

**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD**  
**B.Tech. in COMPUTER SCIENCE AND ENGINEERING**

**III YEAR LABS SYLLABUS (R18)**  
**Applicable From 2018-19 Admitted Batch**

**SOFTWARE ENGINEERING LAB**

**III Year B.Tech. CSE I-Sem**

**L T P C**  
**0 0 3 1.5**

**Prerequisites**

1. A course on "Programming for Problem Solving"

**Co-requisite**

1. A Course on "Software Engineering"

**Course Objectives**

1. To have hands on experience in developing a software project by using various software engineering principles and methods in each of the phases of software development.

**Course Outcomes**

1. Ability to translate end-user requirements into system and software requirements
2. Ability to generate a high-level design of the system from the software requirements
3. Will have experience and/or awareness of testing problems and will be able to develop a simple testing report

**List of Experiments**

**Do the following 8 exercises for any two projects given in the list of sample projects or any other projects:**

- 1) Development of problem statement.
- 2) Preparation of Software Requirement Specification Document, Design Documents and Testing Phase related documents.
- 3) Preparation of Software Configuration Management and Risk Management related documents.
- 4) Study and usage of any Design phase CASE tool
- 5) Performing the Design by using any Design phase CASE tools.
- 6) Develop test cases for unit testing and integration testing
- 7) Develop test cases for various white box and black box testing techniques.

**Sample Projects:**

1. Passport automation System
2. Book Bank
3. Online Exam Registration
4. Stock Maintenance System
5. Online course reservation system
6. E-ticketing
7. Software Personnel Management System
8. Credit Card Processing
9. E-book management System.
10. Recruitment system

**TEXT BOOKS:**

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6<sup>th</sup> edition, Mc Graw Hill International Edition.

2. Software Engineering- Sommerville, 7<sup>th</sup> edition, Pearson Education.
3. The unified modeling language user guide Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education.

## COMPUTER NETWORKS & WEB TECHNOLOGIES LAB

III Year B.Tech. CSE I-Sem

L T P C  
3 0 0 3

### Course Objectives

1. To understand the working principle of various communication protocols.
2. To understand the network simulator environment and visualize a network topology and observe its performance
3. To analyze the traffic flow and the contents of protocol frames

### Course Outcomes

1. Implement data link layer framing methods
2. Analyze error detection and error correction codes.
3. Implement and analyze routing and congestion issues in network design.
4. Implement Encoding and Decoding techniques used in presentation layer
5. To be able to work with different network tools

### List of Experiments

1. Implement the data link layer framing methods such as character, character-stuffing and bit stuffing.
2. Write a program to compute CRC code for the polynomials CRC-12, CRC-16 and CRC CCIP
3. Develop a simple data link layer that performs the flow control using the sliding window protocol, and loss recovery using the Go-Back-N mechanism.
4. Implement Dijkstra's algorithm to compute the shortest path through a network
5. Take an example subnet of hosts and obtain a broadcast tree for the subnet.
6. Implement distance vector routing algorithm for obtaining routing tables at each node.
7. Implement data encryption and data decryption
8. Write a program for congestion control using Leaky bucket algorithm.
9. Write a program for frame sorting technique used in buffers.
10. **Wireshark**
  - i. Packet Capture Using Wire shark
  - ii. Starting Wire shark
  - iii. Viewing Captured Traffic
  - iv. Analysis and Statistics & Filters.
11. How to run Nmap scan
12. Operating System Detection using Nmap
13. Do the following using NS2 Simulator
  - i. NS2 Simulator-Introduction
  - ii. Simulate to Find the Number of Packets Dropped
  - iii. Simulate to Find the Number of Packets Dropped by TCP/UDP
  - iv. Simulate to Find the Number of Packets Dropped due to Congestion
  - v. Simulate to Compare Data Rate & Throughput.
  - vi. Simulate to Plot Congestion for Different Source/Destination
  - vii. Simulate to Determine the Performance with respect to Transmission of Packets

### Web Technologies Experiments

1. Write a PHP script to print prime numbers between 1-50.
2. PHP script to
  - a. Find the length of a string.
  - b. Count no of words in a string.
  - c. Reverse a string.
  - d. Search for a specific string.

3. Write a PHP script to merge two arrays and sort them as numbers, in descending order.
4. Write a PHP script that reads data from one file and write into another file.
5. Develop static pages (using Only HTML) of an online book store. The pages should resemble: [www.amazon.com](http://www.amazon.com). The website should consist the following pages.
  - a) Home page
  - b) Registration and user Login
  - c) User Profile Page
  - d) Books catalog
  - e) Shopping Cart
  - f) Payment By credit card
  - g) Order Conformation
6. Validate the Registration, user login, user profile and payment by credit card pages using JavaScript.
7. Create and save an XML document on the server, which contains 10 users information. Write a program, which takes User Id as an input and returns the user details by taking the user information from the XML document.
8. Install TOMCAT web server. Convert the static web pages of assignments 2 into dynamic web pages using servlets and cookies. Hint: Users information (user id, password, credit card number) would be stored in web.xml. Each user should have a separate Shopping Cart.
9. Redo the previous task using JSP by converting the static web pages of assignments 2 into dynamic web pages. Create a database with user information and books information. The books catalogue should be dynamically loaded from the database. Follow the MVC architecture while doing the website.

**TEXT BOOKS:**

1. WEB TECHNOLOGIES: A Computer Science Perspective, Jeffrey C. Jackson, Pearson Education

**REFERENCES:**

1. Deitel H.M. and Deitel P.J., "Internet and World Wide Web How to program", Pearson International, 2012, 4th Edition.
2. J2EE: The complete Reference By James Keogh, McGraw-Hill
3. Bai and Ekedhi, The Web Warrior Guide to Web Programming, Thomson
4. Paul Dietel and Harvey Deitel, "Java How to Program", Prentice Hall of India, 8<sup>th</sup> Edition
5. Web technologies, Black Book, Dreamtech press.
6. Gopalan N.P. and Akilandeswari J., "Web Technology", Prentice Hall of India

## ADVANCED COMMUNICATION SKILLS LAB

III Year B.Tech. CSE I-Sem

L T P C  
3 0 0 3

### 1. INTRODUCTION:

The introduction of the Advanced Communication Skills Lab is considered essential at 3<sup>rd</sup> year level. At this stage, the students need to prepare themselves for their careers which may require them to listen to, read, speak and write in English both for their professional and interpersonal communication in the globalized context.

The proposed course should be a laboratory course to enable students to use 'good' English and perform the following:

- Gathering ideas and information to organize ideas relevantly and coherently.
- Engaging in debates.
- Participating in group discussions.
- Facing interviews.
- Writing project/research reports/technical reports.
- Making oral presentations.
- Writing formal letters.
- Transferring information from non-verbal to verbal texts and vice-versa.
- Taking part in social and professional communication.

### 2. OBJECTIVES:

This Lab focuses on using multi-media instruction for language development to meet the following targets:

1. To improve the students' fluency in English, through a well-developed vocabulary and enable them to listen to English spoken at normal conversational speed by educated English speakers and respond appropriately in different socio-cultural and professional contexts.
2. Further, they would be required to communicate their ideas relevantly and coherently in writing.
3. To prepare all the students for their placements.

### 3. SYLLABUS:

The following course content to conduct the activities is prescribed for the Advanced English Communication Skills (AECS) Lab:

1. **Activities on Fundamentals of Inter-personal Communication and Building Vocabulary** - Starting a conversation – responding appropriately and relevantly – using the right body language – Role Play in different situations & Discourse Skills- using visuals - Synonyms and antonyms, word roots, one-word substitutes, prefixes and suffixes, study of word origin, business vocabulary, analogy, idioms and phrases, collocations & usage of vocabulary.
2. **Activities on Reading Comprehension** –General Vs Local comprehension, reading for facts, guessing meanings from context, scanning, skimming, inferring meaning, critical reading& effective googling.
3. **Activities on Writing Skills** – Structure and presentation of different types of writing – *letter writing/Resume writing/ e-correspondence/Technical report writing/* – planning for writing – improving one's writing.
4. **Activities on Presentation Skills** – Oral presentations (individual and group) through JAM sessions/seminars/PPTs and written presentations through posters/projects/reports/ e-mails/assignments etc.
5. **Activities on Group Discussion and Interview Skills** – Dynamics of group discussion, intervention, summarizing, modulation of voice, body language, relevance, fluency and organization of ideas and rubrics for evaluation- Concept and process, pre-interview planning, opening strategies, answering strategies, interview through tele-conference & video-conference and Mock Interviews.

#### **4. MINIMUM REQUIREMENT:**

The Advanced English Communication Skills (AECS) Laboratory shall have the following infrastructural facilities to accommodate at least 35 students in the lab:

- Spacious room with appropriate acoustics.
- Round Tables with movable chairs
- Audio-visual aids
- LCD Projector
- Public Address system
- P – IV Processor, Hard Disk – 80 GB, RAM–512 MB Minimum, Speed – 2.8 GHZ
- T. V, a digital stereo & Camcorder
- Headphones of High quality

#### **5. SUGGESTED SOFTWARE:**

The software consisting of the prescribed topics elaborated above should be procured and used.

- Oxford Advanced Learner's Compass, 7<sup>th</sup> Edition
- DELTA's key to the Next Generation TOEFL Test: Advanced Skill Practice.
- Lingua TOEFL CBT Insider, by Dream tech
- TOEFL & GRE (KAPLAN, AARCO & BARRONS, USA, Cracking GRE by CLIFFS)

#### **TEXT BOOKS:**

1. Effective Technical Communication by M Asharaf Rizvi. McGraw Hill Education (India) Pvt. Ltd. 2<sup>nd</sup> Edition
2. Academic Writing: A Handbook for International Students by Stephen Bailey, Routledge, 5<sup>th</sup> Edition.

#### **REFERENCES:**

1. Learn Correct English – A Book of Grammar, Usage and Composition by Shiv K. Kumar and Hemalatha Nagarajan. Pearson 2007
2. Professional Communication by Aruna Koneru, McGraw Hill Education (India) Pvt. Ltd, 2016.
3. Technical Communication by Meenakshi Raman & Sangeeta Sharma, Oxford University Press 2009.
4. Technical Communication by Paul V. Anderson. 2007. Cengage Learning pvt. Ltd. New Delhi.
5. English Vocabulary in Use series, Cambridge University Press 2008.
6. Handbook for Technical Communication by David A. McMurrey & Joanne Buckley. 2012. Cengage Learning.
7. Communication Skills by Leena Sen, PHI Learning Pvt Ltd., New Delhi, 2009.
8. Job Hunting by Colm Downes, Cambridge University Press 2008.
9. English for Technical Communication for Engineering Students, Aysha Vishwamohan, Tata Mc Graw-Hill 2009.

## MACHINE LEARNING LAB

III Year B.Tech. CSE II-Sem

L T P C  
0 0 3 1.5

**Course Objective:** The objective of this lab is to get an overview of the various machine learning techniques and can able to demonstrate them using python.

**Course Outcomes:** After the completion of the course the student can able to:

1. understand complexity of Machine Learning algorithms and their limitations;
2. understand modern notions in data analysis-oriented computing;
3. be capable of confidently applying common Machine Learning algorithms in practice and implementing their own;
4. Be capable of performing experiments in Machine Learning using real-world data.

### List of Experiments

1. The probability that it is Friday and that a student is absent is 3 %. Since there are 5 school days in a week, the probability that it is Friday is 20 %. What is the probability that a student is absent given that today is Friday? Apply Baye's rule in python to get the result. (Ans: 15%)
2. Extract the data from database using python
3. Implement k-nearest neighbours classification using python
4. Given the following data, which specify classifications for nine combinations of VAR1 and VAR2 predict a classification for a case where VAR1=0.906 and VAR2=0.606, using the result of k-means clustering with 3 means (i.e., 3 centroids)

VAR1	VAR2	CLASS
1.713	1.586	0
0.180	1.786	1
0.353	1.240	1
0.940	1.566	0
1.486	0.759	1
1.266	1.106	0
1.540	0.419	1
0.459	1.799	1
0.773	0.186	1

5. The following training examples map descriptions of individuals onto high, medium and low credit-worthiness.

medium skiing design single twenties no -> highRisk  
high golf trading married forties yes -> lowRisk  
low speedway transport married thirties yes -> medRisk  
medium football banking single thirties yes -> lowRisk  
high flying media married fifties yes -> highRisk  
low football security single twenties no -> medRisk  
medium golf media single thirties yes -> medRisk  
medium golf transport married forties yes -> lowRisk  
high skiing banking single thirties yes -> highRisk  
low golf unemployed married forties yes -> highRisk

Input attributes are (from left to right) income, recreation, job, status, age-group, home-owner. Find the unconditional probability of `golf` and the conditional probability of `single` given `medRisk` in the dataset?

6. Implement linear regression using python.
7. Implement Naïve Bayes theorem to classify the English text
8. Implement an algorithm to demonstrate the significance of genetic algorithm
9. Implement the finite words classification system using Back-propagation algorithm

## COMPILER DESIGN LAB

III Year B.Tech. CSE II-Sem

L T P C  
0 0 3 1.5

### Prerequisites

1. A Course on “Objected Oriented Programming through Java”

### Co-requisites

1. A course on “Web Technologies”

### Course Objectives

1. To provide hands-on experience on web technologies
2. To develop client-server application using web technologies
3. To introduce server-side programming with Java servlets and JSP
4. To understand the various phases in the design of a compiler.
5. To understand the design of top-down and bottom-up parsers.
6. To understand syntax directed translation schemes.
7. To introduce lex and yacc tools.

### Course Outcomes

1. Design and develop interactive and dynamic web applications using HTML, CSS, JavaScript and XML
2. Apply client-server principles to develop scalable and enterprise web applications.
3. Ability to design, develop, and implement a compiler for any language.
4. Able to use lex and yacc tools for developing a scanner and a parser.
5. Able to design and implement LL and LR parsers.

### List of Experiments

#### Compiler Design Experiments

1. Write a LEX Program to scan reserved word & Identifiers of C Language
2. Implement Predictive Parsing algorithm
3. Write a C program to generate three address code.
4. Implement SLR(1) Parsing algorithm
5. Design LALR bottom up parser for the given language

```
<program> ::= <block>
<block> ::= { <variabledefinition> <slist> }
           | { <slist> }
<variabledefinition> ::= int <vardeflist> ;
<vardeflist> ::= <vardec> | <vardec> , <vardeflist>
<vardec> ::= <identifier> | <identifier> [ <constant> ]
<slist> ::= <statement> | <statement> ; <slist>
<statement> ::= <assignment> | <ifstatement> | <whilestatement>
              | <block> | <printstatement> | <empty>
<assignment> ::= <identifier> = <expression>
              | <identifier> [ <expression> ] = <expression>
<ifstatement> ::= if <bexpression> then <slist> else <slist> endif
              | if <bexpression> then <slist> endif
<whilestatement> ::= while <bexpression> do <slist> enddo
<printstatement> ::= print ( <expression> )
<expression> ::= <expression> <addingop> <term> | <term> | <addingop> <term>
<bexpression> ::= <expression> <relop> <expression>
<relop> ::= < | <= | == | >= | > | !=
```

<addingop> ::= + | -  
 <term> ::= <term> <multop> <factor> | <factor>  
 <multop> ::= \* | /  
 <factor> ::= <constant> | <identifier> | <identifier> [ <expression> ]  
           | ( <expression> )  
 <constant> ::= <digit> | <digit> <constant>  
 <identifier> ::= <identifier> <letterordigit> | <letter>  
 <letterordigit> ::= <letter> | <digit>  
 <letter> ::= a|b|c|d|e|f|g|h|i|j|k|l|m|n|o|p|q|r|s|t|u|v|w|x|y|z  
 <digit> ::= 0|1|2|3|4|5|6|7|8|9  
 <empty> has the obvious meaning

Comments (zero or more characters enclosed between the standard C/Java-style comment brackets /\*...\*/) can be inserted. The language has rudimentary support for 1-dimensional arrays. The declaration `int a[3]` declares an array of three elements, referenced as `a[0]`, `a[1]` and `a[2]`. Note also that you should worry about the scoping of names.

A simple program written in this language is:

```

{ int a[3],t1,t2;
  t1=2;
  a[0]=1; a[1]=2; a[t1]=3;
  t2=-(a[2]+t1*6)/(a[2]-t1);
  if t2>5 then
    print(t2);
  else {
    int t3;
    t3=99;
    t2=-25;
    print(-t1+t2*t3); /* this is a comment
                       on 2 lines */
  }
endif
}

```

## CONCURRENT PROGRAMMING LAB (Professional Elective - III)

III Year B.Tech. CSE II-Sem

L	T	P	C
0	0	2	1

### List of Experiments:

1. Design and implement Two-thread mutual exclusion algorithm (Peterson's Algorithm) using multithreaded programming.
2. Design and implement Filter Lock algorithm and check for deadlock-free and starvation-free conditions using multithreaded programming.
3. Design and implement Lamport's Bakery Algorithm and check for deadlock-free and starvation-free conditions using multithreaded programming.
4. Design and implement Lock-based concurrent FIFO queue data structure using multithreaded programming.
5. Design a consensus object using read-write registers by implementing a deadlock-free or starvation-free mutual exclusion lock. (Use CompareAndSet( ) Primitive).
6. Design and implement concurrent List queue data structure using multithreaded programming. (Use Atomic Primitives)
7. Design and implement concurrent Stack queue data structure using multithreaded programming. (Use Atomic Primitives)
8. Design and implement concurrent FIFO queue data structure using multithreaded programming. (Use Atomic Primitives)

## NETWORK PROGRAMMING LAB (Professional Elective - III)

III Year B.Tech. CSE II-Sem

L	T	P	C
0	0	2	1

### Course Objectives:

1. To understand inter process and inter-system communication
2. To understand socket programming in its entirety
3. To understand usage of TCP/UDP / Raw sockets
4. To understand how to build network applications

### Course Outcomes:

1. To write socket API based programs
2. To design and implement client-server applications using TCP and UDP sockets
3. To analyze network programs

### List of Experiments

1. Implement programs for Inter Process Communication using PIPE, Message Queue and Shared Memory.
2. Write a programme to create an integer variable using shared memory concept and increment the variable simultaneously by two processes. Use semaphores to avoid race conditions.
3. Design TCP iterative Client and server application to reverse the given input sentence
4. Design TCP iterative Client and server application to reverse the given input sentence
5. Design TCP client and server application to transfer file
6. Design a TCP concurrent server to convert a given text into upper case using multiplexing system call "select"
7. Design a TCP concurrent server to echo given set of sentences using poll functions
8. Design UDP Client and server application to reverse the given input sentence
9. Design UDP Client server to transfer a file
10. Design using poll client server application to multiplex TCP and UDP requests for converting a given text into upper case.
11. Design a RPC application to add and subtract a given pair of integers

### TEXT BOOKS:

1. UNIX Network Programming, by W. Richard Stevens, Bill Fenner, Andrew M. Rudoff, Pearson Education.
2. UNIX Network Programming, 1<sup>st</sup> Edition, - W. Richard Stevens. PHI.

## SCRIPTING LANGUAGES LAB (Professional Elective - III)

III Year B.Tech. CSE II-Sem

L	T	P	C
0	0	2	1

**Prerequisites:** Any High level programming language (C,C++)

### Course Objectives:

- To Understand the concepts of scripting languages for developing web based projects
- To understand the applications the of Ruby , TCL , Perl scripting languages

### Course Outcomes:

- Ability to understand the differences between Scripting languages and programming languages
- Able to gain some fluency programming in Ruby, Perl, TCL

### List of Experiments

1. Write a Ruby script to create a new string which is n copies of a given string where n is a non-negative integer
2. Write a Ruby script which accept the radius of a circle from the user and compute the parameter and area.
3. Write a Ruby script which accept the user's first and last name and print them in reverse order with a space between them
4. Write a Ruby script to accept a filename from the user print the extension of that
5. Write a Ruby script to find the greatest of three numbers
6. Write a Ruby script to print odd numbers from 10 to 1
7. Write a Ruby script to check two integers and return true if one of them is 20 otherwise return their sum
8. Write a Ruby script to check two temperatures and return true if one is less than 0 and the other is greater than 100
9. Write a Ruby script to print the elements of a given array
10. Write a Ruby program to retrieve the total marks where subject name and marks of a student stored in a hash
11. Write a TCL script to find the factorial of a number
12. Write a TCL script that multiplies the numbers from 1 to 10
13. Write a TCL script for Sorting a list using a comparison function
14. Write a TCL script to (i)create a list (ii )append elements to the list (iii)Traverse the list (iv)Concatenate the list
15. Write a TCL script to comparing the file modified times.
16. Write a TCL script to Copy a file and translate to native format.
17. a) Write a Perl script to find the largest number among three numbers.  
b) Write a Perl script to print the multiplication tables from 1-10 using subroutines.
18. Write a Perl program to implement the following list of manipulating functions  
a)Shift  
b)Unshift  
c)Push
19. a) Write a Perl script to substitute a word, with another word in a string.  
b) Write a Perl script to validate IP address and email address.
20. Write a Perl script to print the file in reverse order using command line arguments

## MOBILE APPLICATION DEVELOPMENT LAB (Professional Elective - III)

III Year B.Tech. CSE II-Sem

L	T	P	C
0	0	2	1

Prerequisites: --- NIL---

### Course Objectives:

1. To learn how to develop Applications in android environment.
2. To learn how to develop user interface applications.
3. To learn how to develop URL related applications.

### Course Outcomes:

1. Student understands the working of Android OS Practically.
2. Student will be able to develop user interfaces.
3. Student will be able to develop, deploy and maintain the Android Applications.

### List of Experiments

1. Create an Android application that shows Hello + name of the user and run it on an emulator.  
(b) Create an application that takes the name from a text box and shows hello message along with the name entered in text box, when the user clicks the OK button.
2. Create a screen that has input boxes for User Name, Password, Address, Gender (radio buttons for male and female), Age (numeric), Date of Birth (Date Picket), State (Spinner) and a Submit button. On clicking the submit button, print all the data below the Submit Button. Use (a) Linear Layout (b) Relative Layout and (c) Grid Layout or Table Layout.
3. Develop an application that shows names as a list and on selecting a name it should show the details of the candidate on the next screen with a "Back" button. If the screen is rotated to landscape mode (width greater than height), then the screen should show list on left fragment and details on right fragment instead of second screen with back button. Use Fragment transactions and Rotation event listener.
4. Develop an application that uses a menu with 3 options for dialing a number, opening a website and to send an SMS. On selecting an option, the appropriate action should be invoked using intents.
5. Develop an application that inserts some notifications into Notification area and whenever a notification is inserted, it should show a toast with details of the notification.
6. Create an application that uses a text file to store user names and passwords (tab separated fields and one record per line). When the user submits a login name and password through a screen, the details should be verified with the text file data and if they match, show a dialog saying that login is successful. Otherwise, show the dialog with Login Failed message.
7. Create a user registration application that stores the user details in a database table.
8. Create a database and a user table where the details of login names and passwords are stored. Insert some names and passwords initially. Now the login details entered by the user should be verified with the database and an appropriate dialog should be shown to the user.
9. Create an admin application for the user table, which shows all records as a list and the admin can select any record for edit or modify. The results should be reflected in the table.
10. Develop an application that shows all contacts of the phone along with details like name, phone number, mobile number etc.
11. Create an application that saves user information like name, age, gender etc. in shared preference and retrieves them when the program restarts.
12. Create an alarm that rings every Sunday at 8:00 AM. Modify it to use a time picker to set alarm time.
13. Create an application that shows the given URL (from a text field) in a browser.

## SOFTWARE TESTING METHODOLOGIES LAB (Professional Elective - III)

III Year B.Tech. CSE II-Sem

L	T	P	C
0	0	2	1

**Prerequisites:** A basic knowledge of programming.

### Course Objectives

1. To provide knowledge of Software Testing Methods.
2. To develop skills in software test automation and management using latest tools.

### Course Outcome

1. Design and develop the best test strategies in accordance to the development model.

### List of Experiments:

1. Recording in context sensitive mode and analog mode
2. GUI checkpoint for single property
3. GUI checkpoint for single object/window
4. GUI checkpoint for multiple objects
5. a) Bitmap checkpoint for object/window  
a) Bitmap checkpoint for screen area
6. Database checkpoint for Default check
7. Database checkpoint for custom check
8. Database checkpoint for runtime record check
9. a) Data driven test for dynamic test data submission  
b) Data driven test through flat files  
c) Data driven test through front grids  
d) Data driven test through excel test
10. a) Batch testing without parameter passing  
b) Batch testing with parameter passing
11. Data driven batch
12. Silent mode test execution without any interruption
13. Test case for calculator in windows application