

PROJECT REPORT
(6 WEEKS PROJECT TRAINING)

TIMETABLE MANAGEMENT SYSTEM

SUBMITTED BY:
MONISH GUPTA
100803060

UNDER THE GUIDANCE OF
MR. ANIL KUMAR VERMA

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
THAPAR UNIVERSITY, PATIALA
(DEEMED UNIVERSITY)

JUN-JUL 2010

DECLARATION

I hereby declare that the project work entitled “TIMETABLE MANAGEMENT SYSTEM” is an authentic record of my own work carried out at Thapar University, Patiala as requirements of six weeks project term for the award of degree of B.E. (Computer Science & Engineering), Thapar University, Patiala, under the guidance of Mr. Anil Kumar Verma, during 31 May to 9 July, 2010.

Monish Gupta

10803060

Date: 9 November, 2010

ACKNOWLEDGEMENT

I would thank and congratulate **Mr. Anil Kumar Verma** and **Mr. Ajay Kumar** for the success of this training program under whose supervision, things went so smoothly. Along with managing this training program, they taught us Object Oriented Concepts and its implementation in Java. I owe thanks to **Mr. Guleria** who showed me how actually the timetable scheduling is done in our University. I owe thanks to **Mr. Varinder Pal Singh** who became our right hand man by finding bugs in the coding part that was required to accomplish the project. I am also thankful to **Mr. Parteek Bhatia** who gave the knowledge about DBMS, its implementation in Oracle, and to **Mr. Maninder Singh** who taught us JDBC (Java Database Connectivity) with Oracle. I would thank **Ms. Seema Bawa** for the Software Engineering concepts she taught. I am thankful to **Mr. Rajeev Sharma** and **Mr. Karun Verma** for the guidance and review of conceptual design of this system. Each one of the faculty members mentioned taught and helped in this project, without whose co-operation this project would not be completed on time in spite of a short schedule and oft-missed deadlines.

I own my regards to the entire faculty of the Department of Computer Science & Engineering at Thapar University, Patiala from where I have been learning the basics of computer science since July, 2008 and whose informal discussions and able guidance became light for me in the entire duration of this work. I owe thanks to **Ms. Saloni Jindal** from Shaheed Bhagat Singh College of Engineering & Technology, Ferozepur who proved to be right-hand man and whose participation as a team member encouraged me to look on the brighter side in the bad times and finally, to all the behind-the-scenes people who chipped in with help I'm not even aware of.

MONISH GUPTA

CERTIFICATE

Regn. No. CSED/SWST2010/ 60



THAPAR UNIVERSITY


(Established under Section 3 of UGC Act, 1956)

COMPUTER SCIENCE & ENGINEERING DEPARTMENT


Certificate

It is hereby certified that Mr./Ms. Monish Gupta
student of Thapar University, Patiala has successfully completed Six
Weeks Summer Training on "JAVA - ORACLE" held from May 31" to July 9", 2010 organized by the Department of Computer
Science & Engineering, Thapar University, Patiala.

PATIALA
Date : July 9th, 2010


AJAY KUMAR
(Co-coordinator)


Dr. A.K. VERMA
(Coordinator)


Dr. RAJESH BHATIA
(Head, CSED)

01/01/2008

TABLE OF CONTENTS

TOPIC	PAGE NO.
1. Summary of the project	
a. Problem Statement	1
b. Solution	2
2. Introduction to the development environment	
a. Hardware Requirements	3
b. Software Requirements	3
c. System Development Life Cycle Model	3
d. Client-Server Architecture	5
e. Introduction to Java	6
f. Java Database Connectivity (JDBC)	11
g. Oracle	14
3. Project	
a. Overview	15
b. Procedure and Snapshots	16
4. Design Phase including DFD/E-R diagram	
a. Entities	22
b. Relation between entities	22
c. Key attributes	22
d. E-R Diagram	23
e. Assumptions	23
f. Database Design	24
5. Details of the work program including work program	26
6. Testing	29
7. Conclusion	34
8. Future Scope of the work	34
9. References	34

1. SUMMARY OF THE PROJECT

1. A Problem Statement

The institute currently operates under a manual system for making timetables. The primary function of the person who is going to make the timetable is to remember the information regarding the teachers, batches and rooms.

So there are the five major areas in which the manual system affects costs

- The system is labor and space intensive
- The manual work produces a risk of missing information regarding any time schedule of a teacher or any batch.
- It is very time consuming.
- No back up exists to support disaster recover.
- Updating, insertion and deletion become difficult.

Drawbacks of the Existing System

Due to large amount of data, a lot of complexities are involved in maintaining, updating and retrieving selected information, since old system is totally maintained manually, some of the complexities involved in existing system are as follows:

- **Redundancy of timetable:** In the existing system, all information is maintained for different departments. This often leads to considerable duplication of the information of any room number or teacher. May be room number may become same at a particular time or may be same teacher assigned at two class rooms at same time.
- **Difficulty in Updating the information:** Problem of updating data in the existing systems since there may be the case in which a particular teacher will replace or a teacher will give resign from their job so updating came in role which become difficult in manual systems. It is very difficult and time consuming to update all information.
- **Delay in retrieving information:** Retrieving information is quite time consuming process because numbers of teachers as well numbers of batches are more in colleges. If it is required to locate a particular record, a large number of paper work is needed.

Timetable Management System

- **Problem of keeping the information:** As the information is voluminous, more paper work is required to store it. Also, everything is stored on papers, so this type of storage is more prone to damage with time and due to other accidental factors.

Objective of New System

The new system is based on computer system. Our team had worked on a project in which we can make the timetable on the computer. Main objective of this project is:-

- **Less Time Consuming:** By the project it become easy to find out in which room, a particular teacher is giving lecture to a particular batch. To get all such information it takes very less time.
- **Easy To Retrieve the information:** Information regarding the teacher like either they are free at particular time or not become easy to find out
- **Updating is easy:** If a teacher want to change their timetable schedule or they want to change the timing of any one of their lecture then it is easy to update the data. In case a teacher will be replace from their position then also updating is easy.
- **Record Keeping is Easy:** In this we do not require any paper work it become easy to keep the record for the long period of time.

1. B Solution

My project is on timetable management system. I have considered five working days and nine periods as in generally held in colleges. On this project initially one has to insert Teacher Name, Course Name, Batch Name and Room Number as present in particular college. When all data is inserted it will saved in database and when a particular teacher, course or batch selected corresponding timetable will be displayed. In this if the timetable is already given then it will be shown in table but if it is not given the there is option to save the timetable.

This project will be helpful for preparing timetable within no time. One just has to insert data about the teacher and timetable will be displayed. As now, everything is online. By this, anyone can see timetable on his\her system.

2. INTRODUCTION TO THE DEVELOPMENT ENVIRONMENT

2. A Hardware Requirement

- Primary memory 65MB
- Hardware Platform Intel based 32 bit
- Processer 300 MHz
- Secondary Memory 600 MB

2. B Software Requirement

- Oracle 10g along with JDBC (Java Database Connectivity Drivers) drivers viz. ojdbc14.jar
- Java
- Netbeans 6.5.1 Java Editor

2. C SYSTEM DEVELOPMENT LIFE CYCLE MODEL

This is also known as Classic Life Cycle Model (or) Linear Sequential Model (or) Waterfall Method. This has the following activities.

1. System/Information Engineering and Modeling
2. Software Requirements Analysis
3. Systems Analysis and Design
4. Code Generation
5. Testing
6. Maintenance

System/Information Engineering and Modeling

As software is always of a large system (or business), work begins by establishing requirements for all system elements and then allocating some subset of these requirements to software. This system view is essential when software must interface with other elements such as hardware, people and other resources. System is the basic and very critical requirement for the existence of software in any entity. So if the system is not in place, the system should be engineered and put in place. In some cases, to extract the maximum output, the

system should be re-engineered and spruced up. Once the ideal system is engineered or tuned, the development team studies the software requirement for the system.

Software Requirement Analysis

This is also known as feasibility study. In this phase, the development team visits the customer and studies their system. They investigate the need for possible software automation in the given system. By the end of the feasibility study, the team furnishes a document that holds the different specific recommendations for the candidate system. It also includes the personnel assignments, costs, project schedule, and target dates. The requirements gathering process is intensified and focused specially on software. To understand the nature of the program(s) to be built, the system engineer ("analyst") must understand the information domain for the software, as well as required function, behavior, performance and interfacing. The essential purpose of this phase is to find the need and to define the problem that needs to be solved.

System Analysis and Design

In this phase, the software development process, the software's overall structure and its nuances are defined. In terms of the client/server technology, the number of tiers needed for the package architecture, the database design, the data structure design etc are all defined in this phase. A software development model is created. Analysis and Design are very crucial in the whole development cycle. Any glitch in the design phase could be very expensive to solve in the later stage of the software development. Much care is taken during this phase. The logical system of the product is developed in this phase.

Code generation

The design must be translated into a machine-readable form. The code generation step performs this task. If the design is performed in a detailed manner, code generation can be accomplished without much complication. Programming tools like Compilers, Interpreters, and Debuggers are used to generate the code. Different high level programming languages

like C, C++, Pascal, and Java are used for coding. With respect to the type of application, the right programming language is chosen.

Testing

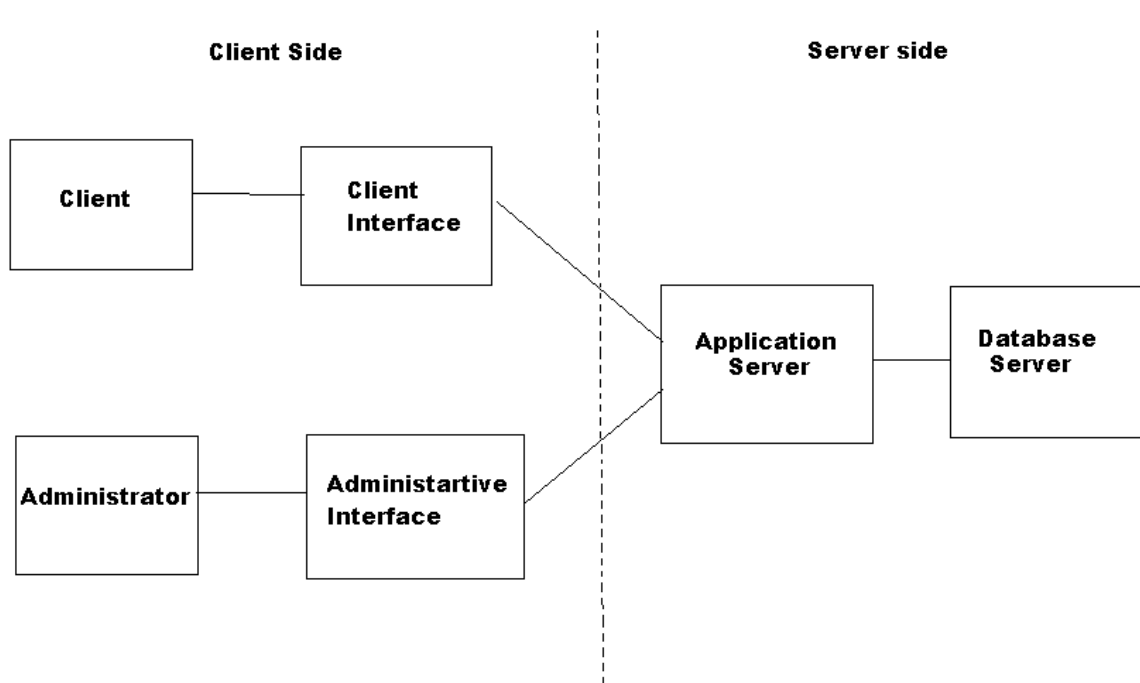
Once the code is generated, the software program testing begins. Different testing methodologies are available to unravel the bugs that were committed during the previous phases. Different testing tools and methodologies are already available. Some companies build their own testing tools that are tailor made for their own development operations.

Maintenance

Software will definitely undergo change once it is delivered to the customer. There are many reasons for the change. Change could happen because of some unexpected input values into the system. In addition, the changes in the system could directly affect the software operations.

2. D CLIENT-SERVER ARCHITECTURE

The client will have client interface in which he can interact with the system.



2. E INTRODUCTION TO JAVA

Introduction

James Gosling, Patrick Naughton, Chris Warth, Ed Frank and Mike Sheridan conceived Java at SUN Micro Systems Inc Corporation in 1991. It took 18 months to develop the first working version. This language was initially called "OAK", but was renamed "JAVA" in 1995. Before the initial implementation of OAK in 1992 and the public announcement of Java in 1995, many more contributed to the design and evolution of the language.

Overview:

Java is powerful but lean on Object Oriented programming language. It has generated a lot of excitement because it makes it possible to program for Internet by creating applets, programs that can be embedded in web page. The context of an applet is limited only by one's imagination. For example, an applet can be an animation with sound, an interactive game or a ticker tape with constantly updated stock prices. Applets can be just little decorations to liven up web page, or they can be serious applications like word processors or spreadsheet.

But Java is more than a programming language for writing applets. It is being used more and more for writing standalone applications as well. It is becoming so popular that many people believe it will become standard language for both general purpose and Internet programming.

There are many buzzwords associated with Java, but because of its spectacular growth in popularity, a new buzz word has appeared ambiguous. Indeed, all indications are that it will soon be everywhere.

Java builds on the strength of C++. It has taken the best features of C++ and discarded the more problematic and error prone parts. To this lean core, it has added garbage collection (automatic memory management), multithreading (the capacity for one program to do more than one thing at a time), security capabilities. The result is that Java is simple, elegant, powerful and easy to use

Java is actually a platform consisting of three components:

- Java is a programming language.

- Java is library of classes and interfaces.

- Java is a Virtual Machine.

The following sections will say more about these components:

JAVA IS PORTABLE:

One of the biggest advantages Java offers is that it is portable. An application written in Java will run on all the major platforms. Any computer with a Java based browser can run the applications or applets written in the Java programming language. A programmer no longer

has to write one program to run on a Macintosh, another program to run on a Windows machine, still another to run on a UNIX machine, and so on. In other words, with Java, developers write their programs only once. The Virtual Machine is what gives Java its cross platform capabilities. Rather than being compiled into Machine language, which is different for each operating systems and computer architecture, Java code is compiled into byte codes.

JAVA IS OBJECT-ORIENTED:

The Java programming language is object oriented, which makes program design focus on what you are dealing with rather than on how you are going to do something. This makes it more useful for programming in sophisticated projects because one can break the things down into understandable components. A big benefit is that these components can then be reused.

Java includes inheritance, or the ability to derive new classes from existing classes. The derived class, also called a subclass, inherits all the data and the functions of the existing class, referred to as the parent class. A subclass can add new data members to those inherited from the parent class. As far as methods are concerned, the subclass can reuse the inherited methods, as it is, change them, and/or add its own new methods.

JAVA MAKES IT EASY TO WRITE CORRECT CODE:

In addition to being portable and object oriented, Java facilitates writing correct code. Programmers spend less time writing Java code and a lot less time debugging it. In fact, developers have reported slashing development time by as much as two thirds. The following is a list of some of Java's features that make it easier to write correct code.

GARBAGE COLLECTION:

Automatically takes care of allocating and de allocating memory, a huge potential source of errors. If an object is no longer being used (has no references to it), then it is automatically removed from memory, or "Garbage Collected". Programmers don't have to keep track of what has been allocated and de-allocated, which makes their job a lot easier, but, more importantly it stops memory leaks.

NO POINTERS:

Eliminates big source errors. By using object references instead of many pointers, problems with pointer arithmetic are eliminated, and problems with inadvertently accessing the wrong memory address are greatly reduced.

STRONG TYPING:

Java cuts down runtime errors. Because Java enforces strong type checking, many errors are caught when code is compiled. Dynamic binding is possible and often very useful, but static binding with strict type checking is used when possible.

SIMPLICITY:

Java is made easier to learn and use correctly. Java keeps it simple by having just one way to do something instead of having several alternatives, as in some languages. Java also stays lean by not including multiple inheritance, which eliminates the errors and ambiguity that arise when you create a subclass that inherits from two or more classes. To replace capabilities, multiple inheritances provide Java lets you add functionality to a class throw the use of interfaces.

JAVA INCLUDES A LIBRARY OF CLASSES AND INTERFACES:

The Java platform includes an extensive class library so that programmers can use already existing classes as it is, create subclasses to modify existing classes, or implement interfaces to augment the capabilities of classes.

Both classes and interfaces contain data members (fields) and functions (methods), but there are major differences. In a class, fields may be either variable or constant, and methods are fully implemented. In an interface, fields must be constants, and methods are just prototypes with no further implementations. The prototypes give the method signature (the return type, the function name, and the number of parameters, with the type for each parameter), but the programmer must supply implementations. To use an interface, a programmer defines a class, declares that it implements the interface, and then implements all the methods in that interface as part of the class.

These methods are implemented in a way that is appropriate for the class in which the methods are being used. Interfaces let one add functionality to a class and give a great deal of flexibility in doing it. In other words interfaces provide most of the advantages of multiple inheritance without its disadvantages.

A package is a collection of related Java classes and interfaces. The following list, though not complete, gives examples of some Java packages and what they cover.

Java.lang – This package is so basic that it is automatically included in any Java program. It includes classes dealing with numeric, strings, objects, runtime, security and threads.

Java.io -- Classes that manages reading data from input streams and writing data to the output streams.

Java.util -- Miscellaneous utility classes, including generic data structures, bit sets, time, date, the string manipulation, random number generation, system properties, notification and enumeration of data structures.

Java.net -- Classes for network support.

Java.awt --Classes that manage user interface components such as windows, dialog boxes, buttons, check boxes, lists, menus, scrollbars, and text fields; the "AWT" stands Abstract Window Toolkit.

Java.awt.image -- Classes for managing image data, including color models, dropping, color filtering, setting pixel values, and grabbing snapshots.

Java.applet -- The Applet class, which provides the ability to write applets, this package also includes several interfaces that connect an applet to its document and to resources for playing audio.

Java.sql -- The JDBC API, classes and interfaces that access databases and send SQL statements.

The first three packages listed, java.lang, java.io, and java.util form the foundation, and they are the basic classes and interfaces for general-purpose programming. Java Development Kit Version 1.1 added some new packages, with JDBC is being one of them. Other new packages include such things as Remote Method Invocation, Security and Java Beans the new API for creating reusable components.

JAVA IS EXTENSIBLE:

A big plus for Java is the fact it can be extended. It was purposely written to be lean with the emphasis on doing what it does very well; instead of trying to do everything from the beginning, it was return so that extending it is very easy. Programmers can modify existing classes or write their own new classes or they can write a whole new package. The JDBC API, the java.sql package, is one example of a foundation upon which extensions are being built. Other extensions are being added or worked on in area such as multimedia, Internet Commerce, conferencing, and Telephony.

JAVA IS SECURE

It is important that a programmer unable to write subversive code for applications or applets. This is especially true with the Internet being used more and more extensively for services such as electronic commerce and electronic distribution of software and multimedia contents. The Java platform builds security in four ways.

The way memory is allocated and laid out: In Java an object's location in memory is not determined until the runtime, as opposed to C and C++, where the compiler makes memory layout decisions. As a result, a programmer can not figure out by looking at a class definition, how it might be laid in memory.

The way incoming code is checked: The Java Virtual Machine doesn't trust any incoming code and subjects it to what is called Byte Code Verification. The byte code verifier, part of the virtual machine, checks that

The format of incoming code is correct

Incoming code doesn't forge pointers

It doesn't violate access restrictions

JAVA PERFORMS WELL:

Java's performance is better than one might expect. Java's many advantages, such as having built in security and being interpreted as well as compiled; do have a cost attached to them. However, various optimizations have been built in, and the byte code interpreter can run very fast the cost it doesn't have to do any checking. As a result, Java has done quite respectably in performance tests. Its performance numbers for interpreted byte codes are usually more than adequate to run interactive graphical end user applications.

JAVA IS ROBUST:

The multi platform environment of the WEB places extraordinary demands on a program, because it must execute reliably in a variety of systems. Thus the ability to create robust programs was given a high priority in the design of Java. To gain reliability, Java restricts you in a few key areas to force you to find your mistakes early in program developments. At the same time, Java frees you from having to worry about many of the most common causes of programming errors. Because Java is strictly typed language, it checks your code at compile time. However, it also checks your code at run time. In fact, many hard to track down bugs that often turn up in hard to reproduce runtime situations are simply impossible to create in Java. Knowing that what you have written will behave in a predictable way under diverse conditions is a key feature of Java.

JAVA SCALES WELL:

Java platform is designed to scale well, from portable consumer electronic devices to powerful desktop and server machines. The Virtual Machine takes a small footprint and Java byte code is optimized to be small and compact. As a result, Java accommodates the need for low storage and for low bandwidth transmission over the Internet. In addition the Java Operating System offers a standalone Java platform that eliminates host operating system overhead while still supporting the full Java platform API. This makes Java ideal for low cost network computers whose sole purpose is to access the Internet.

JAVA IS MULTI-THREADED:

Multithreading is simply the ability of a program to do more than one thing at a time. For example an application could be faxing a document at the same time it is printing another document. Or a program could process new inventory figures while it maintains a feed for current prices. Multithreading is particularly important in multimedia: a multimedia program might often be running a movie, running an audio track and displaying text all at the same time.

2.F JAVA DATABASE CONNECTIVITY (JDBC)

JDBC overview:

JDBC is a Java API for executing SQL statements. (JDBC is a trademarked name and is not an acronym; nevertheless, JDBC is often thought of as understanding for "Java Database Connectivity".) It consists of a set of classes and interfaces written in the Java programming language. JDBC provides a standard API for tool/database developers and makes it possible to write database applications using a pure Java API.

Using JDBC, it is easy to send SQL statements to virtually any relational database. In other words, with the JDBC API, it is not necessary to write one program to access a Sybase database, another program to access an Oracle database, another program to access an Informix database, and so on... One can write a single program using the JDBC API, and the program will be able to send SQL statements to the appropriate database. And with in an application written in Java programming language, one also doesn't have to worry about writing different applications to run on different platforms. The combination of Java and JDBC lets a programmer to write it once run it anywhere.

JDBC extends the concepts, which can be done in Java. For example, with Java and JDBC API, it is possible to publish a web page containing an applet that uses information obtained from a remote database or an enterprise can use JDBC to connect all this employees (even if they are using conglomeration of Windows, Macintosh, and Unix machines) to one or more internal databases via an Internet. With more and more programmers using the Java programming language, the need for easy database access from Java is continuing to grow.

MIS managers like the combination of Java and JDBC because it makes disseminating information easy and economical. Businesses can continue to use their installed databases and access information easily even of it is stored on different database management systems. Development time for new applications is short. Installation and version controls are greatly simplified. A programmer can write an application or an update once, put it on the server, and everybody has access to the latest version. And for business selling information services, Java and JDBC offer a better way of getting out information updates to external customers.

What does JDBC do?

Simply put, JDBC makes it possible to do three things:

1. Establish a connection with the database
2. Send SQL statements
3. Process the results

The following code fragment gives a basic example of these three steps:


```
Connection con=DriverManeger.getConnection (“Jdbc:odbc:dsnName”, “login”,  
“password”);
```

JDBC is a low-level API and a base for high-level APIs:

JDBC is a “low-level” interface, which means that it is used to invoke (or “call”) SQL commands directly. It works very well in this capacity and is easier to use than other database connectivity. JDBC APIs were also designed to be a base upon which to build higher-level interfaces and tools. A higher-level interface is “user-friendly”, using a more understandable or more convenient API that is translated behind the scenes into a low-level interface such as JDBC. At present, two kinds of higher-level APIs are under development on top of JDBC:

An embedded SQL for Java: At least one vendor plans to build this. DBMS implement SQL, a language designed specifically for use with databases. JDBC requires that SQL statements be passed as strings to Java methods. An embedded SQL processor allows a programmer to instead mix SQL statements directly with Java: for ex: a Java variable can be SQL statement to receive or provide SQL mix into Java with JDBC calls.

JDBC Interfaces:

Connection Overview:

A connection object represents a connection with a database. Connection session includes the SQL statements that are executed and the results are returned over that connection. A single application can have one or more connections with a single database, or it can have connections with many different databases.

Opening a connection:

The standard way to establish a connection with a database is to call the method `DriverManager.getConnection`. This method takes a string containing a URL. The `DriverManager` class, referred to as the JDBC manager layer, attempts to locate a driver than it connects to the database represented by that URL. The `DriverManager` class maintains a list of registered Driver classes, and when the method `getConnection` is called, it checks with each driver in the list until it finds one that can connect to the database specified in the URL. The driver method connects uses this URL to actually establish the connection.

The user can by pass the JDBC management layer and call driver methods directly. This could be useful in the rare case that two drivers can connect to a database and the user wants to explicitly select a particular driver. Normally, however, it is much easier to just let the `DriverManager` class handle opening a connection.

The following code exemplifies opening a connection to a database located at the URL "jdbc: ODBC: wombat" with the user ID of "oboy" and "12Java" as the password:

```
String URL = "jdbc: ODBC: wombat";
```

```
Connection con=DriverManager.getConnection (URL,"oboy","12Java");
```

DriverManager Overview:

The DriverManager class is the management layer of JDBC, working between the user and the drivers. It keeps track of the drivers that are available and handles establishing a connection between a database and the appropriate driver. In addition, the DriverManager class attends to things like driver login time limits and the printing of log and tracing messages.

For simple applications, the only method in this class that a general needs to use directly is DriverManager.getConnection. As its name implies, this method establishes a connection to a database. JDBC allows the user to call the DriverManager methods getDriver, getDrivers, and registerDriver as well as the driver method connect, but in most cases it is better to let the DriverManager class manage the details of Establishing a Connection Statement:

Statement Overview:

A statement object is used to send SQL statements to a database. There are actually three kinds of statement objects, all of which act as containers for executing SQL statements on a given connection: Statement, Prepared statement, which inherits from statement, and callable statement, which inherits from prepared statement. They are specialized for sending particular type of SQL statements; a Statement object is used to execute a simple SQL statement with no parameters; a Prepared Statement object is used to execute a precompiled SQL statement with or without IN parameter; and a callable statement object is used to execute a call to the database stored procedure.

The statement interface provides basic methods for executing statements and retrieving results. The PreparedStatement interface adds methods for dealing with IN parameters; CallableStatement adds methods for dealing with OUT parameters.

2.g ORACLE

ORACLE is a relational Database management system that as a transparent interface between the physical storage and logical presentation of data and provides the user with a set of flexible & sophisticated tools to perform the operations basing on data and the data structures. Oracle is a modular system that consists of the Oracle database and several functional packages .

Oracle tools do 3 major kinds of work :

Database Management

Data access & manipulation

Programming

For handling information the user can use these tools to:

Define a database

Query a database

Add,update,edit & delete data

Modify the structure of database

Export & Import data.

Connectivity between back-end & front-end

Database Management tools:

This usually includes the core programming of oracle's database management system .The oracle database with its associated tables & views are stored in the oracle data dictionary.The data dictionary stores information related to every fact of the database sytem
Data access & manipulation tools:

ORACLE uses the SQL command set for the purpose of data access & its manipulation .Structured Query Language(SQL) includes ali commands which are used for data access.

Reason for choosing ORACLE:

- Applications developed on Oracle are completely portable to other versions of programming .Programmers create a complex application in a single user environment and then move it to multiuser platform.
- Oracle has several features that ensures integrity of user database.Database can be made secure by restrictions access to only authorized users using the tools available in ORACLE.
- Oracle can perform effectively on large database .It offers unique clustering technique for storing data on the disk. ORACLE database works effectively in a Intranet/Internet environment. Since the Internet applications demands the data to be secure, ORACLE can be chosen as the right tools for maintaining databases.

3. PROJECT

3.a Overview

In the first frame, there are four comboboxes for teacher, batch, room and course. Corresponding to every combobox there is the option of three basic operations of database i.e. insert, delete and update. Every combobox has a text box also in this initially --SELECT-- option is entered. Until the person does not enter the valid entry of teacher, batch, room and course it will display only select option. All the new information that we will add enter into the database and entry will be saved there. Any time information can be taken easily. If we have entered the wrong entry or forget to select any teacher name, course id, batch id or room id it will display the message at the bottom of frame. It will give the instruction to enter the correct information.

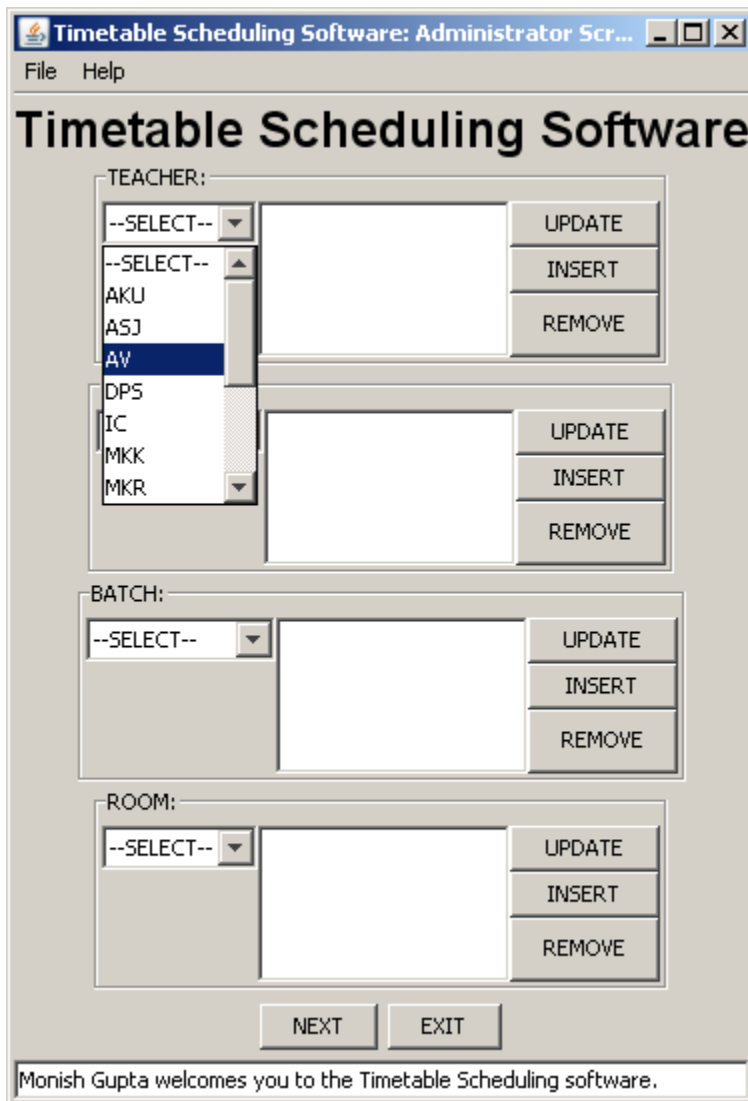
At the bottom of the frame there is a button schedule timetable on clicking it we come on the next frame. When we go to next frame the button that we have selected corresponding to that timetable will be displayed. Like in general colleges there are nine periods in a day and five working days, similarly in this project I have consider nine periods in a day and five working days. The particular teacher working in which room to which batch can be easily find out. Corresponding to that, there are two buttons **Edit** and **Save**. Edit is for changing or update the information that already exist in the database. The edit data will be saved in database by clicking on the save button. A teacher is teaching in which room in a particular day and period can be found out by selecting any one of option. The timetable corresponding teacher will display. At the bottom of the frame there is the button of back which will come on the first frame.

3.b Procedure and Snapshots

This is the first frame of my project there are four comboboxes like batch, course, room and teacher.

The screenshot shows a Windows-style application window titled "Timetable Scheduling Software: Administrator Scr...". The window has a menu bar with "File" and "Help". The main title is "Timetable Scheduling Software". Below the title, there are four sections, each for a different entity: TEACHER, COURSE, BATCH, and ROOM. Each section contains a dropdown menu with "--SELECT--", a text input field, and three buttons: UPDATE, INSERT, and REMOVE. At the bottom of the window, there are two buttons: NEXT and EXIT. A status bar at the very bottom says "Monish Gupta welcomes you to the Timetable Scheduling software."

In this corresponding to every combobox there is three buttons i.e. insert, update and delete. Means existed data can be changed. Corresponding to every entry there is comments displayed with the help of this one can easily read the instructions given there. At the bottom of the frame a button is added i.e. Schedule Timetable. The parameter that we will select from displayed comboboxes, corresponding timetable will appear. When we click on the next button, we come to the next frame.



Timetable Scheduling Software: Administrator Scr...

File Help

Timetable Scheduling Software

TEACHER:

--SELECT--

AKU
ASJ
AV
DPS
IC
MKK
MKR

UPDATE
INSERT
REMOVE

BATCH:

--SELECT--

UPDATE
INSERT
REMOVE

ROOM:

--SELECT--

UPDATE
INSERT
REMOVE

NEXT EXIT

Monish Gupta welcomes you to the Timetable Scheduling software.

In this we have selected one parameter i.e. teacher and corresponding to that the timetable will be displayed. Now click on button schedule timetable.

Timetable Management System

After clicking on the button schedule timetable button, we came to this frame at the top of the frame there are three comboboxes i.e. for batch, teacher and room. It displays the timetable after selecting any one of the parameter out of them. When we select the particular parameter the timetable come according to the working days in colleges. Five days are considered in rows and the nine periods are in the columns.

Timetable Management System

Timetable Scheduling Software: Administrator Screen

File Help

BATCH: --SELECT-- TEACHER: --SELECT-- ROOM: --SELECT--

Period 1 Period 2 Period 3 Period 4 Period 5 Period 6 Period 7 Period 8

MON

TUE

WED

THU

FRI

BACK EXIT

Select any section/teacher/room to schedule or view the timetable.

After that one parameter is selected corresponding to which we want to display the timetable. Suppose we have selected the one particular teacher 'AV', and then the timetable schedule will come.

Timetable Management System

Timetable Scheduling Software: Administrator Screen

File Help

BATCH: --SELECT-- TEACHER: AV ROOM: --SELECT--

	Period 1	Period 2	Period 3	Period 4	Period 5	Period 6	Period 7	Period 8
MON	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save
TUE	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save
WED	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save
THU	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save
FRI	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save	-Select COUR... -Select BATCH- -Select ROOM- Save

BACK EXIT

Timetable for TEACHER: AV loaded.

When we select the particular teacher then there is no timetable schedule for them. So to add the timetable of the teacher there are different comboboxes for batch, room and course. These three parameters are collected in one GridBag. This shows the schedule of teacher in particular period and the day. Corresponding to that at the bottom of the GridBag a button is added for save the new information of teacher. The whole information will go into the database and required data can be retrieved from database.

Timetable Management System

Timetable Scheduling Software: Administrator Screen

File Help

BATCH: --SELECT-- TEACHER: AV ROOM: --SELECT--

	Period 1	Period 2	Period 3	Period 4	Period 5	Period 6	Period 7	Period 8
MON	CA009 L	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...
	COE1	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-
	F101	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-
	Edit	Save	Save	Save	Save	Save	Save	Save
TUE	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...
	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-
	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-
	Save	Save	Save	Save	Save	Save	Save	Save
WED	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...
	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-
	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-
	Save	Save	Save	Save	Save	Save	Save	Save
THU	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...
	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-
	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-
	Save	Save	Save	Save	Save	Save	Save	Save
FRI	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...	-Select COUR...
	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-	-Select BATCH-
	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-	-Select ROOM-
	Save	Save	Save	Save	Save	Save	Save	Save

BACK EXIT

Entry successfully inserted..

In this we have added the time schedule of the selected teacher. Their course, batch and room no. is entered and the data will enter into the database. So it will be saved there, in other case if we want to edit or change the schedule of teacher then there is a button edit. On clicking the button the data can be edited and saved into database. On every new entered information, there is the message displayed at the bottom of frame which give us the information that whether the entered data successfully inserted or not. Like in this frame message is displayed that 'Entry successfully inserted'. Also at the bottom a button is added i.e. back on clicking this button we will able to go on the previous frame.

4. DESIGN PHASE INCLUDING DFD/E-R DIAGRAMS

4.a ENTITIES:

The entities in the timetable management system are given below:-

1. Teachers
2. Batches
3. Courses
4. Rooms

4.b RELATIONSHIP BETWEEN ENTITIES:

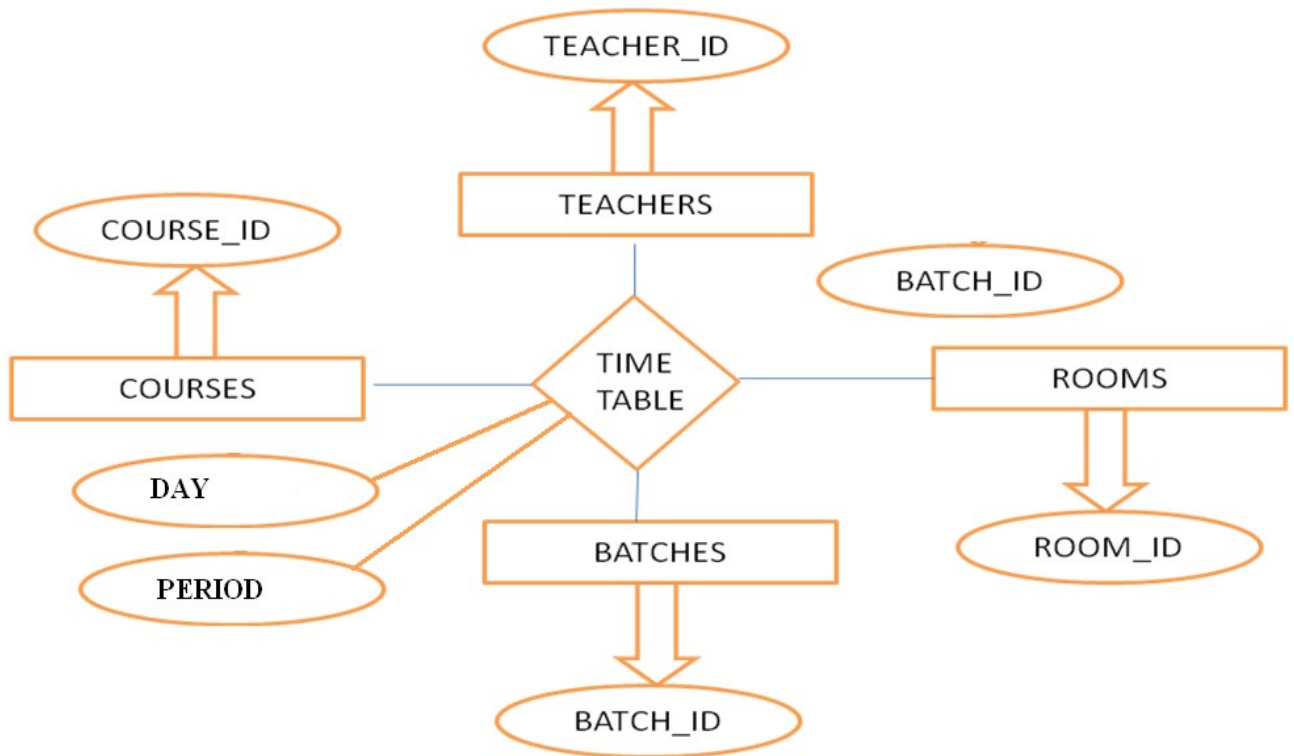
1. A single teacher can teach many courses and a single course can be taught by many teachers. So, it is a many to many relation.
2. One teacher can teach many batches and a single batch can be taught by many teachers so it is also a many to many relation.
3. One teacher can teach in many rooms at different time so it is a many to many relationship and vice-versa is also true.
4. A single course can be learnt by different batches and a single batch can do many courses. Hence, it is a many to many relation.
5. A single batch can study in different rooms and in a single room many batches can come so it is also a many to many relation.
6. One course can be taught in different rooms and vice-versa is also true so it is a many to many relation.

4.c KEY ATTRIBUTES:

1. In teacher's table the key attribute is teacher_id.
2. In course's table the key attribute is course_id.
3. In batch's table the key attribute is batch_id.
4. In room's table the key attribute is room_id.

Other relevant attributes are day and period TT table.

4.d E-R DIAGRAM:



4.e ASSUMPTIONS:

1. In timetable management system, we have many teachers in a particular college, a number of courses, rooms and batches.
2. In the timetable there are **five working days** (vertically) and **nine periods** (horizontally) considered.
3. According to availability in a college insert the name of teachers, corresponding to that enter the course and batch which that teacher will teach. Enter the particular room number also.
4. After entering all the information, on selecting any one entity corresponding to that selected entity timetable will be displayed.
5. If data corresponding to the selected entity is not available then in displayed timetable there is the option to save the information, by this information can be entered there.

6. If in the displayed timetable any information is wrongly entered then there is an option to edit that particular information. The edited information will save in database.

After making all the timetable the person want to insert or update the data then there is an option to go 'BACK', then the front page will displayed and three basic operations can be done over there i.e. Insertion, deletion and updation.

4.F DATABASE DESIGN

Teacher Table:

TABLE	COLUMN	DATA TYPE	LENGTH
Teacher	ID	Varchar2	20

Course Table :

TABLE	COLUMN	DATA TYPE	LENGTH
Course	ID	Varchar2	20

Room Table :

Timetable Management System

TABLE	COLUMN	DATA TYPE	LENGTH
Room	ID	Varchar2	20

Batch Table :

TABLE	COLUMN	DATA TYPE	LENGTH
Batch	ID	Varchar2	20

TT Table :

TABLE	COLUMN	DATA TYPE	LENGTH
TT	BID	VARCHAR2	20
	CID	VARCHAR2	20
	RID	VARCHAR2	20
	TID	VARCHAR2	20
	DAY	NUMBER	-
	PERIOD	NUMBER	-

5. DETAILS OF THE WORK INCLUDING WORK PROGRAM

Date	Work and learning
1 st Jun, 2010	<p>Compilation process</p> <p>Features of java</p> <p>Compilation in java</p> <p>JVM architecture</p> <p>Flavors of java- SE, EE, ME</p> <p>Installation and Configuration</p> <p>Hello World Program</p>
2 nd Jun, 2010	<p>Basic steps in programming viz. start, problem definition, coding, designing, compilation and linking, testing.</p> <p>Tokens: identifiers, keywords, literals, operators, other separators</p>
3 rd Jun 2010	<p>Keywords in java</p> <p>Escape Sequences</p> <p>Taking input from the user</p> <p>Use of final and static keywords</p> <p>Differences between C/C++ and Java</p> <p>JVM architecture</p> <p>Java: Applets, Servlets, Desktop application</p> <p>Wrapper Classes</p> <p>CORBA</p> <p>Static Members</p> <p>Inheritance</p>

Timetable Management System

10 th Jun, 2010	<p>Object slicing</p> <p>Applets and applet tag in HTML</p> <p>Packages - lang, util, net, swing, awt, io</p> <p>2-D Graphics - Font, line, rectangle and other shapes</p> <p>Dynamic Method Dispatch (Polymorphism)</p> <p>Abstract Classes</p> <p>Oracle introduction</p> <p>Problems with manual file-system</p> <p>advantages and disadvantages in RDBMS</p> <p>Concept of Entity and Referencial Integrity</p> <p>Types of keys : Super, Candidate, primary, alternate, artificial, foreign</p> <p>SQL: DCL & DML statements viz. Create, select, insert, update, delete, Alter</p> <p>Implementation of Entity and Referencial Integrity</p> <p>Contraints: not null, check, like, on delete</p> <p>Functions - SUM, AVG, COUNT, MIN, MAX</p> <p>JAVA - Autoboxing</p> <p>Implementing interfaces</p> <p>Software Delevopment - Decision tables and Decision trees</p>
22 nd Jun, 2010	<p>File handling</p> <p>Object Serialization</p> <p>GUI swing package viz. JButton, JFrame, JPanel, Layout Managers</p> <p>Implementing Event Listeners</p>

Timetable Management System

26 th Jun,2010	Oracle - use of Group by - having clause Concept of JDBC and its implementation Java - Adding MenuBar in JFrame JSliders DBMS - Conceptual Design - E-R Diagram Normalization De-normalization Introduction to Software Engineering Wearable Computing
---------------------------	---

Requirement gathering and Design of the timetable management system was started on around 26th Jun. Further the coding was done in the first week of July, 2010.

6. TESTING (TEST CASES)

On the first frame, pressing any of the 3 buttons (viz. UPDATE, INSERT, REMOVE) when '--SELECT--' is selected from the list, produce an error message which is displayed at the bottom as shown in the following cases.

On pressing REMOVE button when '--SELECT--' is selected from the list, produce an error message: --SELECT-- NOT deleted as shown in the following diagram:

The screenshot shows the 'Timetable Scheduling Software: Administrator Scr...' window. It contains four sections: TEACHER, COURSE, BATCH, and ROOM. Each section has a dropdown menu with '--SELECT--' selected, a text input field, and three buttons: UPDATE, INSERT, and REMOVE. The REMOVE button in the TEACHER section is highlighted with a dashed border. At the bottom, there are 'NEXT' and 'EXIT' buttons. A status bar at the very bottom displays the message '--SELECT-- NOT deleted'.

Section	Dropdown	Text Field	Buttons
TEACHER:	--SELECT--		UPDATE, INSERT, REMOVE (highlighted)
COURSE:	--SELECT--		UPDATE, INSERT, REMOVE
BATCH:	--SELECT--		UPDATE, INSERT, REMOVE
ROOM:	--SELECT--		UPDATE, INSERT, REMOVE

Navigation: NEXT, EXIT

Status: --SELECT-- NOT deleted

Timetable Management System

On pressing update button when nothing is mentioned in the text area, produces the error message: ERROR: AKU NOT replaced by

The screenshot shows the 'Timetable Scheduling Software: Administrator Screen' window. It has a menu bar with 'File' and 'Help'. The title bar reads 'Timetable Scheduling Software'. The main area contains four sections: 'TEACHER:', 'COURSE:', 'BATCH:', and 'ROOM:'. Each section has a dropdown menu, a text area, and three buttons: 'UPDATE', 'INSERT', and 'REMOVE'. The 'TEACHER' dropdown is set to 'AKU'. The 'COURSE', 'BATCH', and 'ROOM' dropdowns are set to '--SELECT--'. At the bottom, there are 'NEXT' and 'EXIT' buttons. A status bar at the very bottom displays the error message: 'ERROR: AKU NOT replaced by'.

Section	Dropdown	Text Area	Buttons
TEACHER:	AKU		UPDATE, INSERT, REMOVE
COURSE:	--SELECT--		UPDATE, INSERT, REMOVE
BATCH:	--SELECT--		UPDATE, INSERT, REMOVE
ROOM:	--SELECT--		UPDATE, INSERT, REMOVE

ERROR: AKU NOT replaced by

Timetable Management System

On pressing Insert button when nothing is mentioned in the text area, produces the error message: ERROR: NOT inserted.

The screenshot shows the 'Timetable Scheduling Software: Administrator Screen' window. It features a menu bar with 'File' and 'Help'. The main title is 'Timetable Scheduling Software'. Below this, there are four sections for scheduling: TEACHER, COURSE, BATCH, and ROOM. Each section has a dropdown menu, a text input field, and three buttons: UPDATE, INSERT, and REMOVE. The TEACHER dropdown is set to 'AKU'. The COURSE, BATCH, and ROOM dropdowns are set to '--SELECT--'. At the bottom, there are 'NEXT' and 'EXIT' buttons. A status bar at the very bottom displays the error message: 'ERROR: NOT inserted.'

Section	Dropdown	Text Field	Buttons
TEACHER:	AKU		UPDATE, INSERT, REMOVE
COURSE:	--SELECT--		UPDATE, INSERT, REMOVE
BATCH:	--SELECT--		UPDATE, INSERT, REMOVE
ROOM:	--SELECT--		UPDATE, INSERT, REMOVE

ERROR: NOT inserted.

Timetable Management System

Once a teacher code is inserted, the system does not let it inserted again irrespective of the case (upper, lower or mix) used by the user.

The image displays two side-by-side screenshots of the 'Timetable Scheduling Software: Administrator Scr...' window. Both windows have a menu bar with 'File' and 'Help'. The title bar reads 'Timetable Scheduling Software'.

Left Screenshot: The 'TEACHER:' section has a dropdown menu with '--SELECT--' and a list of teacher codes: AKU, AV, DPS, IC, MKK, MKR, and NIN. The 'BATCH:' and 'ROOM:' sections also have '--SELECT--' dropdowns. Each section has 'UPDATE', 'INSERT', and 'REMOVE' buttons. At the bottom are 'NEXT' and 'EXIT' buttons. A status bar at the bottom says 'Monish Gupta welcomes you to the Timetable Scheduling software.'

Right Screenshot: The 'TEACHER:' section shows the dropdown menu with '--SELECT--' and the text 'ic' entered in the adjacent text box. The 'BATCH:' and 'ROOM:' sections remain the same. The status bar at the bottom says 'Monish Gupta welcomes you to the Timetable Scheduling software.'

For example:

Teacher code IC is already present in the combobox list as shown. Now if 'ic' is written in the textbox and insert button is pressed,

Timetable Management System

Following error message is displayed: ERROR: IC NOT inserted. The user might be inserting duplicate entries.

The screenshot shows a window titled "Timetable Scheduling Software: Administrator Scr...". The window has a menu bar with "File" and "Help". The main title is "Timetable Scheduling Software". Below the title, there are four sections for data entry: "TEACHER:", "COURSE:", "BATCH:", and "ROOM:". Each section has a dropdown menu with "--SELECT--", a text input field, and three buttons: "UPDATE", "INSERT", and "REMOVE". At the bottom of the window, there are "NEXT" and "EXIT" buttons. A status bar at the very bottom displays the error message: "ERROR: IC NOT inserted. The user might be inserting duplicate entries."

Most of the possible cases of improper use have been thought of and fixes are already implemented.

7. CONCLUSION

This project that I undertook was truly a very rewarding experience for me in more than one way. It has given a big thrust to my technical knowledge as prospective Software professional. It has also helped me enhance my skills on the personal front.

And I feel extremely satisfied by the fact that I have managed to develop the project of course with equal contribution from my team member. I think I have exploited the opportunity that came my way to the fullest extent by increasing my technical know-how and also gaining the valuable work experience apart from studying the other subjects in our curriculum.

8. FUTURE SCOPE OF THE WORK

Every system has some advantages and disadvantages .Similarly in my projects some small modules are missing. I tried to include all modules necessary for software but even then there is the print module which is not being considered at all. In this timetable management system, no doubt the human efforts are still required but it reduces the time consumption. In this there is no option of print out of the timetable of a particular teacher. Also sometime in colleges in some particular periods like in labs or in tuts the batch will divide into parts, so there is no such option to show out that partition.

9. REFERENCES

- Core JAVA volume 1-Fundamentals (8th Edition) by CAY S.HORSTMAANN & GARY CORNELL.
- Java2: The Complete Reference (5th Edition) by Herbert Schildt.