DON BOSCO INSTITUTE OF TECHNOLOGY

Skill Lab: C++ and Java Programming
MINI PROJECT REPORT

On

"Learning Management System using concept of Demultiplexers"
2021-22

Submitted By:

Sakshi Vadiraj Kaveri	33
Shreyas Sanjay Nanaware	39
Simran Santosh Koparkar	41

Under the guidance of
Ms. Deepali Kayande

Mini Project Title   **:** Demultiplexers

Institute Name   **:** Don Bosco Institute of Technology.

Institute Address   **:** Premier Automobiles Road,

Kurla (West), Mumbai – 400070

Department   **:** EXTC

Class   **:** SE

Project Group Members   **:**

Names of students	Roll No.
1. Sakshi Kaveri	33
2. Shreyas Nanaware	39
3. Simran Koparkar	41

Date of Submission   **:** 10/12/2021

Guide   **: Ms. Deepali Kayande**

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Project Topic: Demultiplexer

Demultiplexer is used to connect a single source to multiple destinations. The main application area of demultiplexer is communication system, where multiplexers are used. Most of the communication system are bidirectional i.e., they function in both ways (transmitting and receiving signals). A Demultiplexer (Demux) is a combinational logic circuit that includes single input and an 'n' number of outputs. The data which is obtained by a single input line can be transmitted to the 'n' number of output lines. So the function of a demultiplexer is quite opposite to a multiplexer. Multiplexers are called Data Selectors whereas Demultiplexers are Data Distributors because they transmit similar information which is obtained at the input to various outputs.\

There are different types of demultiplexers available depending on the different output configurations like 1 to 2, 1 to 4, 1 to 8 & 1 to 16. These Demultiplexers are available in various IC packages. Some of them are; 74139 IC is a dual 1 to 4 Demux, 74138 IC is a 1 to 8 Demux, 74237 IC is a 1 to 8 Demux including address lines, 74154 IC is a 1 to 16 Demux and 74159 IC is a 1 to 16 open collector Demux. So, the Demux ICs are also known as Decoder ICs.

- 1-to-2 Demultiplexer

- 1-to-4 Demultiplexer

- 1 to 8 Demultiplexer

- 1-to-16 Demultiplexer

For example in a 1-4 Demux includes a single input like D, 2-selection lines like S1 & S0 & 4 outputs like X0, X1, X2 & X3. The data at input transmits to any one of the outputs in a specified time for a specific arrangement of select lines. The 1:4 Demux block diagram and its truth table are shown below.

4
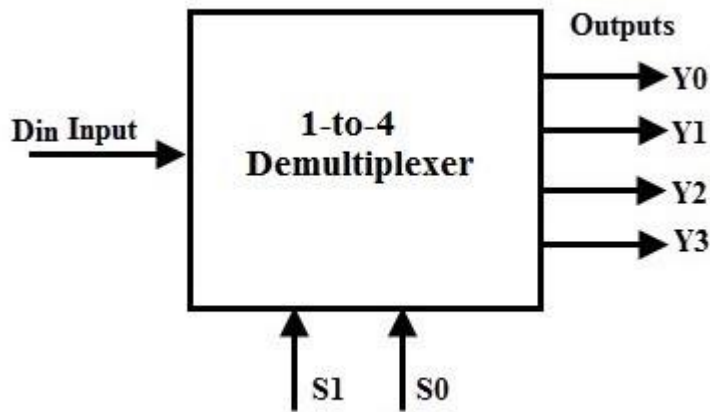
- **1 to 4 Demultiplexer Block Diagram**

- We can conclude that once both the select inputs are 0 & 1, the data input can be connected to output X0. Similarly, once selection lines S0 & S1 are 0 & 1, then data input can be connected to X1 output.

Application:

So here we applied the concept of demux in creating a template of student friendly learning management system similar to that of Moodle where the students can refer to the lecture notes and previous years question papers. Students can also ENJOY various games created by the faculty like SUDOKU, snake and ladders, cryptex and many more. They can also submit their assignments and laboratory works in various links respectively.

The basic aim is to create an useful application that is user friendly and easy to access the required material. The program is written in such a way that it instructs the user exactly what input needs to be given and If the user gives wrong input , the program will display the warning message with proper instructions.

Learning Management System software is helpful for teachers and school supervisors and also support applications and dashboards effective for students for communication and combined data collection.

LMS acts as the bottom line database to store all the day-to-day school operations, maintain the proper records of the past few decades, regulate the various operation modules, managing the critical tasks.

Earlier it was too challenging for teachers to keep track of all the activities and tasks allocated to each student, and sometimes it gets overlooked. But life has become easy with the management software. Because of its efficient dashboard, teachers can easily maintain, monitor, track the performance of every student, and eventually take appropriate measures to get it completed.

Hence due to these reasons, we decided to choose this as our project topic due to the wide range of uses and scope.

Chapter 2

Problem Statement:

Creating a Moodle like application using Demultiplexer.

Moodle where the students can refer to the lecture notes and previous years question papers. Students can also attempt various games created by the faculty like snake and ladders, cryptex and many more. They can also submit their assignments and laboratory works in various links respectively.

We have started with the idea of taking username and password of the user similar to that of Moodle. These will be our primary inputs and we will store them just as the website does for authentication purposes. After logging in successfully, a welcome message is displayed on the screen.

A **Student Management System Java Code** allows you to keep the student records and manage them when needed. This is a simple java project with a good and interactive-looking GUI. The concepts used in this application are as follows:-

1) UI

    UI is a broad term that refers to all sorts of communication between a program and its users. Even the speed with which a program interacts with the user is an important part of the program's UI. **Swing in Java** is a Graphical User Interface (GUI) toolkit that includes the GUI components. The Java Swing library is built on top of the Java Abstract Widget Toolkit (**AWT**), an older, platform dependent GUI toolkit. You can use the Java simple GUI programming components like button, textbox, etc., from the library and do not have to create the components from scratch.

2) Inheritance

    Inheritance is the concept by which one of the class acquires the properties of another class. The aim of inheritance is to provide the reusability of code so that a class has to write only the unique features and rest of the common properties and functionalities can be extended from the other class. There are various types of inheritances and multilevel inheritance is one of them.
    In the Multilevel inheritance, a **derived class will inherit a base class** and as well as the derived class also act as the base class to other class. ... In this situation, each derived class inherit all the characteristics of its base classes.

3) Control Structure- Selection structure

    i)   If-else

```
if (condition) {
 // block of code to be executed if the condition is true
} else {
 // block of code to be executed if the condition is false
}
```

ii)   While
The while loop loops through a block of code as long as a specified condition is true:

```
while (condition) {
  // code block to be executed
}
```

iii)   If-else ladder

```
if (condition)
      statement 1;
else if (condition)
      statement 2;



else
      statement;
```

iv)   Switch case

```
switch(expression) {
case x:
  // code block
  break;
case y:
  // code block
  break;
default:
  // code block
}
```

Implementation

- ➤ The first thing we created was a frame for Username and Password through which the user will login to their personal account on the application.

- ➤ After entering the login credentials , a message saying "Hello" pops up which indicates that the user's login attempt was successful.

- ➤ Further on the console page , a welcome greeting is displayed and the user is then given the options for selection of their choice from the list of available branches.

- ➤ Here we have used an *if* condition which gives the error message when the user enter the wrong branch and the process further continues after the user enters the correct branch.

- ➤ Then the user is given a choice of selecting the year of their choice ranging from 1 to 4 .

- ➤ If the user enters the year less than 1 or greater than 4 which is not in the permissible range , the program will then give an error message for the same and will continue further after the user enters the correct year.

- ➤ According to the year selected , the respective semesters for that particular year will be displayed and now the user will have the choice of selecting the semester of their choice.

- ➤ We have also implemented a while loop for checking the semester value entered by the user. If the user enters a value less than 0 or greater than 8 , the program will then give an error message and will ask the user to enter the correct semester again for proceeding further.

- ➤ Further we have used switch case to display the semester wise content of the respective branch entered by user in the following way:

After the user enters the value in the branch variable it gets stored. And this value is further used as a parameter in the switch case structure. As we have given the branch variable as the parameter in switch , it will have the following four cases created namely:

```
switch(branch) {
    case : "EXTC"{
        //code to be executed
    }
    case : "COMPS"{
        //code to be executed
    }
    case : "IT"{
        //code to be executed
    }
    case : "Mechanical"{
        //code to be executed
```

```
            }

        }
```

➤ Inside each of the cases we have used the concept of *if-else* ladder for the conditions of each semester.

```
if (sem==1) {
        //subjects to be dispayed
}
  else if (sem ==2) {
        //subjects to be displayed
}
  else if (sem ==3) {
        //subjects to be displayed
}
  else if (sem ==4) {
        //subjects to be displayed
}
  else if (sem ==5) {
        //subjects to be displayed
}
  else if (sem ==6) {
        //subjects to be displayed
}
  else if (sem ==7) {
        //subjects to be displayed
}
  else {
        //subjects to be displayed
}
}
```

➤ In this program we have also used the concept of inheritance (to be specific 'Multilevel ' inheritance).

➤ Besides this , we have also used a few constructors to make our code easier. Constructors are the special methods that are called when the objects are created for the respective class.

➤ In the main function we have created the objects for the respective classes and we have used them to call the methods that we have created in those classes.

9

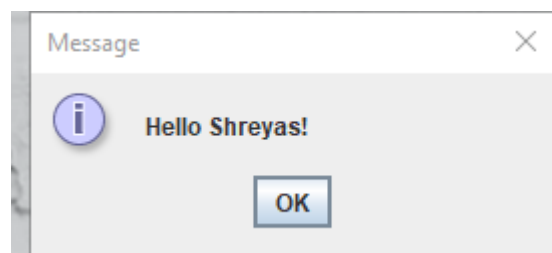➤ In this way , we have applied the concepts of programming taught to us.

CHAPTER 4

RESULTS

1) **Login Page:**

2) Welcome Message:

3)Console Output

```
Welcome !, here is our Course Categories:
EXTC
COMPS
IT
Mechanical
Select your branch:
CyberSecurity
Please enter from the given 4 options!
EXTC
1st Year
2nd Year
3rd Year
4th Year
Select year:
1
SEM 1
SEM 2
Select the Semester :
1
FEC 101 : Engineering Mathematics-1
FEC 102 : Engineering Physics-1
FEC 103 : Engineering Chemistry-1
FEC 104 : Engineering Mechanics
FEC 105 : Basic Electrical Engineering
Select the course of your choice (Just enter the subject code for e.g. for selecting maths enter 101):
101
```

Engineering MAthematics - 1
Syllabus

Teaching and Learning plan

Pre-requisite test

Course Objectives and Outcomes

Module 1: Complex Numbers
   Notes
   Quiz on Module 1
   Tutorial 1 Submission Link

Module 2: Hyperbolic Functions and Logarithm of Complex Numbers
   Notes
   Quiz on Module 2
    Tutorial 2 Submission Link

Module 3: Partial Differentiation
   Notes
   Quiz on Module 3
   Tutorial 3 Submission Link

Module 4: Applications of Partial Differentiation and Successive Differentiation
   Notes
   Quiz on Module 4
   Tutorial 4 Submission Link

Module 5: Matrices
   Notes
   Quiz on Module 5
   Tutorial 5 Submission Link

11

CHAPTER 5

CONCLUSION

So by using the various concepts taught to us of which we have applied majority of them in creating this project.

The concept of Demultiplexer can also be used in various applications and it has wide scope.

We can also create other management systems for Offices, Colleges, and other administrative places by using this concept and a little bit of modifications if required.

CHAPTER 6

REFERENCE

We have referred to the seminar on GUI concept and the Moodle notes provided to us.