



IJIRCCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 10, Issue 6, June 2022

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 8.165



9940 572 462



6381 907 438



ijircce@gmail.com



www.ijircce.com

Messaging Through Web-Application Using - SOCKET-IO

Lalit Siraswa*¹, Seema Nagaraj*²

Dept. of MCA, Bangalore Institute of Technology, Bengaluru, India. *¹

Assistant Professor, Dept. of MCA, Bangalore Institute of Technology, Bengaluru, India*²

ABSTRACT: In the following research paper, I'm going to explain and demonstrate what's Socket.IO, the requirement for Socket.IO, and the way it works? Socket.IO is a javascript library that permits us to do duplex(Bi-directional, Two-way) communication between a client(Browser) and a server(NodeJs-Server) in real-time. Bi-directional communication only works if and only if a client had Socket.IO in the browser and the server has also embedded the socket.io package. Socket.io utilizes engine.io behind the scene to establish connections and to exchange information(data) between the client(Browser) and server(NodeJs). Engine.io is utilized for the server-side implementation whereas engine.io client is used for implementing the client-side. Socket.io utilizes WebSocket behind the scenes.

For instance: In the present age, a Chat application could be an essential tool for the world since it connects from any place and chats with them continuously with no difficulty and this component resolve the query of calling all time either for transmitting or forwarding the information for business use or for personal use, Now with the help of chat application client can basically send the message to the next client and resolve the issue or problem without calling. The most helpful piece of the chat application is that we can communicate a similar piece of the message(information) to various clients simultaneously with practically no difficulty as we can make a group of numerous individuals or even we can communicate the message to indicated clients to whom the source wants to send the message. chat application likewise conveyed the most sensible information in this as you can impart your ongoing area location to anybody anytime and even you'll have the option to share pictures or pdf. Chat applications on the Internet are a tool to convey immediately among Internet clients who are on the web or accustomed to being. This web application permits clients to convey without a doubt however from a significant distance.

I. INTRODUCTION

Socket.IO is great tool for growing constant applications with duplex(two-way, bidirectional) correspondence between the server-side(NodeJS) and the client-side(Browser). It takes helps the power of WebSockets alongside a few backup plans, including JSON(JavaScript Object Notation) long surveying and JSONP(JSON with Padding) long surveying through a solitary bound together API. It very well may be utilized to make two-way(bidirectional) collaborations, like constant dashboards, talk applications, and multiplayer games.

In chat applications, Socket.IO is used for transmits real-time data like if you want to get and show which user is online or offline or you can use the Socket.IO for raising the event for getting the notification of a received message, or it can be used for getting the data from the database and show in frontend in real-time. In Chat-Application, when a client joins a unique Socket ID is generated that uniquely identifies the sockets. The number of currently log-In users is equal to the number of unique Socket IDs. The Socket establishes its independent environment with a unique ID. There are many different types of Socket.IO events you can make use of them accordingly. When one user wants to send a personal message to another individual, he can either use a room-type scheme or can directly emit some event.

As time goes on, the growth of information and communication innovation is extending so quickly and will be a really significant role within the community. With knowledge and communication innovations, we can provide facilities for individuals to receive information and communicate with one another from anywhere and faster than before at any

time. By seeing the current scenario after the improvement in technology people stop using newspapers for acquiring knowledge and not many people use the news channel nowadays for getting knowledge(information), Bulk of individuals move towards using cell-phone devices for obtaining any sort of information and if we consider the chat application, in chat applications everyone is transferring news about what's happening around them or anywhere within the world, as with the help of the chat application information is spreading so faster than before. Nowadays, the chat application is built by a group of programmers and it has its own benefits and drawbacks as every other thing has, and this gives more choices to the people which application is better for us and according to their requirements and the features provided by the application they use the application.

II. LITERATURE SURVEY

In this segment, we examine the generally existing nonconcurrent transmission(communication) techniques and innovations, and their benefits and disadvantages, to get a piece of fair information on the necessities for Socket.IO. Then toward the finish of the part, we look at the foundation of Socket.IO, addressing the significance of Socket.IO progressively web application improvement.

In the Online Chat Application [1]The authors have shown the application which provides the functionality of permission(authorization) that allows the user to use their account anywhere anytime with the use of any mobile phone. The user is just required to send meeting recommendations(requests) before sending any message(communications, information) to any user. When the user accepts their recommendations(request) then only they both can send and receive messages(information) otherwise not. Users should log-In into their application with their mail-id and Password as there is no need for OTP. From this paper, it's been observed that online communication(transmission) allows users to talk with others in a fast and good manner. This paper shows the importance of conversation application in day-to-day life and its effect on the technological world.

Whereas a conversation system for personal networks or organizations is represented in a paper [2]. This system ensures the protection(security, safety) of the communication(information) and personal data which will be shared over the network. It also stores data in a secure way in the background. It develops a two-way communication system to add further features to existing traditional systems currently in demand and allow both group discussion(communication, chat) and private discussion(communication, chat) that enable an effortless and quick way of communication between people. This ensures data transfer without any limit and without the limitation of size, making people get connected(joined) to others at any time, from anywhere, to transfer different formats over the system and have the unlimited size to store communication data.

One more additional social networking application [3] delivers features like Social networking in the preferred language, Image backup, Image theft alert on-demand, Landmark detection, OCR(Optical Character Recognition), and Information about any image. Observed working with React Framework to make the interactive user interface and the web-based real-time chatting application does not need any added third-party account program, and the visual communication could be established alongside. The programming tools used in assembling this application are React.js, Node.js with the Express framework, and Mongo DB database.

jWebSocket([4] jwebsocket): JWebSocket is an open-source project, which was approved under Apache License 2.0 in 2010 as a beta version, and contains support for both client and server (jwebsocket) operations. JWebSocket is a separate program that makes sense of what goes cross-stage. Their downloadable collection is integrated with WebSocket server (Enhanced neighbor java), WebSocket client (Enhanced JavaScript), and Android, Symbian, and IOS Clients. Also, they have enabled a module called "Streak Bridge" with additional carefully planned programs. As shown by the manufacturers, editors can use this structure to create free applications, and in addition, this can be supported with support for existing web applications.

Spring ([5] Spring): Spring is another open-source project that was transferred under Apache License 2.0 in 2003. This WebSocket shows WebSocket affiliates and handshakes. One of the problems with the Wrench is that it uses a single cycle server (No cord disconnection) (Contributors, 2015). Incorporating this as a private circle will not be suitable for large business-level applications, but for small expansion projects. To integrate the Wrench for large business-level applications, creators can use middleware, for example, the RabbitMQ's STOMP WebSocket (Pivotal) module, between a PHP application and a WebSocket client.

Ratchet ([12] WebSockets for PHP): Ratchet is an unrestricted PHP library, which can be used to support frequently supported PHP web applications via WebSocket. Requires PHP 5.4+ to get the best shows. It consists of nine parts, which are transmitted for various purposes. For example IoServer (application base), WsServer component (allows the editor server to transfer applications using the W3C WebSocket API), FlashPolicy (licensed programs, not compatible with WebSocket to communicate with applications and Flash organizations, etc.) By using these components, experts can add functionality as an HTTP display controller, WebSocket display controller, and so on.

Tornado ([6]Tornado): Tornado is a flexible, unrestricted web server and web application architecture, launched in 2009 via Python. It was developed for use by FriendFeed (Tornado) and was later transferred under the Apache License 2.0 as an open-source project. In their web design, there is a module, which is compatible with WebSocket, known as tornado.WebSocket.

Now I will present the protocol we found for developing real-time chat application applications:

Socket.IO ([7] Rauch): Socket.IO is a JavaScript library used to develop web applications in relationship with Node.js. Node.js is a JavaScript-based JavaScript processing component to work on the important development of objectives by using the unexpectedly well-known JavaScript server. Node.js owns an event-driven, non-disruptive I / O model, which spreads the word about a positive response to building robust applications. Socket.IO is one of the libraries that can be used with Node.js. Socket.IO was launched in 2010 and was developed by Guillermo Rauch. The most recent understanding is 1.4.5 which connects two APIs one for the server-side and the other for the client. Socket.IO uses a standalone module called Engine.IO which is a book-based operation based on a two-dimensional system based on two elements. It has made the new type easier.

Available asynchronous communication techniques and technologies:

In this piece of the paper, we look at the open strange correspondence techniques and advances, and their potential gains and drawbacks, to get a good perception of the requirement for WebSocket. Then close to the completion of the part, we discuss the underpinning of the WebSocket, showing the meaning of the WebSocket towards progressing web application headway.

AJAX: AJAX ([8] Garrett, 2005) is an integration of existing development (e.g., HTML, CSS, JavaScript, XML, etc.), which should be seen as a non-complex form of non-compliant communication. AJAX can be defined as a program rather than development, moreover, it can be seen in the same way as a design style. At AJAX, the client requests data/information/resources from the web server; and then, once the response has been received, the client processes it and makes changes to the page accordingly, without reloading the page however by updating key parts of the page, which is suggested as a deficit of the referral page.

Since communication is different, it does not confuse the client, and communication is made as soon as a small portion of the data is provided, appropriately, client information is redesigned. Client requests are submitted as XMLHttpRequest (XHR) (W3C, XMLHttpRequest Level 1, 2014) and sales can include a stand-alone, remote record server or make content possible by force. Items can communicate with the database server and perform CRUD tests (Create, Read, Update, and Delete). Data can be transferred through various game programs such as XML, JSON, HTML, etc.

An important issue with AJAX is that its connection is a dual component in data capture mode, thus not suitable for continuous applications (Liu and Sun, 2012). Similarly, in the setting of this paper, AJAX reflects the issue of C10K (Kegel, 2014), which states how to provide reasonable orders to no less than 10,000 simultaneous customers using a standard server (Liu and Deters). Special tests (Bozdag, Mesbah, and Deursen, 2007) show that if we really need high data understanding and high interaction, should select the AJAX data push method.

COMET: A comet is an umbrella term, which includes AJAX-based techniques such as live streaming and long reading ([9] Gravelle). Attempts to reduce and overcome interrupted points due to its failure to establish communication on the server-side by sending a request to the client. Comet attempts to tolerate these issues of data extraction process in AJAX, by imitating the push of data over data extraction. For example, consider an application made using a long-term update strategy; which consists of two stages: In the basic step, the client sends the transaction to the server and the server hosts the request open until there is data open to the client; and in the next step, data is sent when it is opened from server to client and one request is initiated by the client. This way, the server can send data to the client immediately. The client analyzes data from the server for commercial submissions, which remain open for a very long time until the data is unlocked for immediate transfer. This process is relatively simple yet has its drawbacks. For example, there may be a break if the client request is opened and it is unnecessarily extended. Then, in the next step, the client needs to restart the look. What else to do during a break can go in different situations.

Server-Sent Events (SSE): SSE ([10] W3C, Server-Sent Events, 2015) is expected to transmit message-based event data from the server directly to the client, considering the data push model. The Server-Sent Occasions EventSource API is standardized as part of HTML5 by W3C. SSE is used to drive alerts to a client.

Components of a web application, and that sets it up very well for communication with the ongoing client. SSE is designed to provide consistent web information to customers by pushing data to the client. Either way, it's not a double whammy, because it's just the best messages from a support server.

WebSocket: WebSocket ([11] W3C, The WebSocket API, 2012) offers a way to deal with making endless, less tiring encounters, good for dealing with books initiated by everything imaginable about a client or server. The client extends WebSocket relationships through a cycle known as WebSocket handshake. This cycle begins with the client sending a standard HTTP transaction to a server, where a renewed title is added to inform the server that the client wishes to extend WebSocket interaction. In the event that the server maintains the WebSocket display, it agrees to update and return this to the client using the updated reply header (WebSockets - Quick Guide). development and supporting libraries. WebSocket presentation is supported by HTML5, Java EE7, and more by many projects. In addition to Android, designers of iOS applications can use WebSocket using various libraries such as Autobahn-library. On HTTP, interactions are usually closed after each phase of the response. On the other hand, in a client-initiated WebSocket connection, where the connection is widespread it has never been closed until it is undoubtedly closed by the client or server. While the interaction is kept open for the client and the server can provide bidirectionally for two data transmission methods, as WebSocket maintains complete duplex bi-directional manuals, allowing data transfer to two subjects at once. Also, WebSocket solves the 10k problem, with its smaller titles and avoids the top-notch marketing articles above (Pimentel, Bolívar, and Nickerson, 2012).

III. METHODOLOGY

During our writing study, we determined that there are Socket.IO server implementations in a wide range of programming languages including Java, Python, Golang, JavaScript, NodeJS, and so on, and furthermore the client execution in many significant programming languages like (Java, Swift, Dart, Python, .Net, Rust, and Kotlin).

The Socket instance (server-side):

Whenever a new user joins, 20-character unique random is assigned for identification. This identifier unique id is synced with the value on the client-side. The client-side and the server-side socket ID are the same.

```

// server-side
io.on("connection", (socket) => {
  console.log(socket.id); // ojIckSD2jqNzOqIrAGzL
});

// client-side
socket.on("connect", () => {
  console.log(socket.id); // ojIckSD2jqNzOqIrAGzL
});
    
```

Fig 1: Connection event emitted when a new user joins

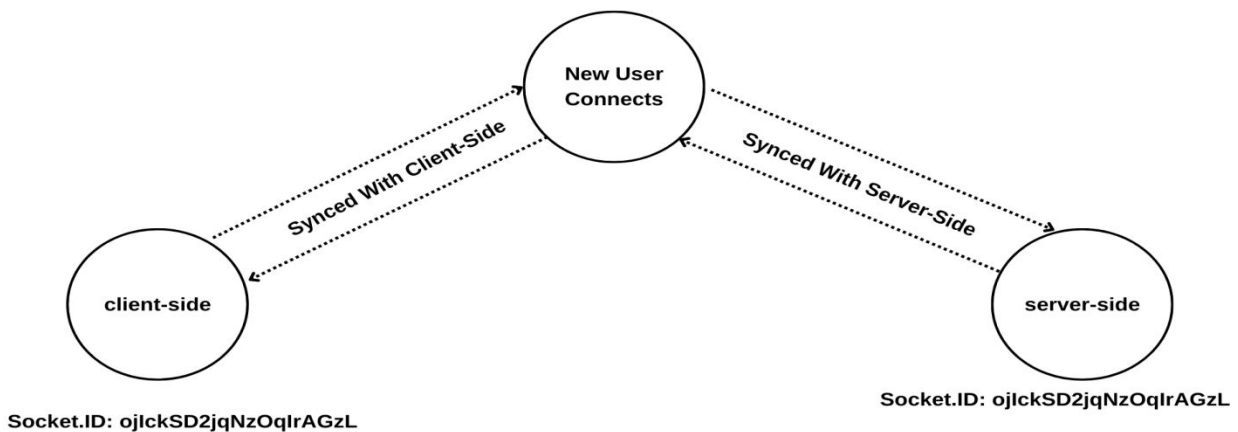


Fig 2: The Socket instance (server-side)

Whenever a user wants to raise some event, there is a predefined class provided by the Socket.IO i.e socket class. It includes all the methods of the Node.js EventEmitter, like emit, on, once, or removeListener. In the below-working flow, we show that there are mainly two functions emit() and on(). emit() is like a person who is speaking on a particular topic, the event() function also takes an event name(topic name) as an argument whereas the on() function is like another person who is paying attention or listening to that event. Here notice one thing the name of the event must be the same.

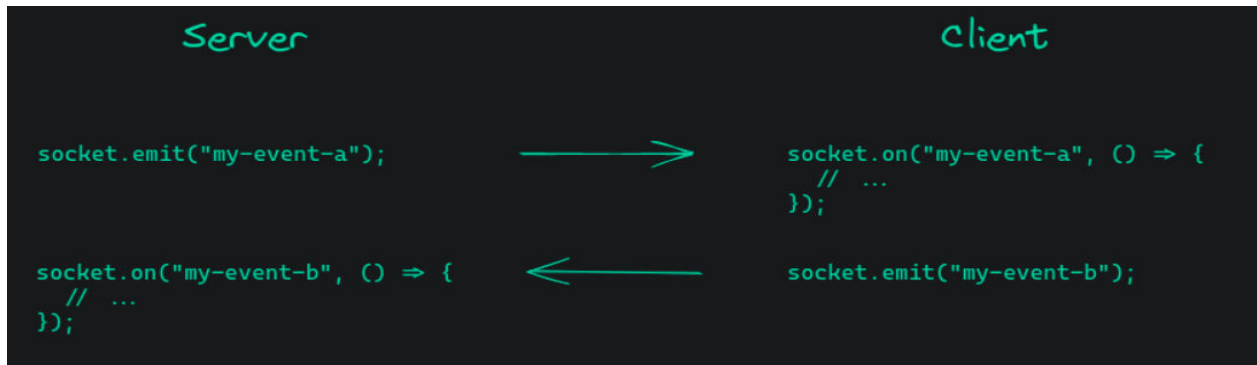


Fig 3: Emitting and listening to events

Some of the predefined events are:

1. connect
2. disconnect

These pre-defined events(attributes) represent whether the socket is currently connected to the server or not.

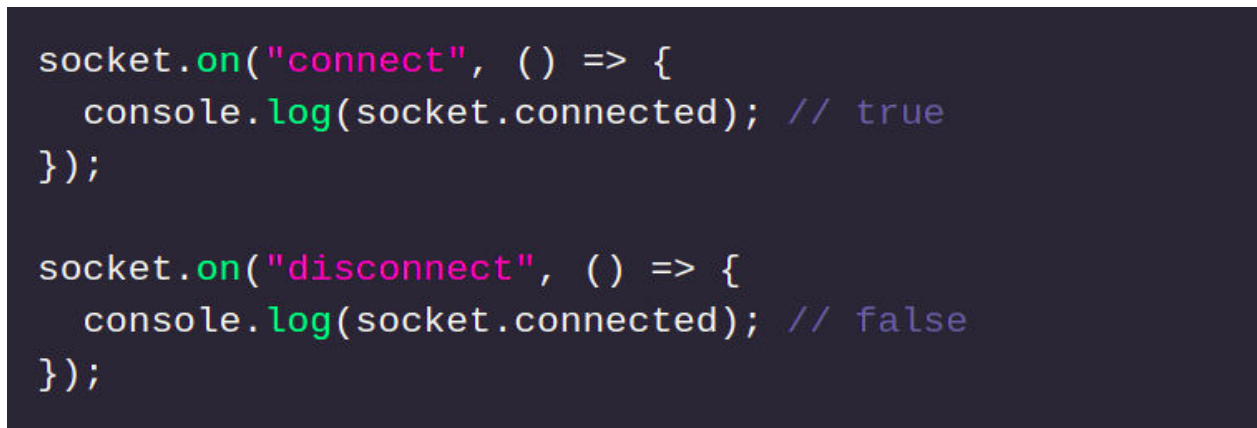


Fig 4: connect and disconnect event

The Socket instance (client-side):

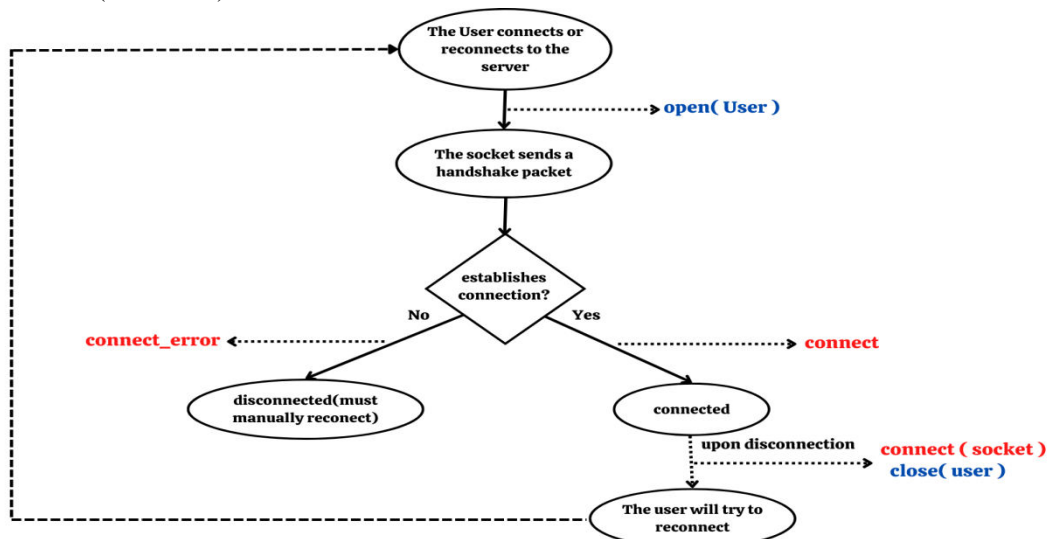


Fig 5: Life-Cycle of Socket

When a user connects or re-connects to the server, It opens a connection with the socket, then it sends a handshake packet to establish a connection. Both the client-side socket and the server-side socket have the same 20-characters unique id. If there is any connection error the connection is dismissed otherwise it is connected. And in the messaging application, you can make use of connecting and disconnect events to notify the other user whether the specific user is online or offline or you can also get information like nowadays we are seen on Facebook, WhatsApp, and Instagram users typing or not.

IV. CONCLUSION

The paper presents and assesses the underpinning of the overall existing techniques and advancements for completing nonconcurrent correspondence, moreover exhibiting their advantages and burdens, towards highlighting the upsides of Socket.IO. Then, at that point, the paper contains the information(knowledge) about the examination of the survey towards the blend of the overall information of Socket.IO improvement methods and developments into a lone distribution. Inside the examination, we've quickly investigated the features of the recognized thoughts, strategies, and headways. The information given in this paper can assist engineers with creating of Socket.IO - based applications, by saving the time that they ought to anyway spend on filtering and learning the procedures and progressions for Socket.IO.

After a study of the live visits in different fields, we observed that there are a few issues that are experienced undeniably and expected to be dealt with to improve the execution of the Chat Applications. Issues - increment the number of clients to be handled all the while, expanding the exhibition rate by a decrease in time delay, Memory spill. The web-based help conveyed by the business to its clients should have elite execution as clients favor completing their work online in light of the fact that it saves their season of truly visiting the specific area and working out there. In the event that the assistance conveyed online is sufficient, it'll clearly help their business in this web world today.

REFERENCES

- [1] In Jhalak Mittal, Arushi Garg, Shivani Sharma, 'Online Chat Application', Jhalak Mittal, International Journal of Research in Engineering, IT and Social knowledges, ISSN 2250-0588, Impact Factor 6.565, Volume 10 Issue 04, April 2020, go-between 10-16
- [2] In R. Gayathri, C. Kalieswari, 'Multi-User Chatting Application', International Journal of Engineering and Advanced Technology (IJEAT) ISSN 2249 – 8958 , Volume- 9 Issue- 5, June 2020
- [3] In Akhilesh Sarjit M S, Srivishak V, Shiddarth S, Saravana Kumar P, Preethi D, ' Web conversation using React Framework', International Journal of Trend in Scientific Research and Development (IJTSRD) Volume4 Issue- 3, April 2020
- [4] jwebsocket. (n.d.). Retrieved 02 16, 2016, from jwebsocket: <http://jwebsocket.org/>
- [5] (n.d.). Retrieved 03 13, 2016, from Spring: <https://spring.io/>
- [6] Thorson, P. (n.d.). zaphoyd/websocketpp. Retrieved March 26, 2016, from github.com: <https://github.com/zaphoyd/websocketpp>
- Tornado. (n.d.). Retrieved February 17, 2016, from Tornado: <http://www.tornadoweb.org/en/stable/>
- [7] Rauch, G. (n.d.). SOCKET.IO 1.0 IS HERE. Retrieved March 04, 2016, from Socket.io: <http://socket.io/>
- [8] Garrett, J. J. (2005, February 18). Ajax: A New Approach to Web Applications. Retrieved from https://courses.cs.washington.edu/courses/cse490h/07sp/readings/ajax_adaptive_path
- [9] Gravelle, R. (n.d.). Comet Programming: Using Ajax to Simulate Server Push. Retrieved April 26, 2016, from webreference: http://www.webreference.com/programming/javascript/rg_28/index.html
- [10] W3C. (2015, February 03). Server-Sent Events.
- [11] W3C. (2012, September 20). The WebSocket API.
- [12] W3C. (2014, January 30). XMLHttpRequest Level 1.
- [13] WebSocket Support. (n.d.). Retrieved 01 16, 2016, from Spring: <http://docs.spring.io/spring/docs/current/springframework-reference/html/websocket.html#webscketserver>



INNO  SPACE
SJIF Scientific Journal Impact Factor

Impact Factor: 8.165

 **doi**[®]
CROSS **ref**

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

 9940 572 462  6381 907 438  ijircce@gmail.com



www.ijircce.com

Scan to save the contact details