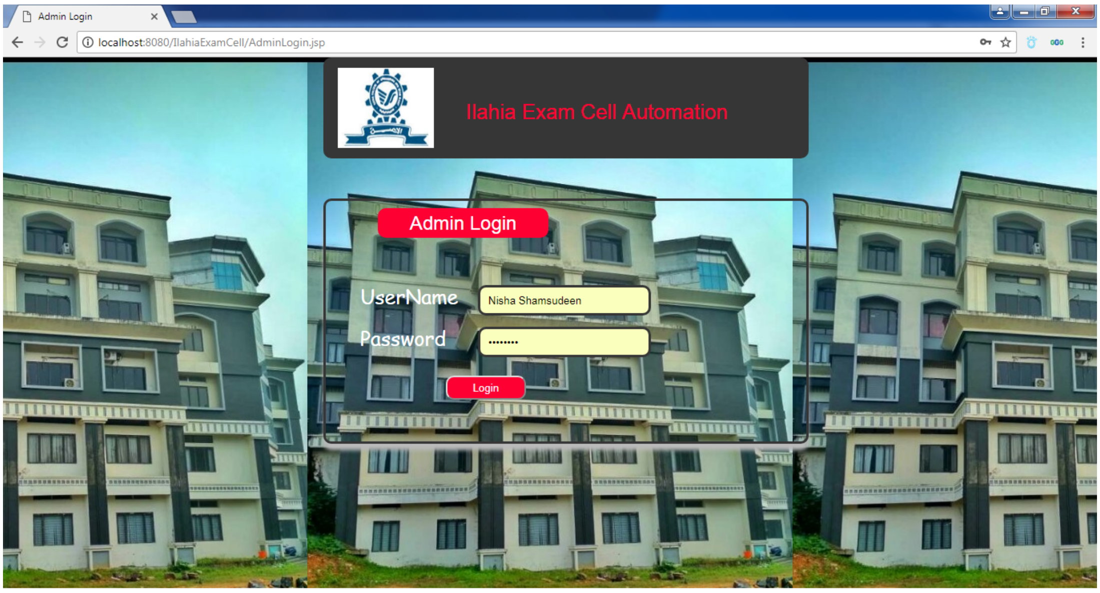
```
<%@page import="java.sql.ResultSet"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<jsp:useBean class="db.dbConnection" id="obj"></jsp:useBean>
```

```
<%@include file="header.jsp" %>

<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Remuneration</title>
         <style>
            #myTable {
  border-collapse: collapse;
  width: 80%;
  border: 1px solid #ddd;
  font-size: 14px;

}

#myTable th, #myTable td {
  text-align: left;
  padding: 12px;
}

#myTable tr {
  border-bottom: 1px solid #ddd;
}
         </style>

    </head>
    <body>
         <h1>Remuneration</h1>
        <table id="hidetable">
             <tr><td>Remuneration for One Duty :</td><td><input type="text" id="
            <tr>   <td>Exam Name:</td><td><select id="ExamCell" onchange="getRem
                <%
                    ResultSet rs=obj.selectData("select * from tbl_examtimetable
                    while(rs.next()){
                        %>
                        <option value="<%=rs.getString("Examid")%>"><%=rs.getStr

                        <%
                    }
                    %>
                    </select></td><tr>


        </table>
                    <div id="remunerationDiv">

                    </div>
    </body>
     <script src="MyJavaScript/jquery.min.js" type="text/javascript"></script>
    <script>


    function getRemuneration(d)
    {

     var r=RemuoneDay.value;
        alert(d);

         $.ajax({url:"AjaxRemuneration.jsp?EID="+d+"&r="+r,
         success: function(result){
         alert(result);
      $("#remunerationDiv").html(result);
    }});
    }
  function MyPrint()
  {
      hidetable.style.display="none";
      prntbtn.style.display="none";
      window.print();
  }
  function saveAttendence(val)
  {
      alert(val);
  }

    </script>
</html>
```

11.2 APPENDIX B

11.2.1 Screenshorts

- 11.2.1.1 login

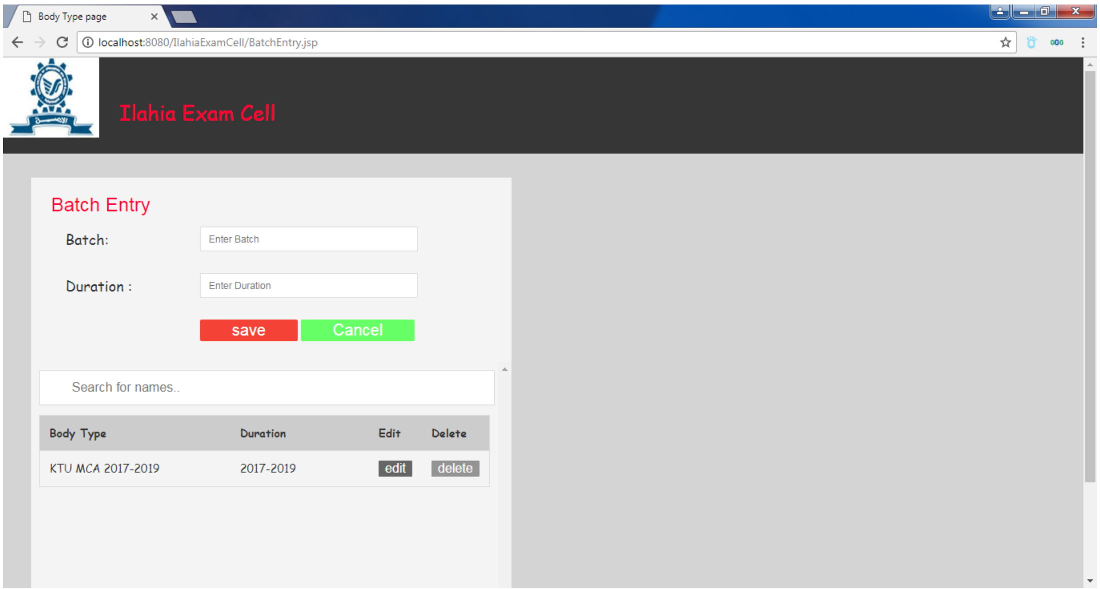


Admin Login
localhost:8080/IlahiaExamCell/AdminLogin.jsp
Ilahia Exam Cell Automation
Admin Login
Login

UserName: Nisha Shamsudeen

Password: ••••••••

- 11.2.1. 2 Batch entry



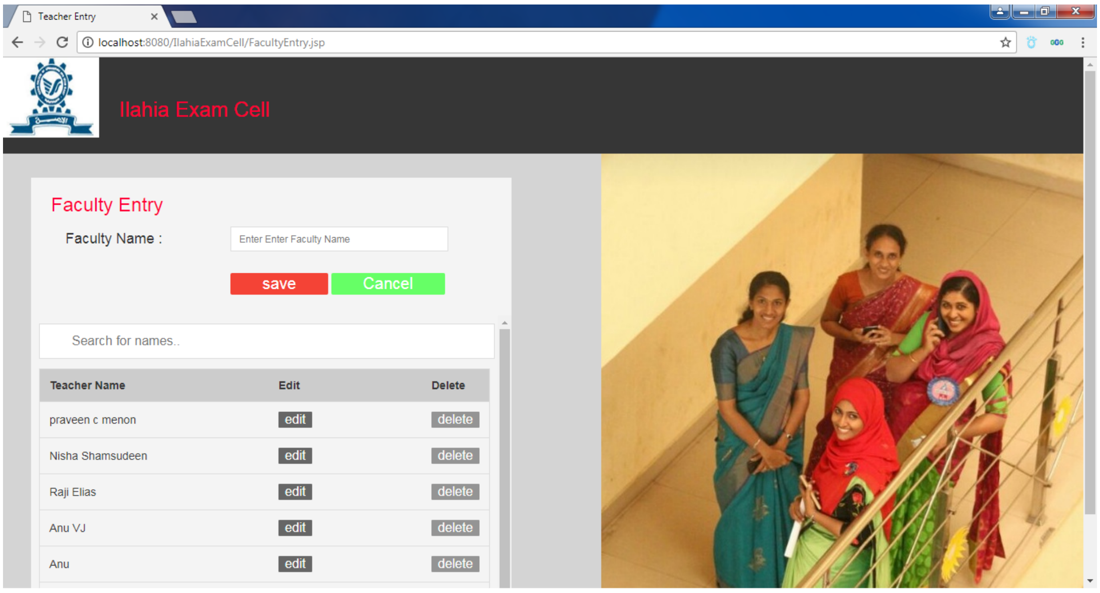
Batch Entry

Batch: Enter Batch

Duration : Enter Duration

Body Type	Duration	Edit	Delete
KTU MCA 2017-2019	2017-2019	edit	delete

- 11.2.1.3 Faculty Entry

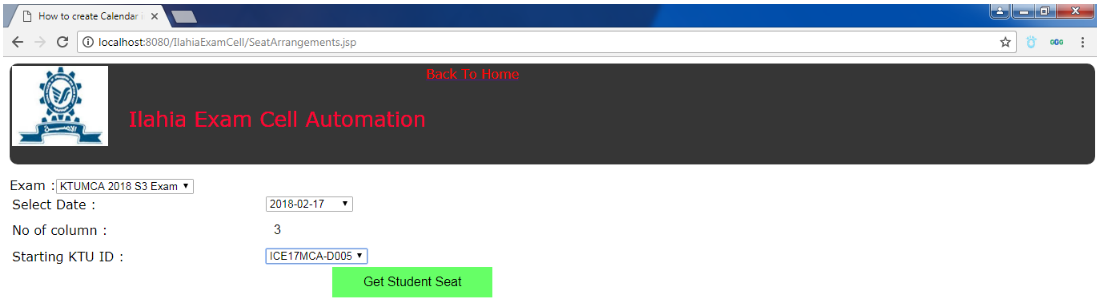


Faculty Entry

Faculty Name : Enter Enter Faculty Name

Teacher Name	Edit	Delete
praveen c menon	edit	delete
Nisha Shamsudeen	edit	delete
Raji Elias	edit	delete
Anu VJ	edit	delete
Anu	edit	delete

- 11.2.1.4 Seat Arrangement


How to create Calendar
localhost:8080/IlahiaExamCell/SeatArrangements.jsp
Back To Home
Ilahia Exam Cell Automation

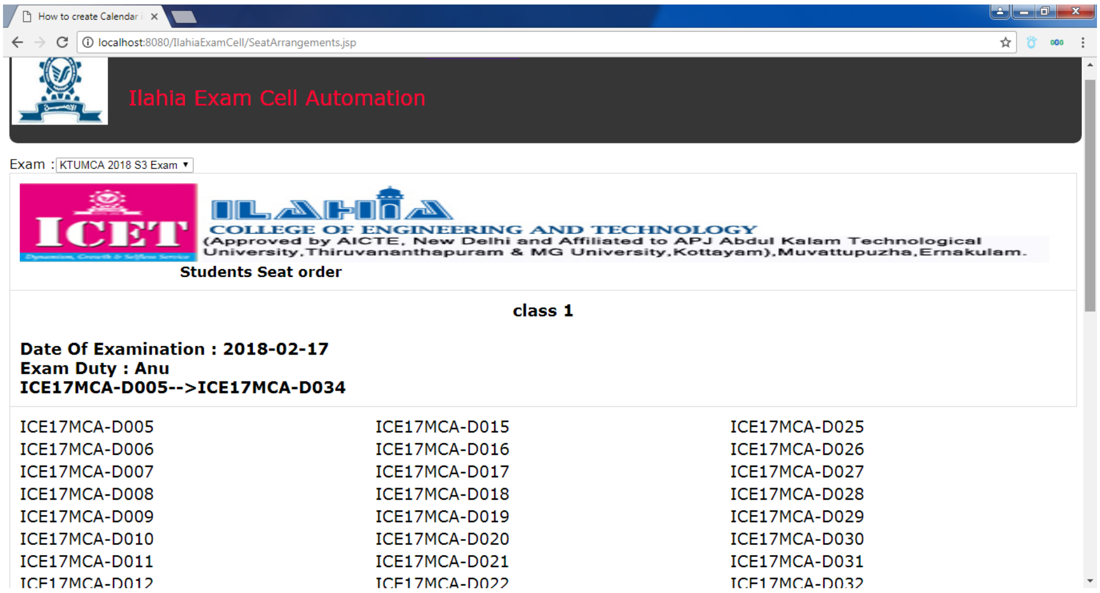
Exam : KTUMCA 2018 S3 Exam ▾

Select Date : 2018-02-17 ▾

No of column : 3

Starting KTU ID : ICE17MCA-D005 ▾

- 11.2.1.5 Seat Arrangement View



Ilahia Exam Cell Automation

Exam : KTUMCA 2018 S3 Exam ▾

ICET ILAHIA COLLEGE OF ENGINEERING AND TECHNOLOGY
(Approved by AICTE, New Delhi and Affiliated to APJ Abdul Kalam Technological University,Thiruvananthapuram & MG University,Kottayam),Muvattupuzha,Ernakulam.

Students Seat order

class 1

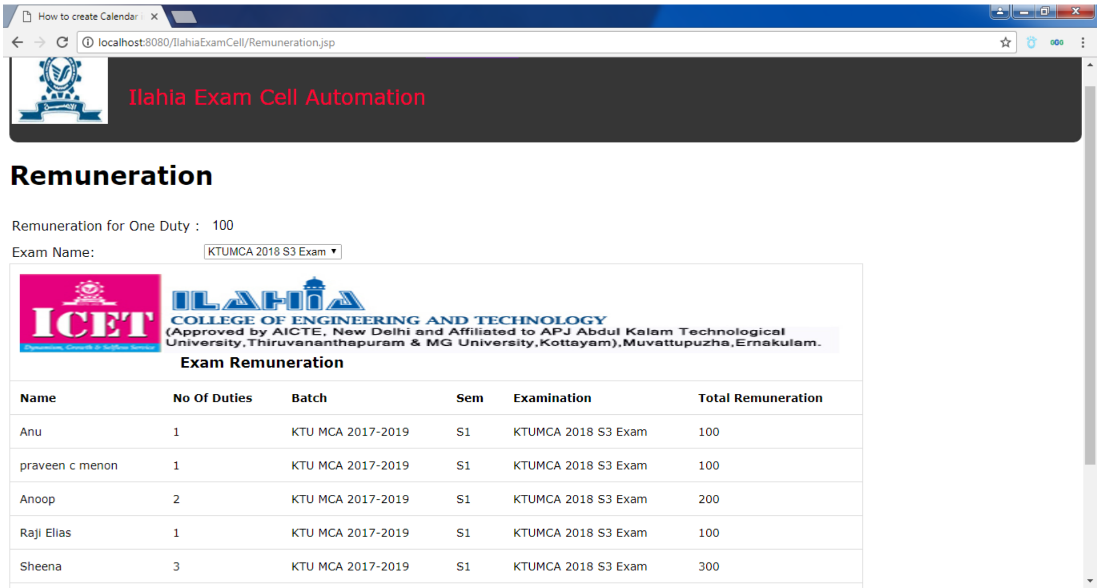
Date Of Examination : 2018-02-17
Exam Duty : Anu
ICE17MCA-D005-->ICE17MCA-D034

ICE17MCA-D005
ICE17MCA-D006
ICE17MCA-D007
ICE17MCA-D008
ICE17MCA-D009
ICE17MCA-D010
ICE17MCA-D011
ICE17MCA-D012

ICE17MCA-D015
ICE17MCA-D016
ICE17MCA-D017
ICE17MCA-D018
ICE17MCA-D019
ICE17MCA-D020
ICE17MCA-D021
ICE17MCA-D022

ICE17MCA-D025
ICE17MCA-D026
ICE17MCA-D027
ICE17MCA-D028
ICE17MCA-D029
ICE17MCA-D030
ICE17MCA-D031
ICE17MCA-D032

- 11.2.1.6 Remuneration



Ilahia Exam Cell Automation

Remuneration

Remuneration for One Duty : 100

Exam Name: KTUMCA 2018 S3 Exam

ICET ILAHIA COLLEGE OF ENGINEERING AND TECHNOLOGY (Approved by AICTE, New Delhi and Affiliated to APJ Abdul Kalam Technological University,Thiruvananthapuram & MG University,Kottayam),Muvattupuzha,Ernakulam.

Exam Remuneration

Name	No Of Duties	Batch	Sem	Examination	Total Remuneration
Anu	1	KTU MCA 2017-2019	S1	KTUMCA 2018 S3 Exam	100
praveen c menon	1	KTU MCA 2017-2019	S1	KTUMCA 2018 S3 Exam	100
Anoop	2	KTU MCA 2017-2019	S1	KTUMCA 2018 S3 Exam	200
Raji Elias	1	KTU MCA 2017-2019	S1	KTUMCA 2018 S3 Exam	100
Sheena	3	KTU MCA 2017-2019	S1	KTUMCA 2018 S3 Exam	300

11.3 APPENDIX C

11.3.1 Acronyms

UML - Unified Modeling Language
HTML - Hyper Text Markup Language
XML - Extensible Markup Language
JSP - Java Server Page
JDBC - Java DataBase Connectivity
JVM - Java Virtual Machine
JDK - Java Developing Kit
CSS - Cascading Style Sheet
JRE - Java Runtime Environment
GB - Giga Bytes
MB - Mega Bytes
QA - Quality Assurance

11.3.2 Bibliography

References:

1-Pure JSP by James Goodwill
2-Softwareb Engineering 2005-by Roger Pressman
3-Android A Beginners Guide-by Vikram Vaswani

Websites:

1. http://www.google.co.in
2. http://www.wikipedia.org
3. http://www.seminarsonly.com
4. http://www.1000projects.org