



# Imparting Tutorials using online E-Commerce Application and Virtualization concept of Standalone PC Environment

<sup>1</sup>N.Rakesh, <sup>2</sup>S.K.Srivatsa

<sup>1</sup>Research Scholar, PRIST University, Tanjore, Tamil Nadu, India

<sup>2</sup>Senior Professor, St.Joseph College of Engineering, Chennai, Tamil Nadu, India

E-mail: [sai\\_rakesh2005@rediffmail.com](mailto:sai_rakesh2005@rediffmail.com), [profsks@rediffmail.com](mailto:profsks@rediffmail.com)

## ABSTRACT

We have acquired knowledge about virtualization through various sources and tutorial materials, which overcomes network resources investment for communication purpose. Accordingly using this virtualization communication perception, people in corporate world have developed Desktop Virtualization sharing and learning respectively online from their desktop system. Our research area also concentrates on such virtual communication in general. Having this background, in this paper we have tried to share and implement the idea about teaching methodology going online, learning from their standalone computers in academic environment which is a new thought. This concept which is a synchronization application, that will change the standards of education in academics, here we have used Java programming language and networking protocols like RMI (Remote Method Invocation) concept while designing various modules for implementing the same. It is achieved by providing effective environment between the Students and the Professor, which is used to challenge the standards of quality education and get rid of drawbacks of customary learning. Such standalone system environment helps in providing various application services. This virtualized tutorial teaching aid concept provides features such as like establishment of virtual standalone environment (here the objects are transmitted from Professor to Students), The File & Clipboard Transfer (helps data transfer between Students and Professor), VoIP, which integrates both voice and data in single communication channel facilitates (Half duplex voice dialog between the Professor and Students).

**Keywords:** *HTTP, NAT, RMI, SSL, VoIP*

## 1. INTRODUCTION

Today students expect some thing new in teaching aid than a formal interaction. With customary method which is constrained to four walls. This online tutorial aid will assist the instructors and students interaction from different places like a collaboration scenario. Here the students can see the standalone PC of the Professor in his PC.

This online tutorial system provides the following features:–

- **Establish Virtual Standalone environment** – The screen objects are transferred from instructor to students.
- **The File & Clipboard Transfer** – Enables file and clipboard transfer between Instructor and Students.
- **The Voice over Internet Protocol** – It helps in voice communication between the Professor and Students.

- **The Personalization** – It assist using a casual, conversational style for spoken and written content.
- **The Multimodality** – Orientation training content using multiple sensory channels (read it, hear it, view it, do it, transfer it).
- **Lean-Forward Learning** – The live demonstrations help in easy learning.
- **The Collaboration** –It provides technologies that allow Students to connect with Professor.
- **Moderation** – Balancing control between the e-learning system and the student to achieve optimal learning.[9],[10],[13],[14]-[15].

## 1.2 LITERATURE SURVEY:

**A) Educational Changes:** Education is going through major changes at a rapid pace. The proportion of non - traditional students is rapidly increasing. In the academic year 2006, 25.5% of all students in public 4-year institutions worked full-time, and 9.2% were single parents (ICES, 2006). Both categories are examples of students in higher education who may not



have the opportunity to be on campus at specific times. So technological change was ubiquitous. [19]-[20]

### B) Technological Changes:

The rapid technological changes and the changing student demographics point to a need to revisit the concept of office hours as a means to give assistance to students outside the classroom shared with other students. With the exception of shared applications, none of these provide an efficient means of giving real-time, graphics-intensive individualized assistance. [21]-[23]

**C) Remote Desktop Technologies:** Using Remote Desktop Technologies, instructors can offer assistance to student's on their own computer, using their own applications, and using their own partially completed files. It does not require the Instructor to be present at the Student's desk for help. The Instructor while seated on his desk can provide Student's with his valuable assistance. [1]

**1.4 Motivation:** Due to speedy advancement of technology, Virtual standalone E-Learning is now a reality and is in its fast development stage as it has many positive features over traditional classrooms as discussed above. There are many advantages in teaching and learning using Virtual standalone E-Learning. It is quick and easy and moreover it provides electronic learning process which is no doubt efficient thus providing effective learning. [21]

**1.4.1. Statement of the problem:** The Virtual standalone E-Learning can be used in many ways. Before looking at the system first we specify what the problem is and then we define the problem in the following sections.

**1.5.1 Problem Definition:** Virtual standalone E-Learning application utilizes remote desktop technologies which offer an opportunity to fill this gap. It forms a synchronization tool which changes the standard of education.

**1.5.2 Problem Explanation:** Here various applications of E-Learning are either based on audio video conferencing or text based i.e. E-Mail, PDF, and Document sharing. Such applications are not providing real-time, graphics-intensive individualized assistance with practical approach. There are few other parameters which have to be considered:

- **The Communications** – Most of the time communication will be through emails or text oriented chatting where students might miss the sound of an instructor.
- **It may be expensive** – Cameras and editing equipment can be expensive.
- **It require a great deal of planning and preparation** – To be effective, the camera crews and the instructor must practice and become a team.

- **It require technical support team** –In view of the complexity of video recording, mixing, and transmission, a technical support team is required.

**1.6 Objective of the study:** Virtual standalone tutorial system allows Professor to capture and transmit the screen over the network, Student to receive and view the desktop of the Instructor.

It provides features like:

- **The Communication through RMI** – Remote Method Invocation (RMI) provides communication between the Instructor and Student.
- **The Multiple User Interfaces** – It allows more than one student to be connected to the Instructor at a time.
- **The Multihomed Server** – It creates custom sockets for each Student connected to the Instructor.
- **The Authentication** – By authenticating, secured connection is established.
- **The Encryption** – It does the encryption of data for security.
- **The File and Clipboard Transfer** – Enables both Instructor and Student to transfer files and clipboard contents.
- **The Color and Image Quality** – Allows Student to modify the color and quality of the image
- **The Data Compression** – Allows compression of the data for transmission over the network.
- **The Screen Functions** – It allows screen to be viewed in full or normal mode.
- **The Connection Details and Remote Host Properties** –

It allows viewing connection details and remote host properties. Virtual standalone tutorial system is structured to allow students to progress at their own pace, using a Read It, Hear It, View It, Do It, Transfer It approach as shown below in the table 1.

That includes like Read It, Hear It, View It, Do It allows Student in effective and efficient way of learning which is otherwise not possible in traditional way of learning.

**Table 1:** Objective of the system

1	Read It	The data transferred from Professor system can be read on student system
2	Hear It	Here the students gets the answer replied by Professor
3	View It	Program execution at Professor system can be view by student
4	Do It	After that, student can start execute in his system
5	Transfer It	This feature allows both Professor and Student to transfer the files.

- Student will request for connection with server for communication, receive and playback the screen.
- Educational institutions can use this system for effective teaching.[1]-[29]

## 2.2. Outline of Functional Requirements:

The system provides its users with the ability to use the application either by using tray icon or by using the GUI based window. The front end provides information such as checking whether the following features are enabled: server, SSL, multihomed, data compression, file and clipboard transfer, voice communication, screen functions. It also provides connection information, server information and host properties.[31]-[33]

**2.3. Overview of Data Requirements:** The system requires Student to login with the user name, password, IP address, SSL encryption as provided by the Professor. The result of the system is capturing and transmission of screen by the Professor and receive and playback by the Student along with voice communication.[39]

**2.4.General Constraints, Assumptions Dependencies, Guidelines:** This Virtual standalone tutorial system is entirely based on Remote Method Invocation, SSL and requires capturing, transmitting, receive and playback of screen and voice so the system must have enough memory to process the screen and voice as per the specification indicated by each and every Student. In order for the effective playback and communication with multiple clients the

bandwidth and network should be configured accordingly.[26][34]-[35]

## 3. LAYOUT OF THE VIRTUAL STANDALONE TUTORIAL SYSTEM

The figure below shows the overview of our project and is explained as below:

- The faculty captures the screen and transmits it over the network using RMI.
- Both the faculty and student can transfers the file and clipboard contents.
- The student receives the screen and plays back the screen.
- The student can perform various operations with the screen.
- The student can use keyboard event (F11) to maximize or minimize the screen.

**1.7 Restrictions and future scope of the study:** The limitations of this system are:

- **Memory constraints** – As the screen is captured and transmitted in terms of frames it consumes more memory of both the Instructor and Student machines.
- **Bandwidth constraints** - Slower computer response time as all communication with our system takes place over the campus network, resulting in some response lag time.
- **Security constraints** - This system is not configured to pass through firewalls and NAT clients.
- The enhancements of this system are:
- To make it accessible through firewalls by hosting a servlet on HTTP server by making use of HTTP tunnel.
- Embedding Video transmission along with Voice transmission in application.
- Implementing Exam oriented application transfer where students can access it dynamically and submit to the instructor after completion of exam.[1],[27],[36]-[39]

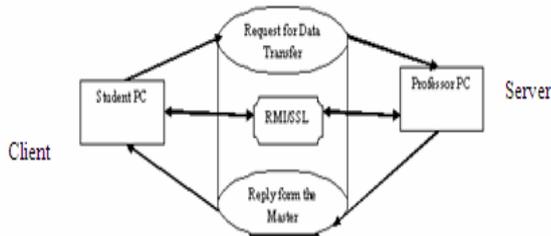
## 2. GENERAL DESCRIPTION

Under this topic we discuss the overall description of the system and users who uses the system along with their characteristics.

**2.1 Users and their Characteristics:** The users of this system are students, educational institutions, instructors, faculties, etc. so all these users have the following characteristics:

- Faculty will start the server, capture and transmit the screen.

- Both faculty and Student can interact with each other using voice communication.[21]-[39]



**Figure 1:** Layout of Virtual Standalone Tutorial System

**3.1. About Technology:** The technology we use for designing our system is Core Java, RemoteMethod Invocation, SSL and Swings. Technologies used to implement Virtual Standalone Tutorial System are explained below:

**Core JAVA:** Java was conceived by James Gosling, Patrick Naughton, Chris Warth, Ed Frank, and Mike Sheidan at Sun Microsystems, Inc. in 1991. This language was initially called “Oak” but was renamed “Java” in 1995. It is related to C++, which is a direct descendent of C. Much of the character of Java is inherited from these two languages. This system is built using Java because Java has many inbuilt features.

Features of Java are:

- Simple** – Since Java is primarily a derivative of C++ which most programmers are conversant with, it implies that Java has a familiar feel rendering it easy to use.
- Object-Oriented** – Even though Java has the look and feel of C++, it is a wholly independent language which has been designed to be object-oriented from the ground up.
- Distributed** – Commonly used Internet protocols such as HTTP and FTP as well as calls for network access are built into Java.
- Interpreted** – When Java code is compiled, the compiler outputs the Java Bytecode which is an executable for the Java Virtual Machine. The Java Virtual Machine does not exist physically but is the specification for a hypothetical processor that can run Java code. The bytecode is then run through a Java interpreter on any given platform that has the interpreter ported to it.
- Robust** – Java compels the programmer to be thorough. It carries out type checking at both compile and runtime making sure that every data structure has been clearly defined and typed. Java manages memory automatically by using an automatic garbage collector. The garbage collector runs as a low priority thread in the background keeping track of all objects and references to those objects in a Java program.
- Secure** – The Java language has built-in capabilities to ensure that violations of security do not

occur. One of the key design principles of Java is safety and security.

**g) Architecture-Neutral** – The Java compiler compiles source code to a stage which is intermediate between source and native machine code. This intermediate stage is known as the bytecode, which is neutral. The bytecode conforms to the specification of a hypothetical machine called the Java Virtual Machine and can be efficiently converted into native code for a particular processor.

**h) Portable** – By porting an interpreter for the Java Virtual Machine to any computer hardware/operating system, one is assured that all code compiled for it will run on that system.

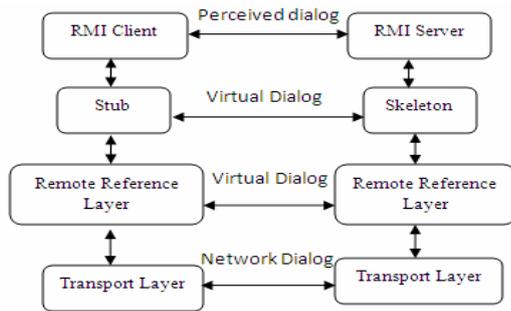
**i) High Performance** – The Java language supports many high-performance features such as multithreading, just-in-time compiling, and native code usage.

**j) Multithreaded** – Java has employed multithreading to help overcome the performance problems suffered by interpreted code as compared to native code.

**k) Dynamic language** – By connecting to the Internet, a user immediately has access to thousands of programs and other computers..

1. **Remote Method Invocation-** RMI applications often comprise two separate programs, a server and a client. A typical server program creates some remote objects, makes references to these objects accessible, and waits for clients to invoke methods on these objects. RMI provides the mechanism by which the server and the client communicate and pass information back and forth. Distributed object applications need to do the following:

- Locate remote objects** – Applications can use various mechanisms to obtain references to remote objects. For example, an application can register its remote objects with RMI's simple naming facility, the RMI registry.
- Communicate with remote objects** – Details of communication between remote objects are handled by RMI. It enables the Instructor's desktop to be transferred to the Student.
- Load class definitions for objects that are passed around** – Since RMI enables objects to be passed back and forth, it provides mechanisms for loading an object's class definitions as well as for transmitting an object's data. The following illustration depicts RMI architecture:



**Figure 2:** Architecture of RMI

When a remote method is invoked from a client, generally we will be passing parameters. The remote method may be returning some value. Parameters are to be transmitted to the server and return value has to be transmitted back to the client. Primitive data type parameters and return value can be transmitted byte by byte. However, if the parameters and return values are complex objects there will be problems. The server requires the parameter objects complete tree structure of objects. Similarly, the complete tree structure of the returned object has to be transmitted to the client. Thus for RMI to work, both the client and server should have full featured object serialization system. [2],[6],[7],[8],[9],[10],[25]-[31]

### 3.2. Modules of the Virtual standalone tutorial system

- **Module 1: Authentication and Secure connection** – In this module the instructor and the student are authenticated under Secure Socket Layer (SSL) implementation provided through Remote Method Invocation (RMI) technology.

- **Module 2: Screen Transfer** – In this module the instructor's desktop is captured and transferred in terms of frames over the network. This is the main module or the feature of the project. The student receives and plays back the captured screen according to his system configuration. Here the student can perform the functions such as data compression, modify the image and color quality.

- **Module 3: File and Clipboard Transfer** – In this module the faculty and the student both can transfer the files and the clipboard contents to each other and save them at the specified location on the system.

- **Module 4: Voice Communication** – In this module the faculty's voice is broadcasted to all the students. The students can put their queries to the instructor by requesting for communication. The faculty has the authority to accept or reject students request as well as disconnect the student in order to communicate with the other students. [1],[11],[12],[14],[32],[34]-[37]

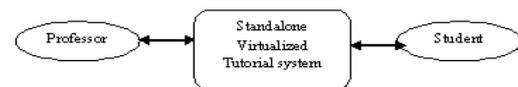
**3.3. Data Flow Diagrams:** A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system. DFD is the best option to represent a system diagrammatically.

There are only four symbols:

1. **Squares** representing external entities, which are sources or destinations of data.
2. **Rounded** rectangles representing processes, which take data as input, do something to it, and output it.
3. **Arrows** representing the data flows, which can either, be electronic data or physical items.
4. **Cylinder** representing data base. [3],[4],[16],[24]-[40]

**3.4.1. Context Level DFD or Level 0 DFD:** Context diagrams are also Called Level 0 DFD. It describes the very base of our project. Context diagram of Virtual standalone tutorial system contains following information:

- a) It contains one process called Virtual standalone tutorial system which is used to connect instructor and student.
- b) It contains two external entities Instructor and Student which work together with the process.

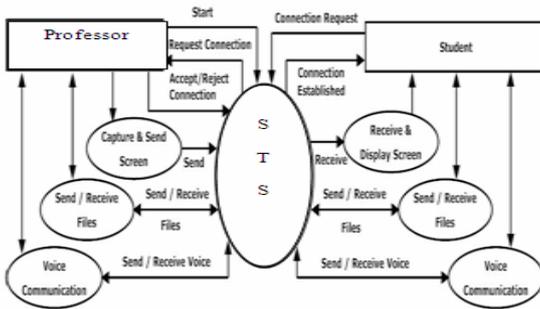


**Figure 3 :** Context Level DFD

**3.4.2 Level 1 DFD:** In this diagram we will have the detail explanation of the context diagram of the system Authentication and Secure connectin. Our system contains seven processes which are:

- **Capture and Send Screen** – The Instructor captures the screen and sends it to the Student over the network using RMI.
- **Receive and Display Screen** – The Student receives the transmitted screen and displays the screen on his machine.
- **Send and Receive Files** – Both the Instructor and the Student can send to and receive files from each other.
- **Voice Communication** – The Instructor and Student can communicate with each other using voice communication, they can send to receive audio messages.

<http://www.cisjournal.org>



Note : STS-Standalone Tutorial System

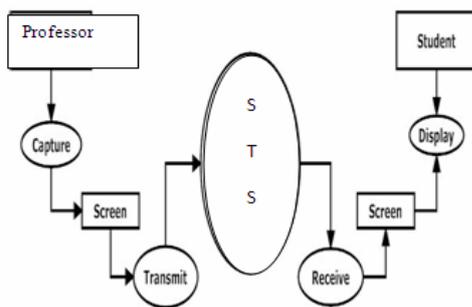
**Figure 4:** Level 1 DFD

**3.4.3 Level 2 DFD:** In this diagram we will have the detail explanation of the context diagram of the system of screen transfer.

**Screen Transfer**

It consists of four processes which are:

- **Capture screen** – The faculty captures the screen and send it to the Student. This is done through RMI.
- **Transmit screen** – The faculty after capturing the screen transmits it over the network.
- **Receive screen** – The Student receives the screen transmitted by the Instructor and the Student can view what the Instructor does.
- **Display screen** –The Student after receiving the screen performs playback or displays the screen on his machine.



Note: STS- Standalone Tutorial System

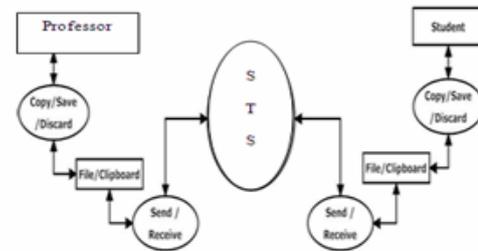
**Figure 5:** Level 2 DFD for Screen Transfer

**3.4.4. File and Clipboard Transfer:** Here both the Instructor and the Student can send and receive Files and clipboard contents to each other they can also save the files to the specified location. This option allows the Instructor and the Student to interact with each other. It consists of two processes which are:

- **To Copy/Save/Discard** – To send a file or clipboard contents the Instructor and Student can copy

the file and send it to each other and on reception of receiving the file they can save the file to the specified location or discard it.

- **Send/Receive** – Both the Instructor and student have the privilege to send and receive files from each other. Therefore, with this feature the Instructor and the Student can interact with each other any images, textual matter, audio data can be sent.

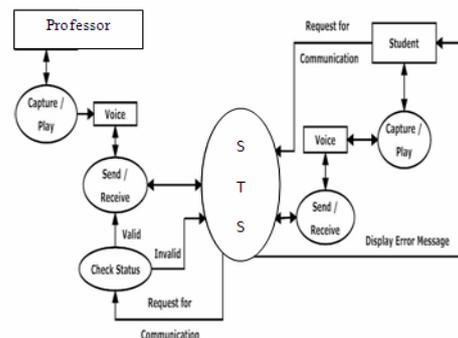


Note: STS – Standalone Tutorial System

**Figure 6:** Level 2 DFD for File and Clipboard Transfer

**3.4.5. Voice Communication:** It consists of three processes which are:

- **Capture/Play voice** – Both Faculty and Student can capture the voice they receive and perform the playback.
- **Send/Receive** – Both Faculty and Student can send and receive the voice of each other.
- **Check status** – In order to communicate with the Faculty the Student sends a request, this process checks the status whether the Faculty is busy communicating with other Student and if yes the status is returned as invalid and an error message is displayed on the Students machine.



Note: STS- Standalone Tutorial System

**Figure 7:** Level 2 DFD for Voice Communication

**4. THE USE CASE VIEW**

A use case diagram presents a collection of use cases and actors and is typically used to specify or characterize the functionality and behavior of a whole application system interacting with one or more

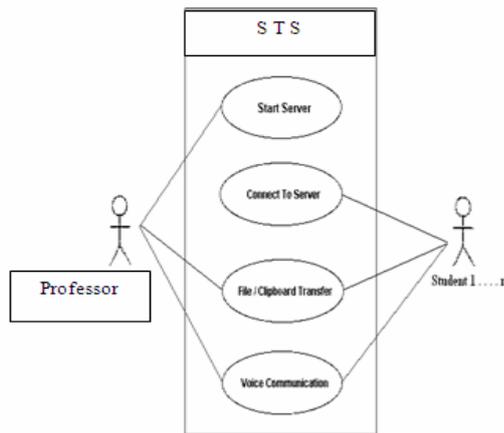
external actors. First we list and describe the different use cases and the actors involved in the system.

The use cases are:

- **Start Server:** This does the configuration and starting of the Instructor.
- **Connect to Server:** Using this, the Student connects to the Instructor.
- **File and Clipboard Transfer:** This option enables sending and receiving of the files.
- **Voice Communication:** This enables voice based interaction between the Student and the Instructor.

The actors of the system are described below:

- **Professor** – The Professor can start the server; perform the transmission of file and clipboard contents and voice communication.
- **Student** – The Student can connect to server; perform the transmission of file and clipboard contents and voice communication.

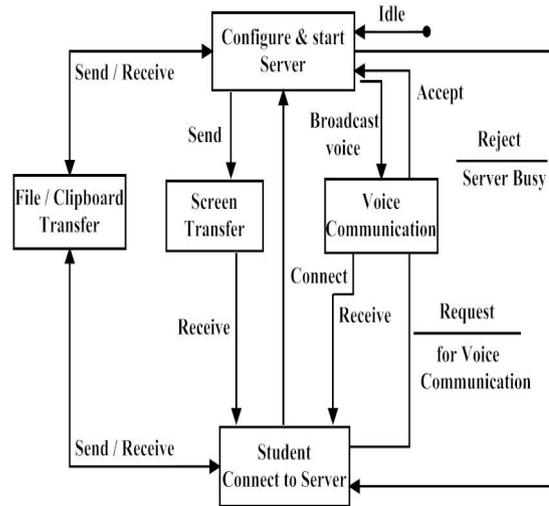


**Figure 8:** Use-Case Diagram

## 5. STATE TRANSITION DIAGRAM

A State is an observable model of behaviour. Each state represents a mode of behaviour of the system. A State transition diagram indicates how the system moves from one state to another. It starts from the idle state which is shown using dot.

The identified system states and its transition for Virtual standalone tutorial system are illustrated below in Figure 5.a. Transitions are those which take the system from one state to another state by any events triggered.



**Figure 9:** State Transition diagram

The description of the states in the State Transition diagram is as follows:

- **To Configure and Start Server** – This state is the initial state where the Instructor configure and starts the system
- **To Connect to Server** – The student connects to Server at this state.
- **The Screen transfer** – In this state the Instructor transmits the screen and student receives and performs the playback.
- **File and Clipboard Transfer** – In this state both the Instructor and Student can send and receive files from each other.
- **The Voice Communication** – In this state, the Server’s voice is broadcasted to Students. The student requests for communication with Instructor, the Instructor can either accept or reject the request and it will be rejected automatically if Instructor is communicating with other student.

[1],[3],[4],[5],[9],[10],[14],[15],[16],[38],[40],[41],[42],[43]-[45]

## 6. IMPLEMENTATION

Implementation is the process of converting a new system design into an operational one. It is the key stage in achieving a successful new system. It must therefore be carefully planned and controlled. The implementation of a system is done after the development effort is completed. The system is fully implemented in Java.

Steps for implementation:

1. Installation of hardware and software utilities.
2. Testing the developed system with sampled data.



<http://www.cisjournal.org>

3. Detection and correction of errors.
4. Data updating.

Here we present the implementation details for all modules that are:

Module 1: Authentication and Secure connection

Module 2: Screen Transfer

Module 3: File and Clipboard Transfer

Module 4: Voice Communication

### 6.1 Authentication and Secure Connection

**Implementation:** In this module, the instructor and the students are authenticated, and a secure connection is established between them using Secure Socket Layer (SSL). The variables used in this class to set the configurations are as follows:

#### Code Fragment:-

```
public class Config {
public static boolean GUI_disabled = false;
public static boolean Systray_disabled = false;
public static void loadConfiguration() ;
public static void storeConfiguration () ;
}
```

**6.2 Screen Transfer Implementation:** In this module, the instructor's desktop is captured and transferred in terms of frames over the network. The student receives and plays back the captured screen according to his system configuration. Here the student can perform the functions such as data compression, modify the image quality and color quality of the screen. The classes and methods that are used to implement this module i.e. Screen Transfer are as follows:

#### Code Fragment:-

```
public class Server extends Thread {
public static void updateData(byte[] data, int index);
public static byte[] updateData(int index);
public static void updateOptions(Object data, int index);
public static int SendClipboardFileList();
public static void setUploadingFolder ();
}
```

### 6.3. The File and Clipboard Transfer

**Implementation:** In this module, the instructor and the student can transfer the files and the clipboard contents to each other and save them at the specified location on the system. The classes and methods that are used to implement this module i.e. File and Clipboard Transfer are as follows:

#### Code Fragment:-

```
public static void SendFile(byte[] filedata, String
fileName, int index) {
fileName = uploadingFolder + fileName;
new File(fileName).getParent().mkdirs();
File file = new File(fileName);
BufferedOutputStream output = new
BufferedOutputStream(new
FileOutputStream(file));
output.write(filedata, 0, filedata.length);
output.flush();
output.close();

connectionsInfos.get(index).incReceivedData(fi
ledata.length);
}
```

### 6.4. Voice Communication Implementation:

In this module, the instructor's voice is broadcasted to all the students. The students can put their queries to the instructor by requesting for communication. The instructor has the authority to disconnect the student in order to communicate with the other students. The classes and methods that are used to implement this module i.e. Voice Communication are as follows:

When the server is started, the instructor's voice is broadcasted to the students. Here there are four Remote Methods:

#### Code Fragment:-

```
public boolean acceptRequest(InetAddress
inetAddress){
if(!SERVER_ENGAGED)
{
disInet = inetAddress;
DISCONNECT = false;
dialogGUI(inetAddress);
for(;(ARDialog.isVisible()&&flag));
ARDialog.dispose();
}
return ACCEPT_REQUEST;
}
```

## 7. TESTING

This section gives the outline of all the testing methods that are carried out to get a bug free system. Quality can be achieved by testing the product using different techniques at different phases of the project development.

**7.1. Testing done in the Project:** Here test cases based on different levels of testing are developed to show how the system responds to the test carried on the system and it even checks the exceptions that may arise.

## 8. EXECUTION OF THE PROJECT

**a) Stage 1:** This snapshot shows the idle Server GUI which is displayed after running the System on server side. Through this GUI the Faculty Starts the Server.

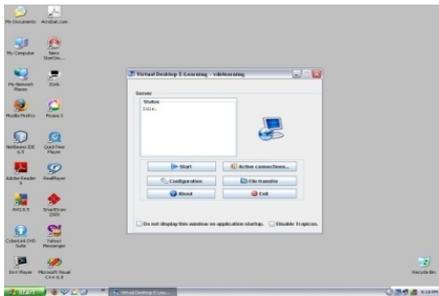


Figure 10.a: Snapshot showing idle Server GUI

**b) Stage 2:** This GUI allows the Instructor to give the Configuration details by entering IP address, Port, Password, etc. If SSL is enabled, secured connection is established. It also provides other option such as multihomed (auto detect), manual.

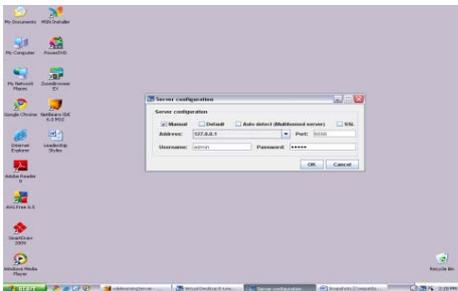


Figure 10.b : Snapshot of Server Configuration

**c) Stage 3:** This snapshot shows that the Server is running after the Instructor has provided the correct configuration to start the server.

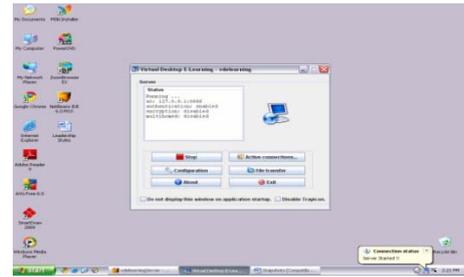


Figure 10.c: Snapshot of running system

**d) Stage 4:** This snapshot shows the Student's main screen, through which the Student has options to connect to the Instructor to start the whole system.

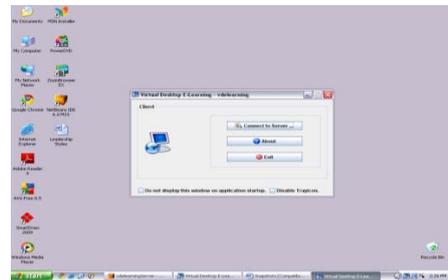


Figure 10.d: Snapshot of Student's main screen

**e) Stage 5:** In this GUI the Student provides the configuration details to connect to Instructor.

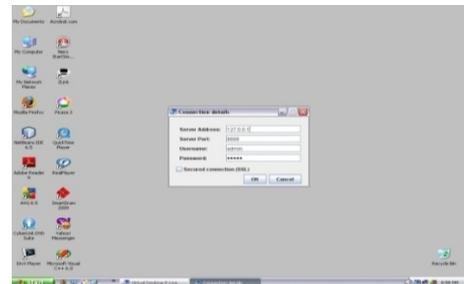


Figure 10.e: Snapshot of Students configuration details

**f) Stage 6:** This GUI is the student's GUI where the instructor's desktop is displayed.

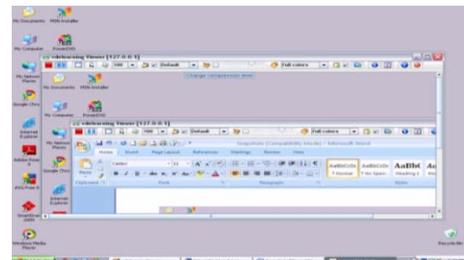
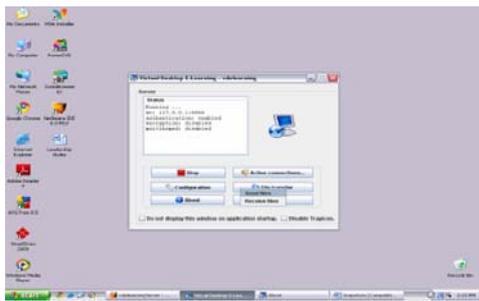


Figure 10.f: Snapshot of Viewers GUI

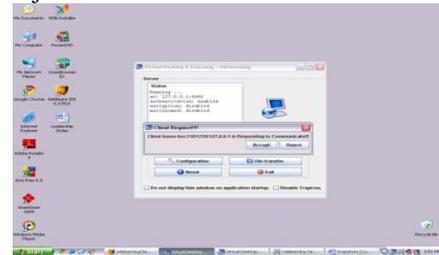
**g) Stage 7:** This GUI displays File Transfer which has two options: send files and receive files at the Instructor's side.

<http://www.cisjournal.org>



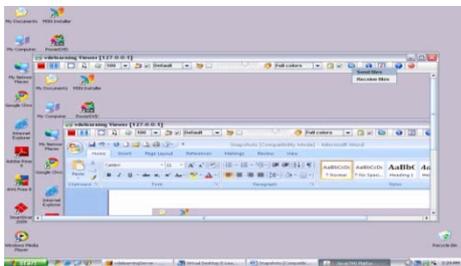
**Figure 10.g:** Snapshot of Instructors File transfer

reject the request.



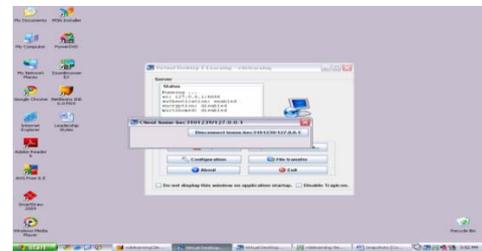
**Figure 10.k:** Snapshot of Students request for communication

**h) Stage 8:** This GUI displays File Transfer which has two options: send files and receive files at the instructor's side.



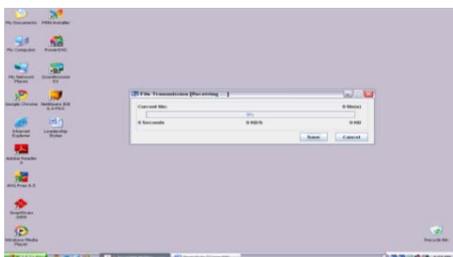
**Figure 10.h:** Snapshot of Students File Transfer

**l) Stage 12:** This GUI is displayed after the instructor accepts the request, after which the instructor can disconnect that student.



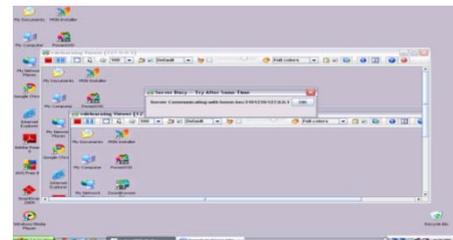
**Figure 10.l:** Snapshot of Student disconnected GUI

**i) Stage 9:** This GUI shows File Transfer on receiving of file, where user has the option to save the file.



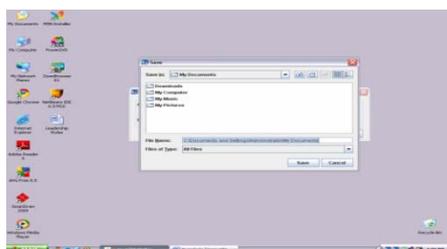
**Figure 10.i :** Snapshot of file received

**m) Stage 13:** This GUI is displayed if another student requests for voice communication with the instructor, while the instructor is communicating with another the student.



**Figure 10.m:** Snapshot of Server Busy Message

**j) Stage 10:** This GUI shows the Save Dialog, which allows the user to select the location where the file can be saved.



**Figure 10.j:** Snapshot to save file in specified location

**n) Stage 14:** This GUI displays the student's Tray Icon.



**Figure 10.n.:** Snapshot of Student's Tray Icon

**k) Stage 11:** This GUI shows the students request for communication the instructor can accept or

**o) Stage 15:** This GUI displays the Instructor's Tray Icon.

<http://www.cisjournal.org>

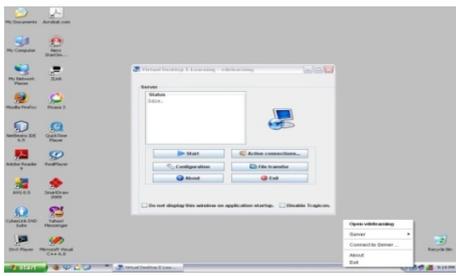


Figure 10.o: Snapshot of Instructor's Tray Icon

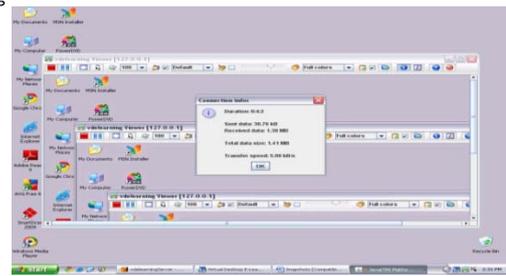


Figure 10.s.: Snapshot of Connection Info's

p) **Stage 16:** This GUI shows the Active Connections which gives details like how many students are connected to the Server.

[2],[6],[7][8],[26],[27],[28],[31],[32],[33],[34],[38]-[39]

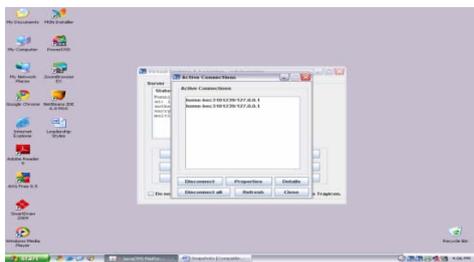


Figure 10.p: Snapshot of Active Connections

q) **Stage 17:** This is the About GUI.

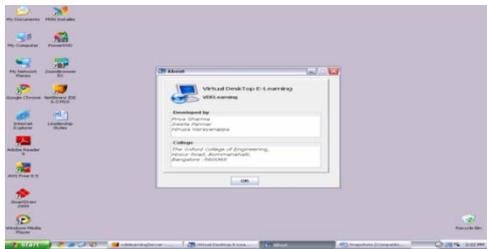


Figure 10.q: Snapshot of About

r) **Stage 18:** This GUI displays about Remote Host Properties.

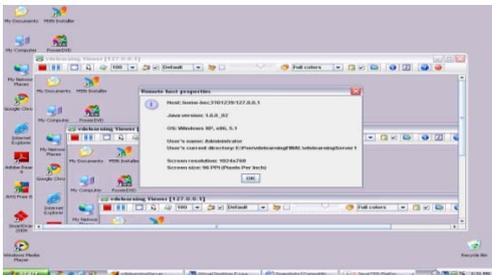


Figure 10.r: Snapshot of Remote Host Properties

s) **Stage 19:** This GUI displays Connection Information.

## 9. CONCLUSIONS

With Server consolidation (which replies the client) using virtualization is gaining significant adoption in data centric environment. The deployment of enterprise telephony solutions in virtualized environments is of interest to reduce the footprint and to provide energy efficient solutions. However, the near real-time requirements of enterprise IP telephony applications pose unique challenges for deploying such applications in virtualized environments.

In this work, we modeled e-tutorial concept to impart education at various places without the barrier of distance as well utilization of network resources. Students can access for tutorials from their standalone PC. Although virtual videoconferences have employed in corporate environment. Here in this paper we tried to implement in academics using Java has software tool, applied e-commerce knowledge and networking concepts. Also this work gives to platform for enhancing applications like,

- Embedding video transmission along with voice transmission in application.
- Implementing Exam oriented application transfer where students can access it dynamically and submit to the instructor after completion of exam.

## ACKNOWLEDGEMENT

We would like to thank SLN SOLUTION PVT LTD, providing the facilities to implement this concept. We are grateful to the whole SLN SOLUTION PVT LTD team and warm regards from us.



<http://www.cisjournal.org>

## REFERENCES

- [1] Dan Hutchison, and Ernst Bekkering, "Using Remote Desktop Applications in Education", The Proceedings of the Information Systems Education Conference 2007, v 24 (Pittsburgh): \$2544. ISSN: 1542-7382, Nov 2, 2007.
- [2] The Java Handbook by Patrick Naughton, Tata McGraw-Hill.
- [3] Roger S. Pressman, "Software Engineering – A practitioner's Approach", Fifth Edition, McGraw-Hill International Edition, 2001.
- [4] Ian Sommerville, "Software engineering", Sixth edition, Pearson education Asia, 2000.
- [5] Yuedong Zhang, Zhenhua Song, Dingju Zhu, Zhuan Chen, and Yuzhong Sun, "Redar: A Remote Desktop Architecture for the Distributed Virtual Personal Computing", National Research Center for Intelligent Computing Systems, Chinese Academy of Science, October 2006.
- [6] Cay S. Horstmann, and Gary Cornell, "Core Java 2", The Sun Microsystems Press, 1999, Indian reprint 2000.
- [7] Mary Campione, and Kathy Walrath, "The Java Tutorial", Second Edition, Addison Wesley Longmans, 1998.
- [8] Patrick Naughton and Herbert Schildt, "Java 2: The Complete Reference", Tata McGraw Hill, 1999.
- [9] N. Doraswamy and D. Harkins. IPsec: The New Security Standard for the Internet, Intranets, and Virtual Private Networks. Prentice Hall PTR, second edition, March 2003.
- [10] Venkateswaran, R. (February/March 2001) "Various Services and Implementation Scenarios: Virtual Private Networks". Institute of Electrical and Electronics Engineers (IEEE) Potentials, 11-15.
- [11] Desmond Keegan Dublin, Ireland "The incorporation of mobile learning into mainstream education and training".
- [12] Tsvetozar Georgiev, Evgenia Georgieva, Angel Smrikarov "M-Learning - a New Stage of Å-Learning" International Conference on Computer Systems and Technologies - CompSysTech'2004
- [13] Bruno Zuga, Ilmars Slaidins, Atis Kapenieks and Armands Strazds "M-learning and Mobile Knowledge Management: Similarities and Differences" International Journal of Computing & Information Sciences Vol. 4, No. 2, August 2006, On-Line
- [14] Christine Armatas. Dale Holt, and Mary Rice, "Balancing the possibilities for Mobile technologies in higher education "ascilite 2005: Balance, Fidelity, Mobility: maintaining the momentum
- [15] Torstein Rekkedal and Aleksander Dye, Mobile Distance Learning with PDAs: Development and testing of pedagogical and system solutions supporting mobile Distance learners The International Review of Research in Open and Distance Learning, Vol 8, No 2 (2007), ISSN: 1492-3831
- [16] Roger S. Pressman, Software Engineering (A Practitioner's Approach), The McGraw-Hill Companies, 6 Editions 2002
- [17] Devine, S., Bugnion, E., and Rosenblum, M. Virtualization system including a virtual machine monitor for a computer with a segmented architecture. US Patent #6,397,242 (Oct. 1998).
- [18] Menon, A., Santos, J. R., Turner, Y., Janakiraman, G. J., and Zwaenepoel, W. Diagnosing performance overheads in the Xen virtual machine environment. Proceedings of the ACM/USENIX Conference on Virtual Execution Environments (June 2005)
- [19] [1] Kolås, L. and Staupe, A. "A requirement specification of a next generation e-learning system," ISBN: 978-82-8055-028-6, [http://www2.tisip.no/quis/public\\_files.php](http://www2.tisip.no/quis/public_files.php).
- [20] [3] D. Griffiths, "UNFOLD: Participate in creating the next generation of eLearning". LCCN Newsletter 8th issue, 2004. Available online at: <http://www.learningcitizen.net/articles/UNFOLDparticipateinc.shtml> (accessed 28 June 2006).
- [21] A. Dye, T. Fagerberg, and T. Rekkedal, "Designing an Always-Online Learning Environment for Mobile Learners and Teachers". EU Leonardo Project "Mobile learning: The next generation of Learning", NKI Distance Education, 2005. Available online at: <http://learning.ericsson.net/mlearning2/files/workpackag e2/designing.doc> (accessed 28 June 2006).
- [22] J. Traxler, "The evaluation of next generation learning technologies: the case of mobile learning", ALT-C 2006 The next generation Research proceedings. ISBN 0-9545870-5-7, 2006.
- [23] M. Johnson, O. Liber, S. Wilson, P. Sharples, C. and Milligan, P. Beauvoir, "Mapping the future: The personal learning environment reference model and emerging technology". ALT-C 2006 The next generation Research proceedings. ISBN 0-9545870-5-7, 2006.
- [24] Boehm, B. "A Spiral Model of Software Development and Enhancement". IEEE Computer, vol. 21, #5, May, pp 61- 72. 1988.
- [25] Heinich, R., Molenda, M., Russell, J.D. and Smaldino, S.E. "Instructional media and technologies for learning" 7<sup>th</sup> edition. Merrill Prentice Hall. ISBN 0-13-030536-7. 2002.
- [26] Di Domenico, F., Sterbini, and A., Temperini, M, "Analysis of commercial and experimental e-learning systems. Project QUIS: Quality, Interoperability and Standards in e-learning, Agreement n. 2004 -- 3538 /001 -- 001 ELE--ELEB14. The TISIP Research Foundation, Trondheim, Dec. 2005, Dec. 2006, ISBN 82-8055-017-8
- [27] JISC, "Effective Practice with E-learning". HEFCE (Higher Education Funding Council for England). 2004. Available online at: [http://www.jisc.ac.uk/uploaded\\_documents/jisc%20effective%20practice3.pdf](http://www.jisc.ac.uk/uploaded_documents/jisc%20effective%20practice3.pdf) (accessed 22 November 2006).
- [28] McCabe, M. (1998). Lessons from the Field: computer conferencing in higher education. Journal of Information Technology for Teachers Education, 7(1), 71-87.

<http://www.cisjournal.org>

[29] Phipps, R., & Merisotis, J. (1999). "What's the difference? A review of contemporary research on the effectiveness of distance learning in higher education". A Report from The Institute for Higher Education Policy, April 1999.

[30] Synnes, K. (1998). Distributed Education using the mStar Environment. World Conference on the WWW and Internet (WebNet'99). Rewarded Top Paper. Also published in an extended version at the Journal of Universal Computer Science.

[31] Ch. Bouras, A. Gkamas, and T. Tsiatsos, "Internet Protocols for Synchronous Distance Learning" 3rd International Workshop on Network-Based Information Systems-NBIS'2000, Greenwich, UK, September 4-8 2000, pp. 46-50.

[32] Jacqueline Bourdeau, Marie Ouellet, Roberto Gauthier, "Interactivity in Videoconferencebased Telepresentations", Proceedings of EDMEDIA/ED-TELECOM 98, Freiburg, Germany, June 20-25, 1998

[33] T. Hodes, M. Newman, S. Mccanne, R. H.Katz, And J. Landay, Shared Remote Control of a Video Conferencing Application: Motivation, Design, and Implementation, Proceedings of SPIE Multimedia Computing and Networking, San Jose, CA, January 1999, pp.17-28.

[34] H Kim, H. Lee, B. Shin, S. Hwang, Egccm:A Conference Control Module for T.120 based Multimedia Teleconference, Proceedings of International Conference on Telecommunications, Porto Carras, Greece, 21-25 June 1998, Vol IV, pp 135-139.

[35] M.A. Pérez, M.J. Verdú, F. Hernández, B.Rodríguez, B. Carro, M.A. Navazo, J. Redoli, R.Mompó, J. García, R. López, "Distance Education and Long-Life Learning", 15th IFIP World Computer Congress, Vienna and Budapest, August 31 - September 4, 1998.

[36] Andrew Burrell, "A Study into the Future use of Synchronous and Asynchronous Technologies in the Distance ESL Education Area with an Emphasis on Collaborative Learning", Proceedings of ED-MEDIA 97 & ED-TELECOM 97, Calgary, Alberta, Canada, June 14-19, 1997.

[37] VoIP: An Overview and Security Aspects, International Journal of Computational Intelligence and Information Security, Vol. 1, No. 9, November 2010.

[38] [3] Jim Wrubel, David White, Julia Allen, "High-Fidelity e-Learning: SEI's Virtual Training Environment (VTE)", TECHNICAL REPORT, CMU/SEI 2009-TR-005 ESC-TR-2009-005, CERT Program, January 2009.

[39] Yuedong Zhang, Zhenhua Song, Dingju Zhu, Zhuan Chen, and Yuzhong Sun, "Redar: A Remote Desktop Architecture for the Distributed Virtual Personal Computing", National Research Center for Intelligent Computing Systems, Chinese Academy of Science, October 2006,

40] Modern Systems Analysis and Design Third Edition Jeffrey A. Hoffer Joey F. George Joseph S. Valacich Copyright 2002 Prentice-Hall, Inc

[41] Object Oriented Analysis and Design Using UML A Whitepaper by Mark Collins-Cope of Ratio Group

[42] I. Craig Larman, "Applying UML and Patterns", Addison Wesley, 2000.

[43] Grady Booch, James Rumbaugh, Ivar Jacobson, "The Unified Modeling Language User guide", Addison Wesley 1999.

[44] Ali Bahrami, "Object Oriented System Development", McGraw Hill International Edition 1999 ( <http://www.cs.ucc.ie/~adrian/bahrami/> )

[45] Fowler, "Analysis Patterns", Addison Wesley, 1994  
5. Erich Gamma, "Design Patterns", Addison Wesley, 1994.

### AUTHORS PROFILE:



Mr. N. Rakesh is a PhD Student in Computer Science at PRIST University Thanjavur, Tamilnadu. Currently he is working as Senior Lecturer in Department of Information Science and Engineering at KNS Institute of Technology, Bangalore. He received the B.E in Information Science and Engineering degree from Visvesvaraya Technological University Belgaum, Karnataka and M.Tech in Computer Science and Engineering degree from Dr.M.G.R. University, Chennai in 2005 and 2007 respectively. Subsequently completed M.B.A in Systems from Alagappa University, Karaikudi, Tamilnadu and M.Sc in Computer Science from Alagappa University, Karaikudi, and Tamilnadu in 2009 and 2011 respectively. He is having 6 years of experience in academics and published papers in international journals, also is a member of CSI. His current research activities pertain to VOIP, ATM, Frame Relays, Network Security, Information Security, Wireless sensor Network and topics related to Networks.



Dr. S. K. Srivatsa received the B.E from Jadavpur University in 1968. He received the M.E and PhD degree from IISC Bangalore in 1972 and 1976 respectively. He retired as a professor of Electronics and communication Engineering, Anna University in 2005. Since then, he is a senior Professor in St. Joseph Engineering college, Chennai, India. He has produced 35 PhD's and he is the author of well over 400 publications in reputed Journals and Conference Proceedings. He is a Life Fellow/Life member in about two dozen professional societies. He is the recipient of about a dozen awards. His areas of interest include Logic design, graph algorithms, networks, image processing and robotics.