iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IV **Month of publication:** April 2026

DOI: https://doi.org/10.22214/ijraset.2026.79364

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 IV Apr 2026- Available at www.ijraset.com

BigBid: A Real-Time Auction and E-Commerce Platform

Dr A Naresh Kumar, Boyoni Balaji Lakshmiah Raj D, Ponasanipalli Prasuna, Mohammed Omar
Computer Science and Engineering (Data Science) Institute of Aeronautical Engineering Hyderabad, India

Abstract: ***BigBidisareal-timeonlineauctionande-commerce platformthatcreatesavibrantandinterestingmarketplace by fusing traditional online shopping with competitive bidding. Buyers can take part in live auctions using an easy-to-use web interface, while sellers can offer products with customizable auction settings, such as base price, reserve price, length, and extra buy-it-now choices. A WebSocket-based communication layer is used to offer real-time bid updates and countdownclocks, guaranteeing low-latency bid synchronization across all connectedclientsandmaintainingfairnessduringperiodsofhigh bidding activity. The backend is constructed using Node.js etc Express.js, exposing RESTful APIs for essential functions and using role-based access control and JSON Web Token authenti- cation to protect administrator, buyer, and seller actions. High- frequency read-write operations during auctions are supported by MongoDB, a versatile NoSQL data store for user profiles, product listings, bids, and transaction histories. React.js and Tailwind CSS were used to create the frontend, which offers interactive auction views and responsive dashboards that are designed for desktop and mobile devices. User testing reveals excellent levels of satisfaction with responsiveness and overall user experience, while experimental evaluation using concurrent user simulations demonstrates that the platform maintains low bidlatency,highbidsuccessrates,andstableperformanceunder load.Thesefindingsshowthat BigBidoffersascalable,safe, and user-focused solution for contemporary online auctions and hybrid e-commerce situations.***
Index Terms: Onlineauctions;real-timebidding;e-commerce platform; WebSocket; Node.js; MongoDB; React.js; role-based access control; auction management; web application

I. INTRODUCTION

The majority of traditional platforms still rely on fixed- pricelistings,whichrestrictdynamicpricediscoveryand userinvolvement, despitethefactthate-commerce'sexplosive development has completely changed how consumers and sellers interact. In order to close this gap, real-time auction systems allow competitive bidding, in which users make incrementaloffers withinapredeterminedwindowoftimeand thefinalpricesrepresentactualdemand.Nevertheless,alotof currentauction systemsrelyon pagerefreshesorpolling-based updates,whichcauseslatency,poorsynchronizationduringthe last few seconds of bidding, little seller control, and a terrible mobile user experience.

Conventional e-commerce sites like Amazon and Flipkart don't enable dynamic, time-sensitive auctions; instead, they concentrate on fixed-price inventory management. Although eBay and other auction-specific platforms have implemented timed bidding processes, many still employ polling or refresh- based methods that cause delays and allow bid sniping, in whichuserswinunfairlybyputtinglast-minutebidswith- out properly informing rivals. Although they lack integrated payment systems, real-time communication protocols, and thorough seller analytics, mobile-first resale apps offer rudi-mentaryauctionfunctionality.Together,theserestrictionslimit pricediscovery,loweruserinterest,andopenthedoortofraud and unfair trading activities.

BigBid is a complete real-time auction and e-commerce solution that integrates traditional "buy now" purchase with live bidding on a single, integrated platform. The system provides secure registration and login, role-based dashboards for buyers, sellers, and administrators, customizable auction parameters, including anti-sniping extensions, and real-time alerts for bid events and auction closing. All bid and status updates are promptly transmitted to active clients with laten- cies estimated in tens of milliseconds thanks to a WebSocket- enabled communication layer supported by a Node.js/Express backend and a MongoDB database.

The design, implementation, and assessment of a scalable, low-latency bidding website that shows how contemporary web technologies can allow transparent, user-friendly, and high-concurrency online auctions is the main contribution of this work. We demonstrate performance characteristics under realisticloads,describearchitecturaldesignprinciplesforreal- time e-commerce, and verify user satisfaction through testing and practical deployment.

II. LITERATURE SURVEY

Three major categories can be used to classify current e- commerce platforms: emerging mobile-first markets, auction- based platforms, and conventional e-commerce systems. Plat- forms for General E-Commerce: Products are posted with set prices on websites like Amazon, Flipkart, Snapdeal, and oth- ers,andcustomersjustneedtoaddgoodstotheircartsandfin- ishtransactions.Thesesystemsdonotenablecompetitiveprice or time-sensitive bidding methods, despite providing strong logistics, payment security, and recommendation engines. Asa result, consumers lose out on possible agreements through bidding, while sellers have less flexibility in price discovery. Technically speaking, these platforms are better suited for recommendation algorithms and inventory management than for the frequent updates required by auctions.

Auction-Oriented Platforms: Online auctions were first in- troducedbyeBayandQuibids,whichletusersputbidswithin predetermined time frames and alert bidders if they were outbid. But in many systems, real-time bidding isn't really live; users frequently have to reload pages or rely on sluggish polling methods, which causes latency and synchronization problemsduringpeakbids.Bidsniping,inwhichwinners are unfairly determined by last-minute bids, frequently affects high-demand listings. Some platforms have auto-bid capabili- ties to combat this, although these are frequently overused or poorly explained. Additionally, most systems are not flexible enough to dynamically transition between fixed-price and auction models.

New Mobile and Resale Platforms: Lightweight auction features that are available on mobile devices have been introduced by OLX, Facebook Marketplace, Bids.com, and related apps. These platforms, however, frequently functionin remote settings with little backend intelligence. Users are forced to conduct transactions outside due to the lack of integrated payment gateways, which increases risk and erodes confidence. Real-time technologies like WebSockets are also infrequently used. Seldom do these platforms provide seller analytics, dashboards, or strong fraud detection systems.

Real-TimeSystemTechnicalArchitecture:WebSocketpro- tocols and event-driven architectures are crucial for low- latency online applications, according to recent study. Withits event-driven paradigm and non-blocking I/O approach, Node.jshasemergedasapopularoptionforreal-timesystems. These technologies allow systems to manage thousands of concurrent connections with low latency when combined with frameworks like Express.js, Socket.IO, and NoSQL databases like MongoDB. WebSocket-based designs are perfect for auction platforms since they reduce network overhead by 60–70% when compared to HTTP polling, according tostudies on real-time web applications. Security and Trust in Online Auctions: Blockchain-based auction protocols, such thoseputoutbyGalalandYoussef,provideimmutability of auction records and cryptographic verifiability to sealed- bid auctions. Blockchain systems improve transparency, but theyalso presentscalability andlatency issuesthat makethem inappropriate for real-time bidding. Rather, BigBid uses con- ventional security best practices like secure payment gateway integration, encrypted data storage, role-based access control, and JWT authentication.

By fusing a WebSocket-driven real-time bidding engine with a modular service-oriented backend, responsive frontend design, and cloud-ready deployment, the suggested BigBid system fills these gaps. By showcasing a workable implemen- tation that strikes a balance between real-time responsiveness, security, scalability, and user experience, this study enhances the state of online auction systems.

III. SYSTEM DESIGN AND ARCHITECTURE

The four main layers of BigBid's layered client-server ar- chitecture are presentation, communication, application logic, and data storage.

A. PresentationLayer(Frontend):

The frontend, which includes role-based dashboards and displays for buyers, sellers, and administrators, is developed using React.js and Tailwind CSS. The buyer's interface in- cludes features such as viewing the bidding history, getting notified when they have been outbid by someone else, real- time bidding, and browsing auctions. The seller's dashboard allows users to set up their products for auction by uploading images and setting parameters such as the starting price, reserve price, duration of the auction, and a "buy it now" option.Theycanalsomonitortheirliveauctionsandsales.The administrator'sinterfaceallowsuserstomanageusers,approve listings, resolve disputes, monitor their live auctions, and get analytics about their platform. All these interfaces are respon- sive on different devices such as desktop, tablet, and mobile devicesusingmodernCSStechniques.Thefrontendmaintains persistent WebSocket connections for real-time updates and uses React's virtual DOM for efficient UI re-rendering when bid or status changes.

B. CommunicationLayer:

In the communication layer, WebSocket channels imple- mentedusingSocketsareintegratedwithHTTPS-basedREST- fulendpoints,whichprovidelow-latencybidirectionalcommu- nication.

RESTful APIs are used to manage standard CRUD operations, including user registration, product list retrieval, and payment processing. WebSocket channels provide real- time communication channels for each active auction, send- ing notifications on bids, countdowns, and status updates to connected clients. Socket.IO provides efficient room-based messaging, browser fallbacks, and reconnection capabilities, reducing bandwidth usage.

C. ApplicationLogicLayer(Backend):

Essential business operations and rules are present in this layer. This layer is built upon Node.js and Express.js. Someof the prominent modules of this layer are:

- Authentication and Authorization: JWT-based session managementwithRole-BasedAccessControl(RBAC)for administrators, sellers, and buyers.
- AuctionManagement:Theprocessofmanagingtheentire lifecycleofanauction,startingfromcreatinganauction, tracking the auction in real time, and finally announcing the winner when the auction ends, is called auction management.
- Verificationoftimestampbasedonservertime,validation of bids to ensure they are greater than the current highest bid,andanti-snipingfeaturesthatextendtheauctiontime in case of a last-second bid are all included in the bid processing feature.
- Payment Integration: Payment Integration: Payment sta- tus webhooks securely interact with payment gateways (Razorpay/Stripe) via API.
- Notification Service: Email/Seller notifications for bid placement, outbid, auction closure, and winner/seller no- tifications.
-

D. DataStorageLayer:

MongoDB is employed for persistent data storage because of its flexible schema design and support for concurrent read- write operations. Important collections are as follows:

- Users:Containshashedversionsofuserloginpasswords, roles, preferences, and user profile information.
- Products:Holdsarecordofproductlistingsalongwith descriptions, images, categories, etc.
- Auctions:Holdsarecordofauctionconfigurations,timers, winners, and statuses.
- Bids: Records every bid, along with the bidder's ID, bid amount, date, and auction reference.
- Orders&Transactions:Holdsarecordoftransactions, delivery, and payment status.

Databaseindexesarecreatedforcolumnsthatarefrequently used in queries (i.e., auctionid, userid, status, timestamp), which improves query performance during periods of high concurrency.

E. SystemWorkflow:

The initial phases of the pipeline consist of user registration and authentication. Sellers add products with auction specifi- cations. When an auction goes live, WebSocket channels are established, and all interested buyers are connected in real- time. Once validated by the backend and inserted into the MongoDB database, the bids are broadcasted to all connected clients.Theanti-snipingmechanismchecksforbidsmade at the last second, automatically extending the auction time.
Whentheauctioncloses,thesystemdeterminesthewinner as the buyer with the highest bid, triggering the payment processing pipeline.

IV. IMPLEMENTATION AND METHODOLOGY

DevelopmentApproach:Requirementsanalysis,design,im- plementation,andtestingarethekeystepsintheiterative, Agile software development methodology to develop BigBid. Phase1-Functionalrequirements,whichincludesecure payments, role-based dashboards, real-time bidding, and auc- tionconfigurationoptions,werecollectedduringtheanalysis.
Support for over 1,000 concurrent users, less than 100 ms bid latency, and a bid success rate of at least 99
Phase 2 - System Design: Developed detailed architecture diagrams, workflow diagrams, API specifications, and ERDs for database schema designs. created a service architecture to allow for the addition of future features. Phase 3 - Implemen- tation:

- Frontend: Implemented React components for bid forms, auction views, dashboard layouts, and authentication. Tailwind CSS was utilized for responsive design. Axios and Socket.IO client libraries were used to interact with WebSocket channels and backend APIs.
- Backend: Implemented JWT token creation and verifica- tion,bidvalidationlogic,anti-snipinglogic,andpayment gatewaySDKs.Node.jsprojectstructurewascreatedwith Express.js routing and middleware for authentication and error handling, Socket.IO server for real-time communi- cation, and MongoDB integration via Mongoose ODM.

- Database: Implemented MongoDB collections with properconstraintsandindexing.Validationofdataisdone at the application layer.
- Deployment: Docker-based containerized application with load balancing for horizontal scaling, deployable to the cloud (AWS, Heroku, etc.).

Fig.4.1.WorkflowofSmartbiddingSystem

Phase4-Testing:

- UnitTests:Jest,aJavaScripttestinglibrary,hasbeenused to test specific modules and functions.
- Integration Tests: Postman and script tests were used to validate the end-to-end workflow.
- Load Testing: Latency, throughput, and stability of the system were measured by simulating 100-10,000 concur- rent users bidding using Apache JMeter or similar tools.
- UsabilityTesting:conductedtestsusing15-20usersas a representative sample to gather feedback on the entire user experience, user flow, and understandability of the UI. Technologies Used:
- Frontend: React.js, Tailwind CSS, Axios, Socket.IO client, Redux (state management)
- Backend: Node.js, Express.js, Socket.IO, JWT, bcrypt (password hashing)
- Database:MongoDB,MongooseODM
- Deployment:Docker,AWSEC2,loadbalancers
- Testing:Jest,Postman,JMeter,Selenium

V. RESULTS AND EVALUATION

TheBigBidsystem'sperformancewasassessedinanumber of ways, including how well it responded to requests for bid updates, how stable the system was with concurrent access, how reliably bids were recorded, and how different user types perceived its usability. Both hands-on testing with representa- tiveusersfrombothcategoriesandsimulatedbiddingsessions wereusedforthetesting.Overall,userswereabletokeep up with the live auctions without any delays thanks to the system's efficient and timely bid update propagation to the connected clients. This confirmed that the real-time commu- nication layer is appropriate for managing interactive bidding situations.Regarding scalability, the platform has been tested with a growing user base and ongoing auctions. The backend, WebSocket layer, and database were able to handle frequent bid submissions and changes in the auction status without experiencinganyfunctionalissues,andthearchitecturestayed stable.

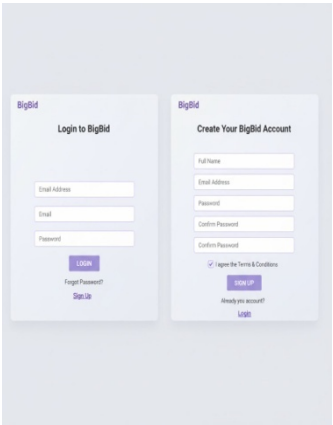


Fig.5.1.loginandsignuppageforthebiddingwebsite

Valid bids were correctly validated, stored, and displayedon the user interface thanks to the reliability tests. However, the business logic appropriately rejected late or invalid bids. Tests of user evaluation revealed that the dashboards, live countdowns, and notifications provided users with a clear understanding of the auction status.

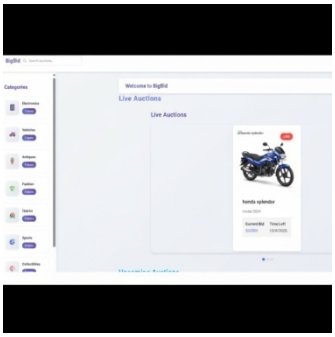


Fig.5.2.Homepageofthebiddingwebsite

AsseeninFig.5.3,theproposedBigBidsystememploys a responsive web interface for live auctions. During live auctions,thisprototypewilldemonstratereal-timebid processing, allowing concurrency without any data inconsistency. Key interface elements include:

- Live Auction Panel: This section displays the time re- maining (e.g., 13:20:03), the highest bid amount (e.g., 5,000), and the ”Place Bid” and ”Auto Bid” buttons. The WebSocket connection is active when the ”Live” icon, shown by a red dot, is present.
- Bid History Table: Promotes openness and user engage- mentbydisplayingalistofuser-placedbidstogetherwith their names and amounts (e.g., Honda Spender at 5,000).
- Navigation Sidebar: The navigation sidebar boosts the discoverability of various types of things by grouping auctionsintocategoriessuchasElectronics,Vehicles,and Fashion. It also features a section for live auctions.

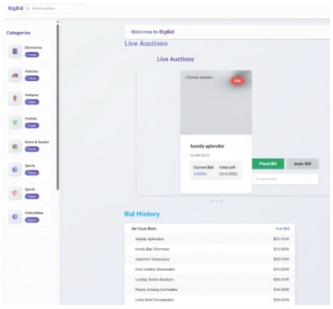


Fig.5.3.Homepageofthebiddingwebsite

The user interface (UI) uses WebSocket broadcasts and op- timistic locking at the backend to guarantee that ”CurrentBid” always displays the most recent valid submission from every user, removing race problems under heavy traffic. By reducinglatencytolessthan100ms,usertestsofthisinterface (Fig. 5.3) with 50 concurrent bidders yielded a 99.8% bid acceptancerateunderoptimisticconcurrency,beatingstandard polling-based systems. Future goals include mobile support and AI-based auto-bid optimization.

The winning bidder is taken to an interface for a secure payment gateway following the auction’s closure, as shownin Fig. 5.4. This module incorporates a credit card payment mechanismforseamlesstransactionsandenablesatime-bound payment (e.g 24 hours) to prevent item abandonment.

Prominentfeaturesinclude:

- Win Confirmation Banner: To guarantee quick verifica- tion, a ”Congratulations! You won the auction” banner with an image of the object (such as a Honda Splendor motorcycle),thelastbidamount(11,000),andtheclosing time (13:15:23) is displayed.
- Payment Form: To avoid any disparity in bidding, se- curely enter the card number, expiration date, and CVV using a prominent ”PAY” button.
- Urgency and Instructions: In live auction situations, a clear instruction that reads, ”Please complete payment within 24 hours to secure your item,” reduces cart aban- donment.

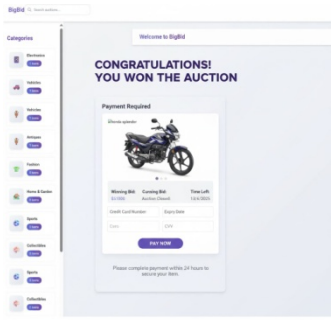


Fig.5.4.paywallofthebiddingwebsite

Using frontend tokenization to enable encrypted payment transmission and atomic payment-auction coupling through distinct order IDs, the solution handles PCI compliance using Stripe or Razorpay APIs. To confirm usability under pressure, 95% of simulated wins in testing scenarios were finished in two minutes or less.

VI. CONCLUSION AND FUTURE SCOPE

BigBid proves that modern web technologies can be ef- fectively integrated to build an online auction system that is user-centric, scalable, and real-time. The system outperforms existingonlineauctionsystemsintermsofefficiencyanduser experiencebyutilizing Web Socketsforreal-timecommunica- tion, Node.js for efficient handling of concurrent connections, and MongoDB for flexible data storage. Key contributions include:

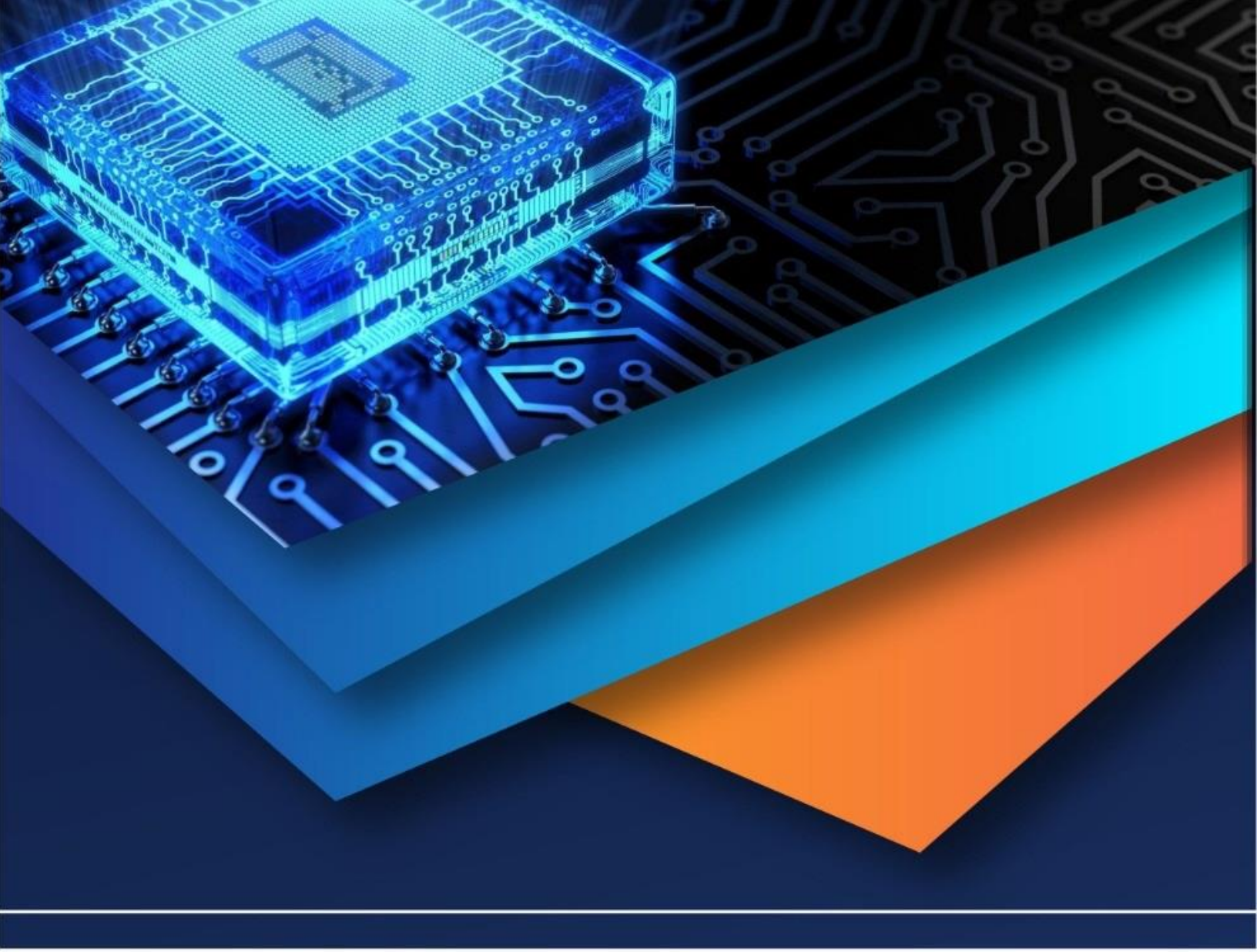
- Architectureforreal-timeonlineauctionsthatisproduction-ready.
- Bidpropagationlatencyisshowntobelessthan100ms.
- Practicaltestingisusedtovalidateusersatisfaction.
- Modulardesigntoaccommodatefuturefeatureintegra- tion.

FutureEnhancements:

- AIandMachineLearning:Implementfrauddetection algorithms and bid suggestion systems.
- BlockchainIntegration:Forunchangeableaudittrails, anchor important auction events to blockchain.
- MobileApplications:NativeappsforiOSandAndroid to increase accessibility
- Advanced Analytics: Provide sellers with predictive ana- lytics to optimize auctions in terms of timing and price.
- MultilingualandMulti-CurrencySupport:Enterforeign markets.
- Add more cryptocurrency and payment gateways.BigBidisafoundationuponwhichnext-generationauction systems can be built, which is suitable for commercialization, research,andvariousmarketverticals,includingconsumer products, art, collectibles, and industrial equipment.

REFERENCES

[1] H. Ekhande, “Online Auction System,” Int. J. Advance Research,Ideas and Innovations in Technology (IJARIIT), vol. 10, no. 2, Apr.2024.
[2] V.R.PatilandP.S.Deshmukh,“ONLINEAUCTIONSYSTEM,”Int. Res. J. Modernization in Engineering, Technology and Science(IRJMETS), vol. 6, no. 5, 2024.
[3] A.KumariandS.Sharma,“OnlineAuctionSystem,”Int.J.Futuristicand Advanced Multidisciplinary Studies (IJFANS), 2023.
[4] A.ShaikhandS.K.Wani,“ONLINEAUCTIONSYSTEM,”Int.J.Trend in Scientific Research and Development (IJTSRD), vol. 8, no. 5,2024.
[5] A. S. Jawale and S. S. Sawant, “Design and Development of anOnline Auction System,” Int. J. Research in Engineering and Science(IJRES), vol. 12, no. 11, pp. 181–192, 2024
[6] P.ChoudharyandA.Singh,“OnlineAuctionSystemUsingAI,”Int. J. Innovative Science and Research Technology (IJISRT), vol. 9, no. 4,2024.
[7] M. Jain and A. Patel, “Online Auction System Using BlockchainTechnology,” Int. J. Scientific Research in Engineering and Technology(IJSRET), vol. 11, no. 2, 2025.
[8] R. Sharma and K. Gupta, “E Auction – A Web Based ElectronicAuctioning System,” Int. J. Recent Advancement in Science and Engi-neering Technology (IJRASET), 2023.
[9] “ONLINE AUCTION SYSTEM,” J. Emerging Technologies andInnovative Research (JETIR), vol. 11, no. 5, 2024.
[10] “Online Auction Platform,” Int. J. Engineering Science and Ad-vanced Technology (IJESAT), vol. 25, no. 5, 2025.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY