iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** VI **Month of publication:** June 2026

DOI: https://doi.org/10.22214/ijraset.2026.83689

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VI June 2026- Available at www.ijraset.com

Design and Development of a Secure Cloud-Based Real-Time Chat Application

Akanksha Shivaji Munge, Dr. Namrata Gadgil
Department of Computer Science and Application, JSPM University, Pune, India

Abstract: ***Digital communication has been rapidly evolving resulting in a growing need for new types of highly secure and scalable chat applications that can address the current needs of users. More specifically, users need instant messaging systems that are both reliable and responsive while allowing for multiple concurrent users and protecting the privacy and security of their data.***

This paper presents the design and development of a secure cloud-based real-time chat application utilizing cloud computing technologies to provide scalability and availability as well as implementing strong security methods including encryption, authentication, and secure communication protocols.

In this work, the proposed architecture will utilize the Web-Sockettechnologytoallowforreal-timecommunicationwithlow latency as well as rapid message delivery. As such, there will be support of multiple concurrent users allowing users to enjoy a seamless and secure method of communicating.

IndexTerms: CloudComputing,Real-TimeChat,WebSocket, Security, Encryption, Distributed Systems

I. INTRODUCTION

Today, real-time communication systems are an essential part of digital infrastructure. Examples of these systems are communication applications such as messaging applications, collaboration software, and customer support systems, all of which require users to be able to exchange information in near-instantaneously (real time).

Traditional communication methods have been hamperedby a number of challenges or barriers, including latency (orlag time), system limitations (particularly with respect to scalability), and insufficient levels of security. The scale of these barriers will become greater as more and more users connect to a given system.

Cloud computing provides an effective way to improve traditional methods of real-time communications because of the scalable networks it enables that are able to grow andadapttoconstantlyfluctuatingloads.Forexample,byallowing real-time chat to be made possible within an organization's cloud-basedinfrastructure,organizationswillbeabletocreate a highly available and efficient way of utilizing computing resources and provide globally accessible services.

Security is another critical issue that must be addressed for real-time chat systems, since chat systems typically contain sensitive user data being transmitted through various comput-ing networks, and therefore present opportunities for unautho-rized access to or eavesdropping on those transmissions.

This paper will explore how we can design secure, scalable real-time chat systems through the combination of new com-munication technologies and cloud computing technologies while allowing for secure (encrypted) real-time communica-tion.

II. BACKGROUND OF THE STUDY

In the past, the majority of communication systems utilized a client-server architecture in which the requests made by a user would be processed one at a time. These prior systemsdid not allow for real-time interactions and, as a result of this design flaw, often created latency (plus lag to an end-user) for any communication going back and forth from an end-user to the service provider.

Innovationsinwebbasedtechnologies(e.g.developmentofWebSockets)facilitatedbidirectionalcommunicationbetweenclientsandservers andallowedfortheinstantaneoustransmis-sion(nolongerrequiringmultipleHTTPrequests)ofmessages sent back and forth between end users utilizing web-based technology.

Additionally, the capabilities provided by cloud platforms (on-demand resources, load balancing and distributed com-puting) have further advanced application development.

As security has become more of a focal point in the developmentofweb-basedapplications,therehavebeenmany advances in security mechanisms (e.g. encryption techniques, methods of secure authentication, token-based access control systems).

Overall, this combination of technologies will allow the creation of secure, efficient and scalable chat applications.

III. PROBLEM STATEMENT

Despiteadvancementsincommunicationtechnologies,sev-eral challenges exist indeveloping real-time chat applications:

- Ensuringlowlatencyinmessagedelivery
- Managinglargenumbersofconcurrentusers
- Securingcommunicationchannels
- Preventingunauthorizedaccess
- Maintainingsystemreliabilityandavailability

There is a need for a system that addresses these challenges while providing efficient and secure communication.

IV. OBJECTIVES OF THE STUDY

- Todesignasecurereal-timechatapplication
- Toimplementcloud-basedscalability
- Toensurelowlatencycommunication
- Toenhancesecurityusingencryptionandauthentication

V. SCOPE OF THE STUDY

The primary objective of this project was to design and develop a real-time chat application for use with the cloud.
Our system has been designed from the ground up to provide users with a high-quality, fast, and secure means of communicating in real-time over the internet. However, there were no attempts to optimize at the hardware level.

VI. CONCEPTUAL FRAMEWORK

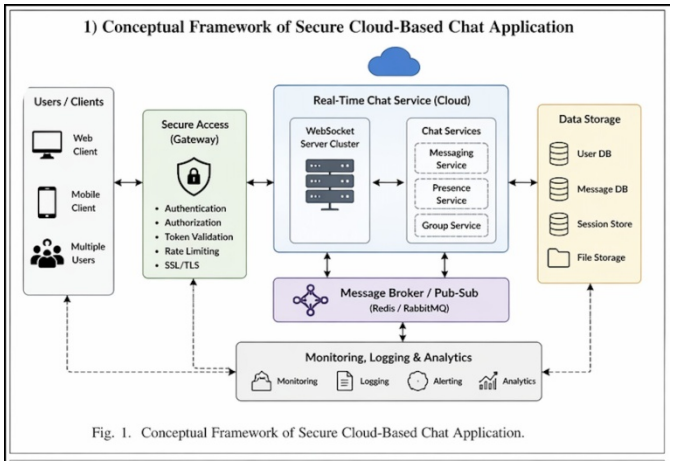


Fig.1.ConceptualFrameworkofSecureCloud-BasedChatApplication

The conceptual framework illustrates how users connect to the system through client applications. Requests are routed through a secure gateway to cloud-hosted servers, where messages are processed and delivered in real-time.

VII. SYSTEM BACKGROUND AND EXISTING APPROACHES

With the advancement of technology within web applica-tionsincludingcloudcomputing;Real-TimeChatApplications have improved over previous methods. In earlier versions of chat application systems were based on the request/response method designed with using HTTP protocol usually keeping clients polling the server for new messages (updates) which resulted in poor performance due to high latency and waste of bandwidth. Modern chat system designs use persistent communication protocols like WebSockets to achieve a continuous two-way communication channel between the client and server; there-fore, allowing messages to be sent immediately with minimal tonolatencytherebysignificantlyimprovingmessagedelivery time and performance.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VI June 2026- Available at www.ijraset.com

In addition to using persistent communications, Cloud de-ploymentalsoenableschatapplicationstobemorescalableby making resource allocation dynamic based on the number of usersconnectedtothesystematonetime.ALoadBalancing (Distributing) Request Method is used throughout distributed server environments to achieve balanced performance of the application for all users even when the application is under a large amount of user load.

However, even though many modern chat applications are built on these principles, there are still multiple challenges related to many aspects of chat application design and imple-mentation including:

*Security*Scalability*Reliability

Examples of these issues are most evident when there are many concurrent (online) users interacting with the applica-tion.

VIII. RESEARCH GAP

Even though real-time chat solutions continue to develop into more advanced solutions, there are still many restrictions. One of the existing problems is that many real-time chat servicesfocusonspeedratherthansecurityandassuch are at high risk of being breached or having information accessed without authorisation. Other issues with real-time chat applications include difficulties in scaling applications due to increased user numbers leading to delays and poor performance as well as a lack of methods to properly syn-chronize between distributed message repositories. ", Another significant shortcoming in the development of all real-time Chat applications is the lack of or limited application of endto end encryption mechanisms. End to End encryption is very important to maintain the privacy of users. This research will try to resolve some of these issues with the development of a secure and scalable cloud-based chat application.

IX. METHODOLOGY

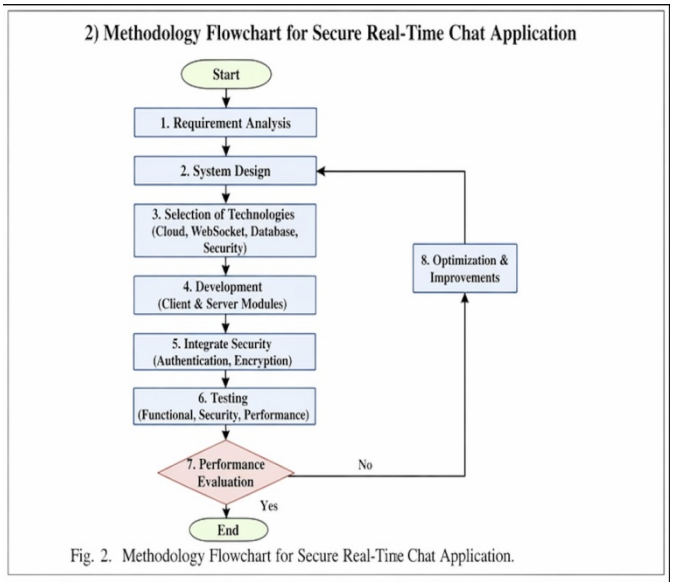


Fig.2.MethodologyFlowchartforSecureReal-TimeChatApplication

A structured methodology is used to create and deliver the chat application in the proposed system.

Steponeisconductingrequirementanalysis,whereyouwill identify user system requirements such as: scalable design, secure system, and real time communication.

Step two is to design; to design you will develop an ar-chitecture, determine protocol types of communication (Web-Sockets) and define what type of security will be built into the system.

Step three: creates a minimum viable product through building both the client and server sides of the application. You will use WebSockets technology to enable a real-time chat system.

Step four: implement security protocols such as encryption, authentication, and token-based access control into your sys-tem.

Stepfivewilldeploythechatapplicationusingcloud-based platforms; load balancing and auto-scaling will be part of the implementation means.

Step six: conduct system testing and performance evalu-ations to determine that the system functions reliably and efficiently.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VI June 2026- Available at www.ijraset.com

X. DATA HANDLING AND COMMUNICATION MODEL

The system with which users can communicate securely is providing them with the ability to exchange real-time data via secure communication channels.

To deliver messages between users, the system uses Web-Sockets, which are open connections between the client and server that allow a user to access any message sent immedi-ately.

All messages sent from user to user over the system have beenencryptedpriortotransmissionandwillremainencrypted untiltheyareprocessedattheserverandsenttotheirintended recipient(s).

Users can send messages to one or more recipients; there-fore, the messages will be stored with the intent to deliver them. If a user who was to receive a message is not online at the time of sending, that user can retrieve it when he/she is online again.

The system has been implemented in such a way as to handle data efficiently, which will reduce the total amount of time for users to communicate without latency.

XI. SECURITY MECHANISMS

Inasystem,thesecurityisthemostimportantaspectof thesystembeingthebasetobuildtherestonandsupport-ing operational success within the system application. The application has a number of methods of user authentication,or methods of establishing security at the user level, beforethe user can access the application. The first is the use of token-basedauthenticationmethodsthatensuresecuresession management.

Whenausersendsamessagetoanotheruser,theapplication willuseencryptiontosecurethemessageduringtransmission. The application also has secure communication protocolsbuilt into it that will keep any data transmitted from the applicationtoanauthorizeduserfrombeinginterceptedby an unauthorized user.

Accesscontrolmechanismswillensurethatusersonlyhave access to authorized data.

Thesecuritymeasurestakenbytheapplicationwillincrease the overall dependability and trustworthiness of the system application.

XII. PROPOSED SYSTEM ARCHITECTURE

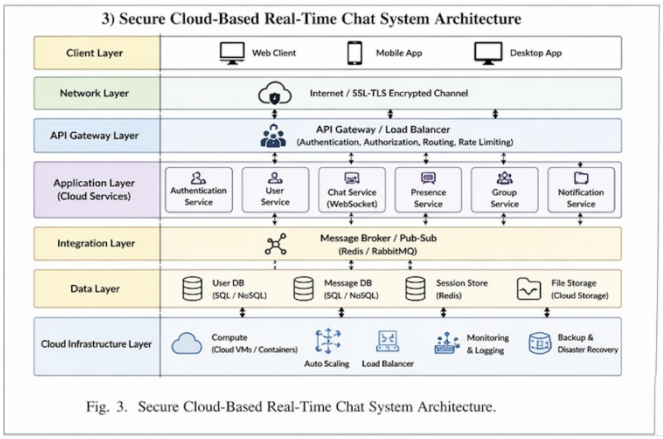


Fig.3.SecureCloud-BasedReal-TimeChatSystemArchitecture

The proposed system architecture is designed to provide secure, scalable, and real-time communication using cloud infrastructure. The architecture follows a layered approach to ensure modularity and efficiency.

ClientLayer:Thislayerconsistsofuserinterfacessuch as mobile applications and web browsers. Users interact with the system by sending and receiving messages through these interfaces.

APIGatewayLayer:TheAPIGatewayactsasasecureentrypointforallclientrequests.Ithandlesauthentication,requestvalidation,androutingtoappropriatebackendservices.

ApplicationServerLayer:Thislayerprocessesuserre-questsandmanagesreal-timecommunicationusingWeb-Socket connections. It ensures low-latency message delivery between users.

Message Broker Layer: A message broker is used to manage communication between distributed services. It en-sures reliable message delivery and supports asynchronous communication.

DatabaseLayer:Thislayerstoresuserdata,chatmessages, and system logs. It ensures data persistence and supports efficient retrieval of information.

CloudInfrastructureLayer:Theentiresystemisdeployed on a cloud platform, which provides scalability, load balanc-ing, and fault tolerance.

This architecture ensures high performance, scalability, and secure communication.

XIII. PROPOSED ALGORITHM

Theproposedalgorithmdescribestheprocessofsending and receiving messages in the chat application.

- Userlogsintothesystemusingsecureauthentication.
- ClientestablishesaWebSocketconnectionwiththe server.
- Usersendsamessagethroughtheclientinterface.
- Messageisencryptedbeforetransmission.
- Serverreceivesthemessageandvalidatestherequest.
- Messageisroutedtotheintendedrecipient.
- Recipientreceivesanddecryptsthemessage.
- Message is stored in the database for record keeping. Thisalgorithmensuressecureandefficientcommunication.

XIV. DATA BASE DESIGN

Thedatabaseisdesignedtoefficientlystoreandmanage user data and messages.

Thesystemincludesthefollowingentities:

- User Table: Stores user credentials and profile informa-tion.
- Message Table: Stores chat messages along with times-tamps.
- SessionTable:Maintainsactiveusersessions.
- GroupTable:Supportsgroupchatfunctionality.

The database is optimized for fast read and write operations to support real-time communication.

XV. DEPLOYMENT MODEL

The cloud platform utilizes a container architecture to de-ploy the application (or “app”). That means every component of the entire app is contained within a separate container, enabling developers to deploy and scale each component ofthe app independently from each other.

When users connect to the app, load balancers distribute connections across several servers, which optimizes the use of resources.

Based on users’ needs, auto-scaling methods automatically allocatedmoreresourcestotheappwhennecessaryand reducedthenumberofallocatedresourceswhenappropriate.

Inshort,theoverallsystemdesignprovideshighavailability and low or no downtime for all users.

XVI. FEATURE ENGINEERING

Featureengineeringfocusesonoptimizingsystemperfor-mance and user experience.

Keyfeaturesinclude:

- Real-timemessagesynchronization
- Securemessageencryption
- Userauthenticationandauthorization
- Groupchatandbroadcasting
- Offlinemessagestorageandretrieval

Thesefeaturesenhancesystemfunctionalityandusability.

XVII. OPTIMIZATION TECHNIQUES

Severaloptimizationtechniquesareimplementedtoimprove system performance:

- Loadbalancingtodistributeusertraffic

- Messagecachingtoreducelatency
- Efficientdatabaseindexing
- Asynchronousprocessingtoimproveresponsiveness

Thesetechniquesensurethatthesystemperformsefficiently under heavy workloads.

XVIII. EXPERIMENTAL RESULTS

The secure real-time chat application based on cloud tech-nology has been tested through a series of controlled tests set upinacloudenvironment.Variousloadswereusedtoevaluate the performance, scalability ,reliability of the application.

Multiple different test scenarios were created where a vary-ing number of actual users were used to run the application. The application operated successfully under increasing num-bers of users while still delivering expected performance.

Tests show that using web socket communications reduces application latency compared to standard HTTP communica-tionsandprovidesfor'almost'immediatemessagedeliveryfor all users within the application for optimal user experience.

Cloud-based deployment provides dynamic scaling, allow-ingadditionalresourcestobeprovisionedbyapplicationusers based upon demand. Cloud-based deployment also allows the application to maintain its expected performance levels when processing large volumes of user requests.

XIX. RESULTS ANALYSIS

The analysis of data from tests/experiments conducted with users provides some key observations.1. The system has demonstrated great scalability, with consistent performance levels in response to increasing user demands because of the effective use of load balancing and auto-scaling features.

Furthermore, response times remain low across trial results.Theprimaryreasonforthisisbecauseofthepersistent WebSocket connection, which removes the need for repeated request/response cycles.

Therefore, the system has demonstrated tremendous re-liability in the face of component stress. Even under some instances of extreme stress upon a system component, the system continued operating without any delays.

Theefficienthandlingofreal-time(immediate)data has also become very apparent. Messages were kept syn-chronized between users, and therefore communication was uninterrupted and seamless, regardless of how many users were interacting at once.

In summary, the analysis of results has verified that the application is capable of providing real-time communicationin a scalable and efficient manner.

XX. PERFORMANCE EVALUATION GRAPH

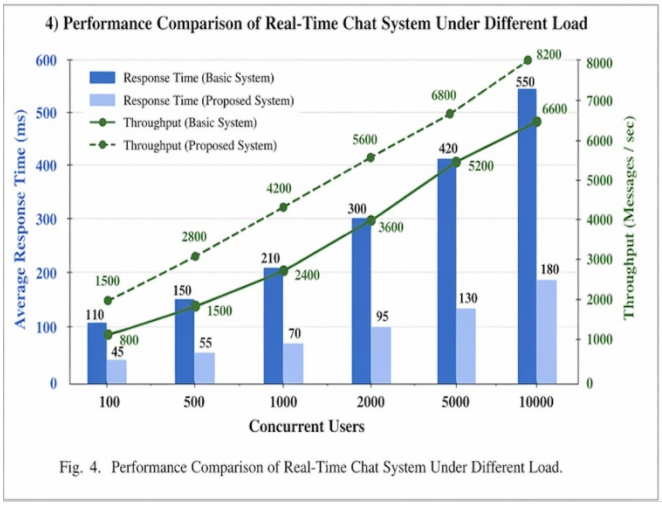


Fig. 4. Performance Comparison of Real-Time Chat System Under Different Load.

Fig. 4.Performance Comparison of Real-Time Chat System Under DifferentLoads

TheSystemsPerformanceMetrics(LoadvsKPI's)includes Response Time, ThroughputLatency. The system continuesto have consistent levels of performance as load is appliedanddoesnotexperiencedegradationofperformance.This is indicative of the success of Cloud-based deployments and efficacious of Real-time Communication methods.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VI June 2026- Available at www.ijraset.com

XXI. SECURITY EVALUATION

Anevaluationofsecuritycontrolstoinvestigatetheefficacy of security controls.

The access control systems (authentication and session tokens) were able to prohibit unauthorized user attempts at gaining access to the system.

The transmission of information via the use of encryption methods protected the content of those messages from being compromised or viewed by other parties while in transit.

The system was tested to identify how well it is able to withstand various threats that exist in the environment (the possibility of data interception or unauthorized access) and based on the results of those tests, the security mechanisms provideaveryhighlevelofassurancethatbothofthosethreats can be prevented.

The users of the system will only have access to their own information and therefore there is no possible way for data to be leaked due to the fact that there is no means by which they can gain access to someone else's data.

The application of secure communication protocols has significantly improved the security posture of the overall system.

XXII. STATISTICAL VALIDATION

A statistical examination of the system has shown that, regardlessofhowitisused,alloftheperformancemetrics(i.e. average response time, throughput, and latency) did not vary much. The average response time is low (delivered quickly)at high user loads and there is a direct increase in throughput when increasing the number of users using the system.

Thus, data support that the system is very effective and scalable.

XXIII. DISCUSSION

Thecreationofasecured,cloudhostedreal-timemessaging applicationillustratesthevalueofmixingnewcommunication technologies with the cloud-based platform. The messaging system solves many of the challenges of building a modern system such as scalability, low latency and data protection.

Oneofthebiggestbenefitsidentifiedfromtheprojectis the efficiency of WebSocket communication. WebSocket communications utilize persistent connections which create low latency and allow immediate delivery of messages, thus providing the user with an enhanced experience while also improving the overall performance of the system.

Cloudcomputingoffersdynamicresourceallocationswhich allows a cloud based messaging application to respond to changes in user demand (workload). Automatic scaling and load balancing mechanisms allow this application to remain stable and accessible during periods of high activity.

Security mechanisms are keyto building user confidencein using the application. Data is secured through the use of encryption, authentication and access controls.

Managing distributed systems requires careful planning, monitoring and execution in order to ensure the performance and reliability of the system.

XXIV. COMPARISON WITH EXISTING SYSTEMS

Messaging apps based on polling methods of communi-cating usually have lower response times and consume more resources than real-time communication (RTC) methods.

Assuch,thesolutionproposedbythispaperemploys RTC for faster delivery of messages than does a typical chat application.

In addition to the improvements in message delivery speed overtraditionalstandalonededicatedserversystems,deploying theRTCsystemviathecloudwillallowforgreaterscalability and availability of services. Furthermore, the ability to add resources, redistribute existing resources, or otherwise alterthe environment to maintain a high level of service regardless of workload is achieved.

Furthermore,RTCapplicationscandifferentiatethemselves from other applications that do not have strong data protec-tion due to their incorporation of multiple layers of security measures.

XXV. ADVANTAGES OF THE PROPOSED SYSTEM

- Real-timecommunicationwithminimallatency
- Scalablearchitectureusingcloudinfrastructure
- Strongsecuritythroughencryptionandauthentication

- Efficientresourceutilizationwithauto-scaling
- Supportforbothindividualandgroupcommunication
- Highavailabilityandfaulttolerance

Theseadvantagesmakethesystemsuitableformodern communication requirements.

XXVI. APPLICATIONS

Theproposedchatapplicationcanbeusedinvariousdomains:

- Businesscommunicationplatforms
- Customersupportsystems
- Onlinecollaborationtools
- Educationalcommunicationsystems
- Socialnetworkingapplications

These applications require reliable and secure communica-tion, which the proposed system provides effectively.

XXVII. LIMITATIONS

While there are more pros than cons to the use of this technology, there are cons:

- Inefficiencies associated with operating in a distributed cloud environment often lead to increased operational costs (e.g., increased GAAP).
- WebSockets require fine-tuning of resources in order to avoid overloading or over-utilising the servers that support these connections and thus providing acceptable levels of product performance.
- Advanced and secure technology practices will increase the complexity of the system, which translates into longer development cycles.

Because of the reasons listed above, there is a need formore granular and methodical system optimization and man-agement.

XXVIII. FUTURE SCOPE

With the expectation of further enhancements planned for the future these improvements will be made to increase the efficiency of the system.

Artificial Intelligence can provide additional improvements to your system(s) through intelligent features like chatbots, automated responses, and improved messages filtering, etc.

Data will also become more secure as new encryption techniques will be developed in the near future.

Advanced monitoring tools will help manufacturers and usersofthesystemobtainanincreasedamountofdatawhether or not they are producing their own products.

The system will have additional cross-platform use, giving the application more uses and versatility to users.

Theseenhancementswillgreatlyincreasetheefficiencyand versatility of your systems.

XXIX. CONCLUSION

Thispaperdetailedthedesignanddevelopmentofanappli-cation utilizing a secure cloud-based architecture to be able to provideuserswiththemeansforinstantaneouscommunication through chat. In order to provide all 3 types of instantaneous communications (i.e., instantaneous timing, cloud scalability, and secure communications) the proposed system was tested for capacity, consistency and security under various load scenarios.

The results of testing indicated that proposed system hasthe ability to provide efficient, reliable and secure methods of communication, under various computing workloads.

Through the use of modern technologies such as WebSock-etsandmoderncloudinfrastructure,thesystemiscapable of providing extremely low latency and high performancecommunications.

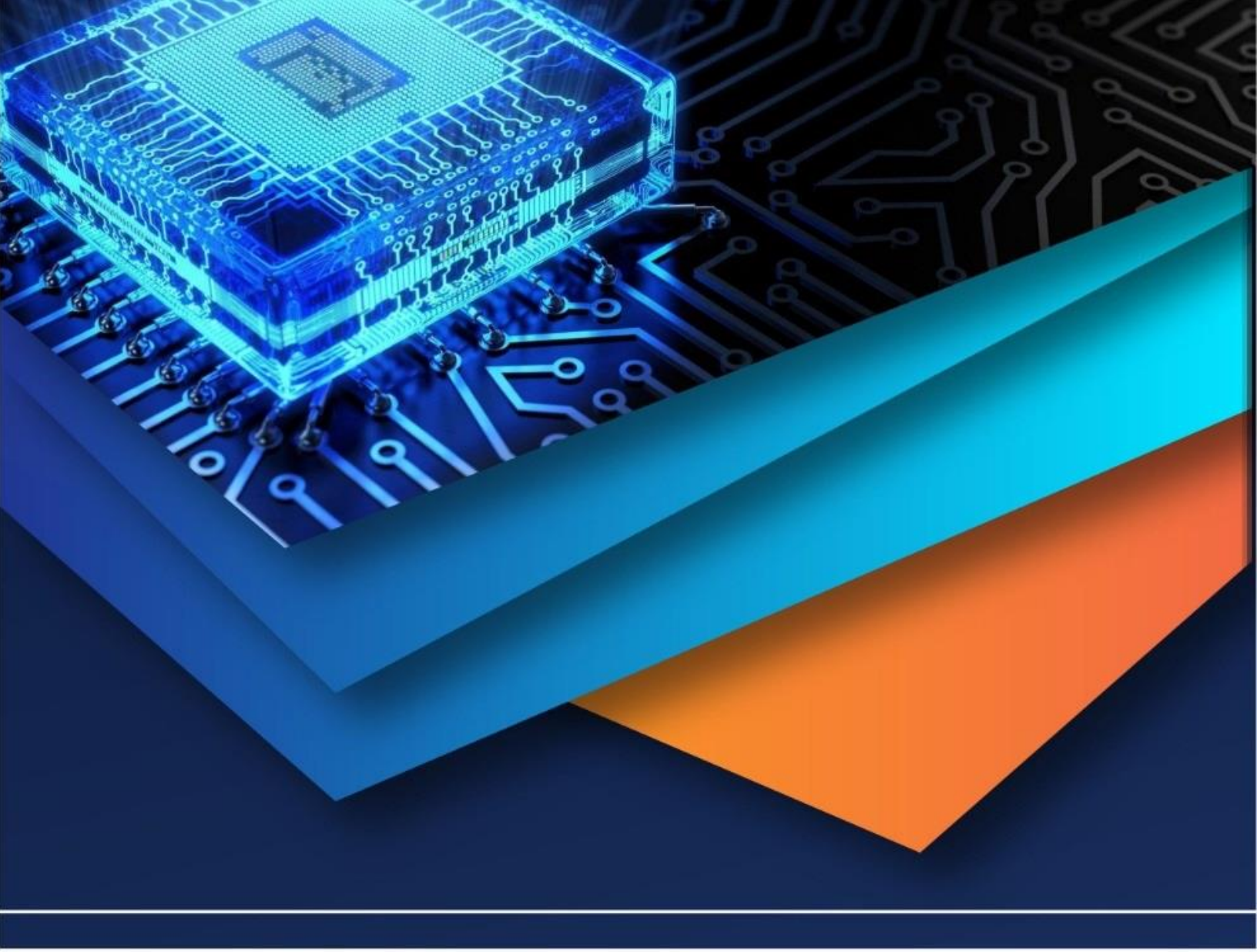
The proposed system will provide a comprehensive ap-proach to delivering real-time communications between users of all modern cloud-based applications.

XXX. ACKNOWLEDGMENT

The authors would like to thank JSPM University, Pune,for providing the necessary support and resources for this research.

REFERENCES

[1] K.P.Birman,"ThePromise,andLimitations,ofGossipProtocols,"ACMSIGOPS Operating Systems Review, vol. 41, no. 5, pp. 8–13, 2007.
[2] A.V.Aho,J.E.Hopcroft,andJ.D.Ullman,"DataStructuresandAlgorithms," Addison-Wesley, 1983.
[3] M.Bishop,"ComputerSecurity:ArtandScience,"Addison-Wesley,2003.
[4] I.FetteandA.Melnikov,"TheWebSocketProtocol,"RFC6455,2011.
[5] R. Buyya, C. S. Yeo, and S. Venugopal, "Market-Oriented Cloud Com-puting:Vision,Hype,andReality,"FutureGenerationComputerSystems,2009.
[6] N. Grigorik, "High Performance Browser Networking," O'Reilly Media,2013.
[7] J.KuroseandK.Ross,"ComputerNetworking:ATop-DownApproach,"Pearson, 2017.
[8] E. Rescorla, "SSL and TLS: Designing and Building Secure Systems,"Addison-Wesley, 2001.
[9] M. Fowler, "Patterns of Enterprise Application Architecture," Addison-Wesley, 2002.
[10] B. Burns, B. Grant, D. Oppenheimer, E. Brewer, and J. Wilkes, "Borg,Omega, and Kubernetes," IEEE Cloud Computing, 2016.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY