RESEARCH ARTICLE OPEN ACCESS

Smart Online Auction System Using Rule-Based Decision Mechanisms

Jitesh Jaysingh Yadav*, Manoj Singh**
*(Department of Computer Science, S.I.E.S College of Arts, Science, and Commerce (Empowered Autonomous), Mumbai, Maharashtra, India Email: jyadav20101@gmail.com)
**(Department of Computer Science, S.I.E.S College of Arts, Science, and Commerce (Empowered Autonomous), Mumbai, Maharashtra, India Email: siescsdept@gmail.com)

----------**********----------

Abstract:

This paper presents Auctify, a web-based online auction system developed using Flask and SQLite. The system enables users to create auctions, request participation, and place bids with near real-time updates. A key feature of the system is a rule-based decision mechanism that ensures fair bidding through bid validation, participation control, and anti-sniping logic. The anti-sniping mechanism automatically extends the auction duration when bids are placed in the final seconds, preventing unfair last-moment wins.

The system uses AJAX-based polling to provide near real-time updates for bid prices, user status, and auction activity. Experimental evaluation demonstrates that the system performs reliably under multiple-user scenarios while maintaining data consistency and fairness. The proposed solution provides a lightweight yet effective approach for implementing real-time auction platforms using simple web technologies.

Keywords — **Online Auction, Real-Time Bidding, Flask, Web Application, SQLite**

----------**********----------

I. INTRODUCTION

The Online auctions are widely used today in many platforms where users can buy and sell items through bidding. These systems allow multiple users to compete by placing bids, and the highest bidder wins the item.

However, many simple auction systems do not handle real-time updates properly and may allow unfair practices like last-second bidding (sniping).

In this project, we developed Auctify, a web-based auction system that solves these problems by using real-time updates and an anti-sniping feature. The system is designed to be simple and easy to use while still including all the important functionalities of an auction platform.

II. RELATED WORK

Many online auction systems have been developed using different technologies. Large platforms use advanced systems to handle real-time bidding and a large number of users.

Some systems use complex technologies like distributed servers and machine learning models. However, these are not always required for small-scale applications.

Simple web frameworks like Flask can also be used to build efficient auction systems. These systems focus more on core functionality such as bidding, validation, and user management.

This project follows a simple and practical approach while still including useful features like real-time updates and anti-sniping.

[1] COMPARISON OF SYSTEMS

FEATURE	EXISTING SYSTEMS	AUCTIFY
ANTI-SNIPING MECHANISM	NOT COMMONLY IMPLEMENTED	IMPLEMENTED
PARTICIPATION APPROVAL SYSTEM	NOT AVAILABLE	AVAILABLE
RULE-BASED BID VALIDATION	LIMITED	STRONG
REAL-TIME UPDATES	PARTIAL	NEAR REAL-TIME (AJAX-BASED)

A. *III. METHODOLOGY*

B. *A. System Overview*

C. *The system works in the following steps:*
- *User registers and logs in*
- *User creates an auction*
- *Other users send requests to participate*
- *Approved users place bids*
- *System updates price in real time*
- *Auction ends and winner is decided*

D. *B. Technologies Used*

E. • *Flask (Python) for backend*
- *SQLite for database*
- *HTML, CSS, JavaScript for frontend*
- *AJAX for real-time updates*

F. *C. Database Design*

G. *The system uses four main tables:*
- *Users*
- *Auctions*
- *Bids*
- *Requests*

H. *These tables are connected using IDs such as auction_id.*

I. *D. Auction Logic*

J. *Auctions are divided into three types:*
- *Live*
- *Upcoming*
- *Ended*

K. *This is based on the start and end time of the auction.*

L. *E. Bid Validation*

M. *Before accepting a bid, the system checks:*
- *The bid is higher than the current price*
- *The bid follows the minimum increment rule*
- *The user is allowed to participate*

N. *F. Anti-Sniping Feature*

O. *To make the system fair, an anti-sniping feature is used.*
If a bid is placed in the last few seconds of the auction, the auction time is extended. This gives other users a fair chance to respond.

P. *G. Real-Time Updates*

Q. *The system uses AJAX to update data every few seconds. It helps in:*
- *Updating current price*
- *Showing top bids*
- *Checking if a user is outbid*
- *Displaying bid history*

IV. RULE-BASED DECISION MECHANISM

The system uses a rule-based approach to control auction behavior and ensure fairness. The key decision rules implemented are as follows:
• If the bid amount is less than the current price plus minimum increment, the bid is rejected.
• If the bid exceeds a predefined maximum threshold, it is rejected.
• If a user has not been approved by the auction owner, bidding is not allowed.
• If a bid is placed within the last few seconds, the auction end time is extended automatically.
• If the auction time has expired, further bids are blocked.
These rules ensure controlled participation and prevent unfair practices without requiring complex machine learning models.

V. SYSTEM ARCHITECTURE

The system architecture follows a three-tier model consisting of the client interface, backend server, and database layer. The client interacts with the Flask-based backend through

HTTP requests, while the backend processes auction logic, bid validation, and user authentication. The backend communicates with the SQLite database for storing and retrieving auction-related data.

This layered architecture ensures modularity, scalability, and efficient data management.

VI. EXPERIMENTAL RESULTS

The system was tested using different users and auction scenarios.

Observations:

- Average bid update latency: 1–2 seconds
- System handled concurrent bidding without data inconsistency
- Anti-sniping feature successfully extended auctions
- No invalid bids were accepted

[2] Sample Output

Metric	Example Result
Current Price	₹ 50,000
Status	Live
Top Bidder	user@gmail.com

[3] A. Evaluation Discussion

Advantages:

- Easy to use
- Real-time functionality
- Fair bidding system
- Simple implementation

Limitations:

- Uses polling instead of advanced real-time methods
- Limited scalability due to SQLite
- Basic user interface

VII. CONCLUSION

This paper presented Auctify, an online auction system with real-time bidding and an anti-sniping feature. The system is simple but effective and provides all the necessary features for a fair auction.

It demonstrates that rule-based decision mechanisms can ensure fairness and reliability without requiring complex computational models.

VIII. FUTURE WORK

Future improvements can include:

- Integration of WebSockets for real-time updates
- Secure payment gateway integration
- Use of scalable databases such as PostgreSQL
- Enhanced UI/UX design
- Mobile application development

REFERENCES

[1] eBay Inc., "Online Auction Platform Overview," Available: https://www.ebay.com

[2] M. Grinberg, *Flask Web Development*, O'Reilly Media, 2018

[3] Mozilla Developer Network, "AJAX Guide," Available: https://developer.mozilla.org

[4] SQLite Documentation, Available: https://www.sqlite.org/docs.html

[5] P. Klemperer, "Auction Theory: A Guide," *Journal of Economic Surveys*, 1999

