**Indian Institute of Information Technology, Allahabad**

**Object Oriented Methodology (OOM)**

**Mini Project-7**

**TAs: Mr. Ayush Sinha, Mr. Bagesh Kumar Ms Uphar Singh**

**Instructors: Prof. O. P. Vyas and Dr. Ranjana Vyas**

**OOM Mini Project # 07:** Implementation of Tic Toc Toe game (0 and X game)

**Abstract**

The program consists of a TicTacToeServer application that allows two TicTacToeClient application to connect to the server and play Tic-Tac-Toe.

**TicTacToeServer Class:**

As the TicTacToeServer receives each client connection, it creates an instance of innerclass Player to process the client in a *separate thread*. These threads enable the clients to play the game independently. The first client to connect to the server is player X and the second is player O. Player X makes the first move. The server maintains the information about the board so it can determine if a player’s move is valid.

**TicTacToeClient Class :**

Each TicTacToeClient application maintains its own GUI version of the Tic-Tac-Toe board on which it displays the state of the game. The clients can place a mark only in an empty square. Inner class Square implements each of the nine squares on the board. When a TicTacToeClient begins execution, it creates a JTextArea in which messages from the server and a representation of the board using nine Square objects are displayed. The

startClient method opens a connection to the server and gets the associated input and output streams from the Socket object. Connection to the server.will be built. Class TicTacToeClient implements interface Runnable so that a separate thread can read messages from the server. This approach enables the user to interact with the board (in the event-dispatch thread) while waiting for messages from the server. After establishing the connection to the server, execution of the client with the worker ExecutorService will be done .The run method controls the separate thread of execution.The method first reads the mark character (X or O) from the server,and reads messages from the server. Each message is passed to the processMessage method for processing.

**Specific Technology**

Java, Swing, other related technology

**Project Tasks**

Use case analysis, Design: Class Diagram, CRC and Implementation, User Interface

**Functional components of the project**

* Player X connected to server
* Player O connected to server
* Player X moved.
* Player O sees Player X’s move
* Player O moved
* Player X sees Player O’s move
* Player X moved.
* Player O sees Player X’s last move.

### **Submission** The project presentation has to be done by each member to show periodically the progress and the complete project submission should contain the following:

* UML diagrams:-Use Case diagram, Class diagram, CRC diagram(s), illustrating the design of your program.

### All the Java source code should necessary to compile and execute.