2016 IEEE 14th Intl Conf on Dependable, Autonomic and Secure Computing, 14th Intl Conf on Pervasive Intelligence and Computing, 2nd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress

uAuction: Analysis, Design and Implementation of a Secure Online Auction System

Nazia Majadi
School of ICT
Griffith University
Queensland, Australia
Email: nazia.majadi@griffithuni.edu.au

Jarrod Trevathan
School of ICT
Griffith University
Queensland, Australia
Email: j.trevathan@griffith.edu.au

Neil Bergmann
School of ITEE
University of Queensland
Queensland, Australia
Email: bergmann@itee.uq.edu.au

***Abstract*—Online auctions are now an immensely popular component of the electronic marketplace. However, there are many fraudulent buying/selling behaviours that can occur during an auction (e.g., shill bidding, bid shielding, etc.). While researchers are proposing methods for combating such fraud, it is extremely difficult to test how effective these countermeasures are. This is primarily due to it being unethical to engage in fraudulent behaviour just for the purpose of testing countermeasures. Furthermore, there is limited commercial auction data available due to the sensitivities of an online auctioneer being willing to admit that fraud has, or is occurring. In order to test fraud countermeasures in a controlled environment, we have created our own online auction server for conducting auction-related research. This paper presents our experiences with designing and implementing our own online auction system which we call *uAuction*. At present, there is limited useful literature on auction system design. We present an analysis and design of the auction system by employing Unified Modeling Language (UML) to show the architectural model, subsystems, use cases, activity workflows, class diagram, user interfaces, and system sequence diagrams. Our auction model is grounded in object-oriented techniques and is open source so that other researchers can expand upon our approach.**

***Keywords*—Auction fraud; Domain model class diagram; Design class diagram; Shill bidding; System sequence diagram; UML.**

I. Introduction

Online auction sites, such as eBay and Yahoo! Auctions, are experiencing a dramatic increase in their popularity. The number of auction items hosted by eBay has increased from 110 million to approximately 266 million between July 2010 and September 2014 [8], [15]. A seller lists an item online for a set amount of time and buyers must place a bid higher than the last bid in order to purchase. Online auctions have removed the physical and logistical limitations of geographic proximity, time to organise, physical space, and small target audience.

However, the online environment creates many unique opportunities for people to cheat. Auction fraud can occur prior to an auction (e.g., misrepresentation of items, selling of black market goods, and triangulation), during an auction (e.g., shill bidding), or after the auction terminates (e.g., buyer does not pay for the item). Much research has been conducted around pre and post auction fraud [5], [11]. However, in-auction fraud is typically the hardest to develop effective countermeasures for as it deals with human behaviours and strategies that are somewhat unclear.

Shill bidding is the practice whereby a seller bids on his/her own auction in order to artificially increase the price that the winning bidder must pay. While it is understood that this is a problem, there are multiple strategies a shill bidder can engage in. As such there is much confusion over what actually constitutes shill bidding and how to effectively detect and prevent shill bidding. An even more significant problem is how to test the effectiveness of in-auction fraud counter measure proposals.

A major factor in the difficulty of testing in-auction fraud counter measures is the lack of available commercial online auction data. Online auctioneers do not share their auction data, commonly citing privacy reasons. However, it is more likely due to fear of damage to their public image should it be discovered that fraud is rampant in their auctions. Another significant issue with testing fraud counter measures is due to ethics/legality. For example, it is actually illegal for a researcher to engage in shill bidding in commercial online auctions primarily for the purpose of testing fraud counter measures. Due to these two major impediments, an alternative proposal for in-auction fraud testing must be examined.

We were driven to create our own online auction system due to there being limited useful literature available on auction software design. Moreover, the existing auction software literature are typcially not based on Unified Modeling Language (UML) [6], [7], [14], [16]. While there are vendors who sell auction software [2], such software is expensive and cannot be customised for our research requirements. This paper presents an analysis and design of our auction system which we call *uAuction*. We employ UML to show the architectural model, subsystems, use cases, domain modeling, activity diagrams, database schema, website navigation, user interface, and system sequence diagrams. *uAuction* is being used to test the effectiveness of our own shill bidding detection and prevention proposal.

This paper is organized as follows: Section II discusses related work on designing online auction system and the motivation for this research. Section III describes the major participants of an online auction and auction formats. Section

978-1-5090-4065-0/16 $31.00 © 2016 IEEE
DOI 10.1109/DASC-PICom-DataCom-CyberSciTec.2016.66

IV presents an analysis of *uAuction* using UML. Section V describes the design of *uAuction*. Section VI shows testing approaches available for detecting and preventing fraud. Finally Section VII provides the concluding remarks with future work.

II. Related Work and Problem Motivation

This section describes some existing online auction systems and the motivation for designing and implementing *uAuction*.

A. Existing Auction Systems

Wurmann et al. [13] provide a software design for online English auctions that supports both software and human agents. Their proposed auction server named the Michigan Internet AuctionBot provides for flexible specification of auctions considering different parameters so that agent researchers can explore the design space of auction mechanisms. However, the authors do not show how they developed their auction system. Furthermore, the AuctionBot has been decommissioned since the early 2000s.

Kumar and Feldman [12] present a software architecture and describe the various processes that comprise the auction application. They present an auction system which implements a collection of auction types. The key features of the underlying objects, processes, and interaction models are described along with how an auction relates to some commonly used trading models such as brokerages, two party negotiations, and competitive bid-based procurement. Moreover, they describe issues such as delay, security, and collaboration. However the paper does not use UML for analysing and designing the proposed auction system. Furthermore, the paper is not research-oriented.

Rumpe et al. [1] describe an architecture for developing a web-based, real-time online auction system. The paper also discusses the functional and technical requirements of developing an auction system combining standard software and self-developed components. Although the authors use UML components to a large extent for implementing their auction system, their UML diagrams are incomplete and do not strictly follow the UML standards.

Sheldon et al. [4] design a web-based auction system using UML and component-based programming. They use UML in a limited capacity in an attempt to model various aspects of the auction system. Moreover, intelligent agents are used for helping bidders participate in the auctions without always being present to view the status of the auction. The authors describe a case study pointing to the best practices in designing and building a web-based auction system. However, the UML diagrams are unclear and are of limited use for helping others to understand how to implement their own auction system.

Trevathan et al. [10] design an online auction server for facilitating auction research (referred to as the Research Auction Server). The paper describes the design of online auction software by presenting a basic online model and addresses the main auctioning processes, web navigation, preliminary security, the database schema, and transaction and timing issues. The authors also show how their proposed model can interact with software bidding agents. Their auction design is developed based on object-oriented techniques and is an open source tool. However, the authors do not present any UML diagrams for describing their auction system.

Ren [3] developed an online auction system based on a university campus network for providing ecommerce facilities to students. The author designed the auction system using UML. The paper describes use case diagrams, activity diagrams, sequence diagrams, class diagrams, and deployment diagrams for designing their proposed online auction scheme. But the UML diagrams are incomplete and do not strictly adhere to UML standards. Furthermore, this paper is not research-oriented.

B. Problem Motivation

Considering the limitations of the existing literature, we have designed *uAuction*. Our main target research is to detect shill bidding in real-time. To do this, we need to develop our own auction system as it is illegal/unethical to engage in shill bidding using commercial auction sites (e.g., eBay, TradeMe, etc.) for testing purposes. This paper's aim is to describe our knowledge and experiences with developing a web-based auction model.

The reasons for developing this auction system are as follows:

- To fully understand online auction system requirements;
- To gain experience with administering an auction server and participating in online auctions;
- To enable testing of fraud detection/prevention techniques; and
- To educate auction users about fraud/auctioning behaviours.

III. Participants and the Online Auction Format

A. Online Auction Participants/Stakeholders

There are three main stakeholders in an online auction:

- **Seller** – A seller lists an item (or collection of items) for sale. The seller is typically after the highest price possible for the item(s).
- **Bidder** – A bidder submits a bid for an item listed by the seller. The amount the bidder bids is an indication of what the bidder is willing to pay for the item being auctioned. The bidder is typically after the lowest price possible in order to win.
- **Auctioneer** – The auctioneer is responsible for hosting the auction, providing the resources required for the auction, and conducting the auction proceedings according to the auction rules. The auctioneer is usually paid a listing fee by the seller. In some cases, the auctioneer may receive a commission based on the winning price. In this case, the auctioneer will typically want the item to sell for the highest price possible.

B. Online Auction Format

There are different types of auctions such as English, Vickrey, Dutch, Continuous Double auction, etc. [2], [12]. The English auction is an *open bid*, *ascending-price* auction in which bidders place competing bids against other bidders in order to purchase the auctioned item. When a given time expires, the highest bidder wins the auction and must pay an amount equal to the winning bid. This type of auction is commonly used in real estate. Many online auctions are modelled on the English auction, except that an auction finishes at a predetermined closing time.

TABLE I: Bid history of an online auction

Bid ID	Bidder	Bid Amount	Bid Time
8	David	$65	Jan-31-16 09:18:16
7	Jessica	$59	Jan-31-16 09:17:40
6	William	$55	Jan-30-16 09:12:58
5	Shannon	$51	Jan-28-16 21:54:09
4	Jessica	$50	Jan-28-16 20:35:05
3	Robert	$45	Jan-28-16 21:05:14
2	Jessica	$42	Jan-28-16 21:04:44
1	Shannon	$40	Jan-28-16 21:04:08

In general, the format of a bid in an online auction is: <*bidder*, *bid amount*, *bid time*>. A bidder can request for a price quote of an auctioned item from the Auctioneer at any time. The price quote comprises the bid information of the current highest bid. The auction *bid history* is a list of all bids that have been placed by the participating bidders in the auction (see Table I). Here the winning bid is $65.

An online auction can have three additional parameters:

- *Starting price* – the minimum price at which the bidding must start;
- *Reserve price* – the minimum price the seller will accept. If the final price is below the reserve price, the result of the auction is void; and
- *Minimum bid increment* – the minimal amount required to outbid the current highest bid.

There are many variations available in online auctions. For example, bidders can alter/cancel bids, confirm their placed bids, auction multiple goods, *Buy It Now* auctions, etc. However, the auction model in this paper only focuses on English auctions with a predetermined closing time.

IV. Systems Analysis

This section analyses the requirements for uAuction.

A. Software Subsystems

Software subsystems provide an intuitive way of visualising, understanding and analysing the major functional requirements. Table II shows the *uAuction*'s subsystems.

B. Use Cases

The use case consists of a set of possible sequences of interactions between systems and users in a particular environment for achieving a particular goal. The use cases for *uAuction*'s subsystems are as follows:

TABLE II: *uAuction* subsystems

Subsystem	Users/Actors	Description
User account management	Seller, Bidder, Auctioneer	A seller/bidder can register himself and can also update or view his details. The auctioneer can view/edit a user's details and delete users.
Auction management	Seller, Auctioneer	A seller lists products for auction.The auctioneer approves or disapproves an auction listing, and acknowledges the corresponding seller. He is also responsible for adding product categories.
Auction searching	Bidder	A bidder searches products on auction.
Bidding	Seller, Bidder, Auctioneer	Bidders submit bids in an auction and the auctioneer notifies the bidder who wins the auction.
Payment	Seller, Bidder, Auctioneer	A seller pays the auction listing fee to the auctioneer. The winning bidder has to complete transactions for paying the seller. The seller sends the product to the winner upon receiving the payment.
Fraud detection	Seller, Bidder, Auctioneer	The auctioneer can detect auction fraud. The bidder and seller are provided with information to determine whether fraud has occurred.

- **User account management:** Fig. 1 presents the use case diagram that deals with the issues relating to the authentication of trading companies or users (sellers/bidders), creation of a profile for each user that reflects his interest in different kinds of products, location details, and membership information. A seller/bidder can also update or view his details. The auctioneer can view the details of any user and delete any user account.

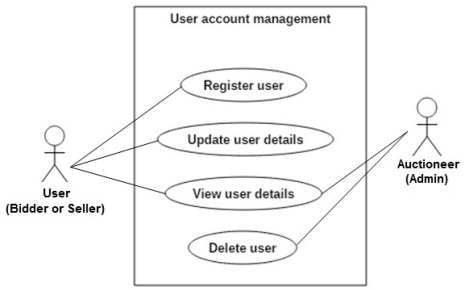

Fig. 1: User account management use case

- **Auction management:** A seller sets up the auction listing of products. This step deals with describing parameters such as price, reserve value, delivery dates, shipping details, payment type, starting and ending date and time of the auction, auction status, etc. The auctioneer approves the auction listing. After getting approval, the corresponding seller will be able to process his auction. Moreover, the auctioneer sets up the auction rules and guidelines. (see Fig. 2)
- **Auction Searching:** Registered bidders are able to search for specific products being auctioned (see Fig. 3).
- **Bidding:** Fig. 4 shows the use case diagram of bidding subsystem that views the list of auctions and handles the collection of placing bids from different registered bidders and implements the bid control rules of the auc-

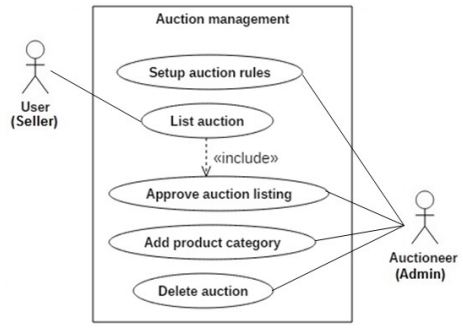

Fig. 2: Auction management use case

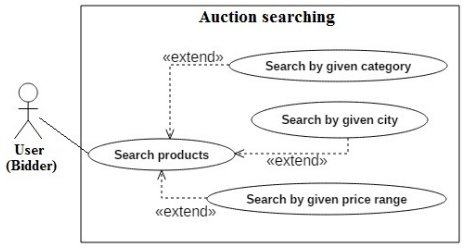

Fig. 3: Searching use case

tion (e.g., minimum bid, bid increment, deposits required with bids). The auctioneer terminates an auction and then determines the winner according to the auction rules.

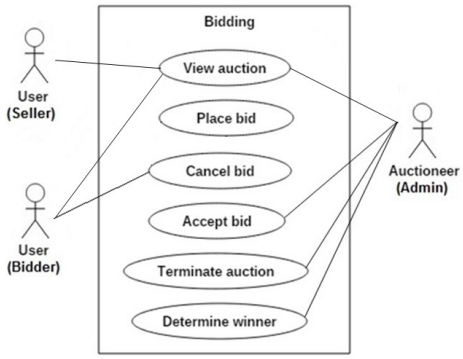

Fig. 4: Bidding use case

- **Payment:** This subsystem handles the payment of auction listing fees to the auctioneer. A bidder who wins an auction sends payment to the corresponding seller. At the same time, the seller transfers the auctioned products to the bidder (see Fig. 5).
- **Fraud detection:** Fig. 6 illustrates the fraud detection subsystem. This subsystem monitors an auction for signs of fraudulent bidding behaviour. Bidders or sellers can provide information or are notified if fraud is suspected.

C. Domain Modeling

The domain model class diagram represents the types of objects that exist in the system and shows the associations among them. It also shows the attributes of a class and the constraints that apply to the way the objects are connected.

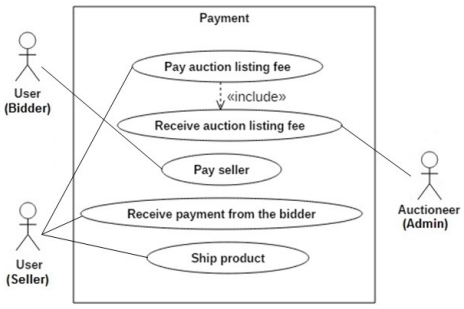

Fig. 5: Payment use case

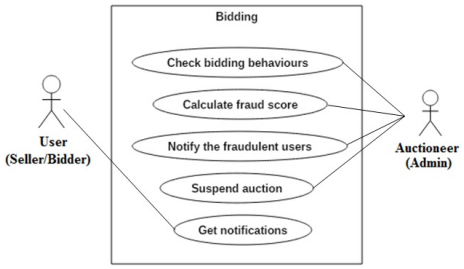

Fig. 6: Fraud detection use case

Fig. 7 shows *uAuction*'s domain model class diagram. There are three classes: *Bidder*, *Auction* and *Bid*.

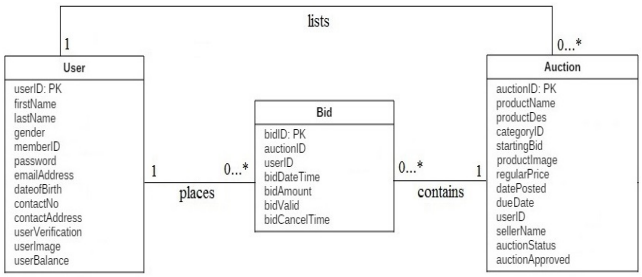

Fig. 7: Domain model of an online auction system

The *User* class contains the attributes related to the user. A user becomes a valid user after completing a registration process by providing user details. A valid user can list products for auction through the association with the *Auction* class and also can make a bid through the association with the *Bid* class. The *Bid* class describes the bid amount, bid time and the user making the bid. There are some variations available in online auctions such as the ability to cancel a submitted bid. This is typically allowed when a bidder has accidently entered the wrong bid amount. In the case of cancelling a bid, the bid could simply be deleted from the database. Nonetheless, a cancelled bid should be recorded into the database due to security issues. This can be done by including *bidValid* and *bidCancelTime* attributes to the *Bid* class of the class diagram.

D. Activity Diagram

An activity diagram is graphical representation of work flows to describe the dynamic aspects of a system. While there are many such diagrams related to *uAuction*, for space

constraints will only provide the most significant activity diagram.

Fig. 8 shows the activity diagram for bid submission. When a bidder submits a bid for a particular auction, his placed bid is compared to the current highest bid or last bid of the auction. If the bidding amount does not meet the minimal required amount to outbid the existing highest bid, then it is discarded and the bidder is asked to resubmit a new bid. On the other hand, the bid value meets the required amount it is timestamped by the auctioneer and entered into the database.

In case of first bidding in an auction, the bid is compared to the starting price of the auction. If the bid is higher than the starting price, then the bid value is recorded by the auctioneer and stored in to the database. Otherwise, the auctioneer rejects the submitted bid and notifies the bidder to place a new bid again.

E. System Sequence Diagram

A sequence diagram depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Fig. 9 represents the sequence diagram of bid submission in our proposed auction system.

V. System Design

The design of *uAuction* is discussed in this section. Design class diagram and *uAuction* design interfaces are also discussed in details.

A. Software Model

Fig. 10 presents a high level software model for performing online auctions in *uAuction*. There are two main parties: a user (bidder or seller), and an auctioneer. A communication link is used to join the two parties.

The participants of an online auction system are described as follows:

1) **User:** A user can be either a bidder or a seller. A bidder interacts with an auctioneer using an HTML browser. A two-way communication link is used for communicating with the auctioneer to place a bid or get information such as auction status from the auctioneer.
   A seller interacts with an auctioneer using an HTML browser. Using a two-way communication, the seller can post one or more items with the details of the items to the auctioneer for auctioning, as well as, can get notification from the auctioneer about the winner of the auction item after the end of the auction.
2) **Auctioneer:** The auctioneer runs a web server (e.g., MySQL server) and a scripting language like PHP. The auctioneer is responsible for taking information from the bidders and the sellers. The auctioneer provides registration service, log service, access control service, data persistence service, etc.

The entire auction is database driven. All state information (e.g., bids, timing, bidding amount, cancellation of bids, etc). about the auction is stored into the database. The transaction of the database starts when a bidder submits a bid or requests a price quote. Using the scripting language, the database generates dynamic web pages according to the bidder activity.

B. Interface Design

In this section we first discuss the usability issues in auction software from the auctioneers perspective. As the software becomes more flexible, allowing a wide variety of auction styles to be used, the auctioneers task of specifying the complete set of rules for an auction becomes more arduous.

1) The User Interface: There are two types of user interface available in *uAuction*. Both users have to register into *uAuction* for performing selling and bidding activities. Our auction site, *uAuction* puts a user(seller/bidder) on the Welcome page from where a registered user can authenticate him to the site and initiate a secure session (login). An unregistered user will get the opportunity to fill in a registration form that may be processed online or off-line.

Other additional features of these two user interfaces are discussed in this section.

- **The Seller Interface:** A seller lists auction products by providing details of the product (e.g., the name of the product, starting price, regular price, product category, product description, image of the product, etc). After listing an auction, a seller waits for approval from the auctioneer. The listing products will be ready for auctioning when the seller gets the notification of approval. Fig. 11 shows a seller interface of *uAuction*.
- **The Bidder Interface:** Fig. 12 shows how the bidders navigate the auction web site in the auction implementation. Each bubble shows a web page and arcs from one page to another indicate that a hot link is available from the first page to the second. The two-way communication link indicates that it is possible to browse the first page to the second and also possible to go back to the first page from the second.
  After registration, a bidder can browse through or search the products in the auction site, which will possibly result in a product being selected, and its description presented to the bidder (see Fig. 13). If the product is on auction, the rules of auction and bid history can be viewed, and bids can be submitted for that product.
  From the home page the bidder can also see a list of all auctions at *uAuction* or a subset of these that are in his personal auction *watchlist*. An auction gets added to a bidder's auction *watchlist* when the bidder explicitly takes an action to do so from certain auction pages, or implicitly when the bidder places his first bid. From lists, all auctions or auction *watchlist*, the bidder can select an auction and access the description of the product being auctioned, see the rules of the auction, or bid on the product.

2) The Auctioneer Interface: Fig. 14 shows an auctioneer interface of *uAuction*.

The auctioneer interface consists of five components:

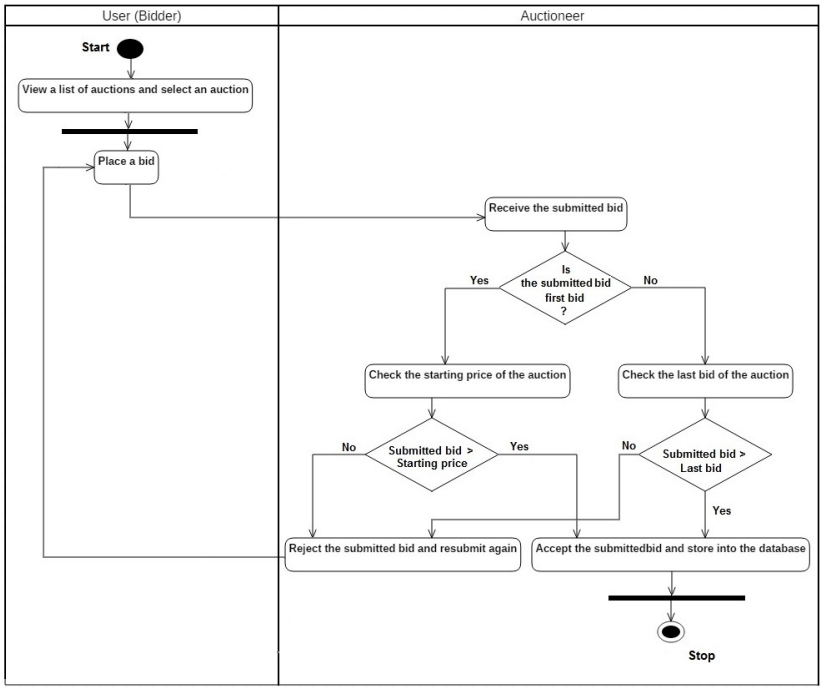

Fig. 8: Activity diagram of bid submission

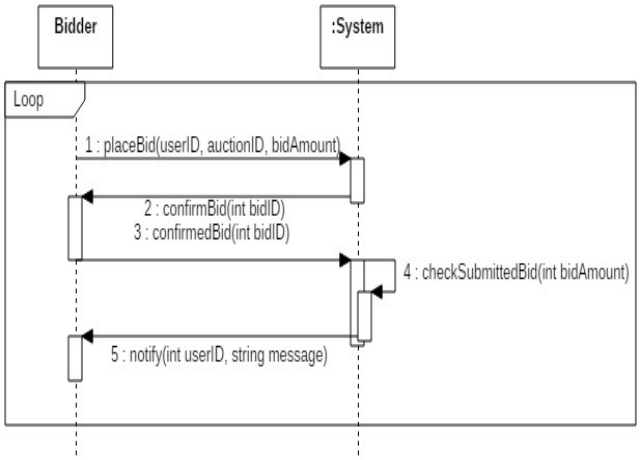

Fig. 9: Sequence diagram of bid submission

(i) Provide approval to auction listings requested by a particular seller after which the product will be ready for auctioning process;
(ii) Allow the auctioneer to view the details of the products and monitor the progress of the various running auction;
(iii) Add new categories of product;
(iv) Notify the seller if his product is not satisfied the auction rules or the bidder if he wins an auction;
(v) Cancel any auction if any violation or fraud activity takes place.

C. Design Class Diagram

The design class diagram for developing our auction system, *uAuction*, is shown in Fig. 15. The diagram shows both the attributes and operations of the associated class. The *User* class consists of operations like, user registration, search auction products, auction listing, place bid, cancel bid, update the details of user, etc.

The domain modelling contains all attributes discovered during analysis activities (see Fig. 7). The design class diagram

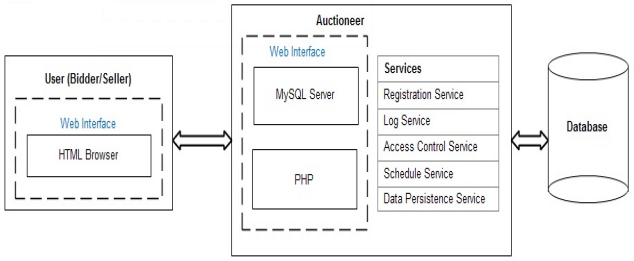

Fig. 10: *uAuction*'s software model

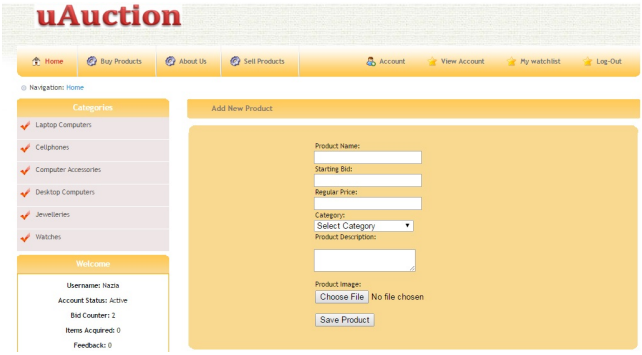

Fig. 11: The seller interface of *uAuction*

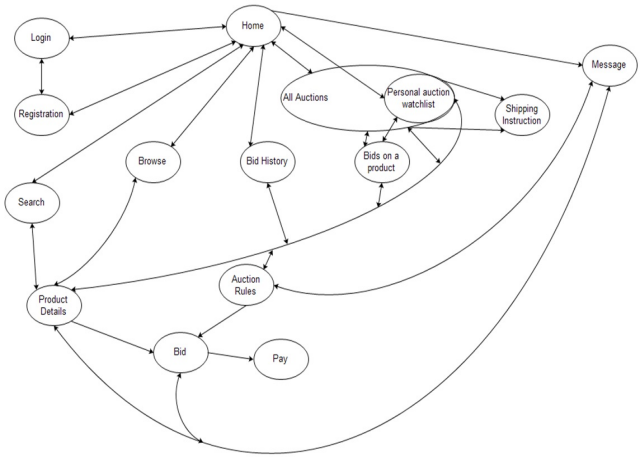

Fig. 12: The navigation process of a bidder

Fig. 13: The bidder interface of *uAuction*

includes more information on attribute types, initial values, and properties. It can also include the operations performed by the class.

VI. Testing Approaches for Fraud Detection/Prevention Methods

As outlined in Section IIB, the motivation for creating uAuction was to have a platform capable of testing our in-auction fraud detection/prevention proposals. Table III shows four types of in-auction fraud [9]. Each type of fraud is aimed at disadvantaging different (or multiple auction) participants. Currently we are focussing on developing mechanisms to detect the presence of shill bidding.

Shill bidding is a very difficult type of in-auction fraud to detect as there are many strategies a shill bidder can engage in. There is also some confusion over what constitutes shill bidding behaviour. Some proposals for shill bidding exist [5]. However, these proposals tend to take a static approach. That is, wait until the auction has terminated and then look for evidence of shill bidding. This means that no recourse can be taken against a shill bidder until an innocent bidder has become a victim. Our current research (being facilitated using uAuction) is to develop a real-time shill detection algorithm.

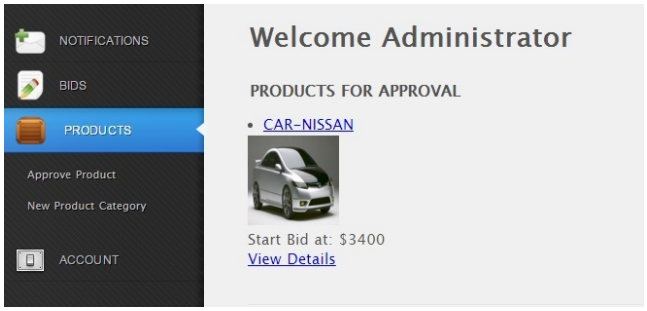

Fig. 14: The Auctioneer Interface of *uAuction*

TABLE III: Details of Fraud Types

Sr. No.	Fraud Type	Description
1	Shill bidding	A seller places bid on his own auctioned item by himself or by involving his friends and relatives to submit bids on his behalf.
2	Bid Shielding	Bidders raise the price of an item up using aliases, which frustrates the legitimate bidders, and then at last moment the high bids are withdrawn to secure a low bid.
3	Bid Sniping	Bid sniping is a practice performed by a bidder to prevent being outbid. A sniper submits a bid during the closing seconds of an auction in order to deny other bidders time to react.
4	Bid Siphoning	Siphoning happens when an outsider observes an auction and offers a similar item to the bidders for a lower price. The siphoner avoids all of the costs essential to conduct an auction, and thus reducing the number of bidders to place bid in the auction of the legitimate seller.

The algorithm looks for signs of shill bidding while an auction is in progress. This means that actions can be taken against a suspected shill bidder prior to an innocent bidder becoming adversely affected. Penalties could range from warning a suspicious user, suspending/terminating an auction, suspending a users account, or enforcing economic restrictions on a users future activities.

As previously mentioned, it is not possible to test our real-time shill detection algorithm using a commercial online auction, nor is there sufficient reliable commercial auction data available. uAuction allows use to conduct various types of tests in order to determine the effectiveness of our shill bidding detection proposal.

Table IV describes possible approaches for detecting/preventing fraud mechanisms.

VII. Conclusion

This paper discussed our experiences in designing an online auction system. Much of the existing auction software literature is dated and is of limited usefulness to researchers. Furthermore, most existing proposals do not adhere to sound UML standards. We provided a simple and elegant design for an online auction system based on UML. We presented an analysis and design for the auction system illustrating key system components using UML diagrams. uAuction is being used to facilitate our research into real-time shill bidding detection. uAuction provides us the ability to conduct various

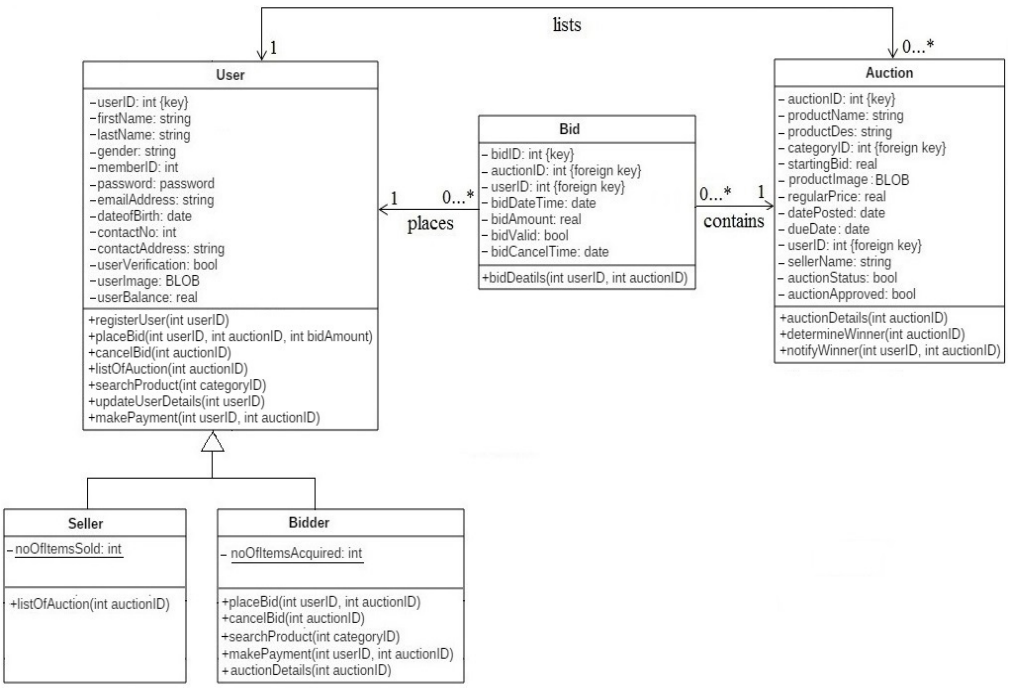

Fig. 15: Design class diagram of *uAuction*

TABLE IV: Testing approaches for detecting/preventing fraud

Sr. No.	Testing Approach		Description
1	Import real/synthetic data		Real/synthetic data can be imported into testing system for detecting/preventing fraud.
2	Simulated auction data	Real users without any money	Real users can register into an auction system that is developed as a testbed and can acquire any items without spending any money through auction processes.
		Real users with real money	Real users can participate in an auction running in a testbed with some real money.
3	Bidding agents		A bidding agent can be used for submitting bids automatically on behalf of the bidder.

types of testing using a combination of human users, simulated auctions and/or synthetic data.

Future work involves further development of the auction system so that it has more functionality and can potentially undertake different types of auctions. We also intend on implementing our real-time shill bidding detection algorithm as part of the fraud detection subsystem. This will then allow us to undertake some of the aforementioned testing approaches in order to help refine our proposed fraud detection techniques.

REFERENCES

[1] B. Rumpe and G. Wimmel, *A framework for realtime online auctions,* in Proceedings of Information Resources Management Association (IRMA) International Conference, pp. 208912, 2001.

[2] *Best Auction software*. Available: http://www.capterra.com/auction-software/. [Accessed: 22-Jan-2016]

[3] C. Ren, *Research and design of online auction system based on the campus network using uml,* in Proceedings of the 2nd Pacific-Asia Conference on Web Mining and Web-based Application(WMWA09), pp. 129–133, 2009.

[4] F. T. Sheldon, K. Jerath, Y. J. Kwon, and Y. W. Baik, *Case study: Implementing a web based auction system using uml and component-based programming,* in Proceedings of the 26th Annual Internationa Conference on Computer Software and Applications Conference (COMPSAC 2002), Vol. l, pp. 211–216, 2002.

[5] F. Dong, S. M. Shatz, and H. Xu, *Combating online in-auction fraud: Clues, techniques and challenges,* Computer Science Review, Vol. 3(4), pp. 245–258, 2009.

[6] H. Gomaa, *Object oriented analysis and modeling for families of systems with uml,* In Software Reuse: Advances in Software Reusability, pp. 89–99, 2000.

[7] H. Gomaa, *Designing concurrent, distributed, and real-time applications with uml,* in Proceedings of the 23rd International Conference on Software Engineering, pp. 737–738, 2001.

[8] Internet Crime Complaint Center, *2014 internet crime report*. Available: https://www.fbi.gov/news/news_blog/2014-ic3-annual-report. [Accessed: 22-Jan-2016]

[9] J. Trevathan and W. Read, *Undesirable and fraudulent behaviour in online auctions,* in Proceedings of International Conference on Security and Cryptograpghy, pp. 450–458, 2006.

[10] J. Trevathan, W. Read, and R. Balingit, *Online auction software fundamentals,* in International Proceedings of Computer Science and Information Technology, Vol. 2, pp. 254–259, 2009.

[11] K. M. Dolan, *Internet auction fraud: the silent victims,* Journal of Economic Crime Management, vol. 2(1), pp. 1–22, 2004.

[12] M. Kumar and Feldman, S. I. Feldman, *Internet auctions,* in Proceedings of the 3rd USENIX Workshop on Electronic Commerce, volume 31, 1998.

[13] P. R. Wurman, M. P. Wellman, and W. E. Walsh, *The michigan internet auctionbot: A configurable auction server for human and software agents,* in Proceedings of the 2nd international conference on Autonomous agents, pp. 301–308, 1998.

[14] P. Allen, *A practical framework for applying uml,* in The Unified Modeling Language. UML98: Beyond the Notation, pp. 419–433, 1999.

[15] Power Sellers Unite, *Auction site count: keep track of the number of listings on auction sites*. Available: http://www.powersellersunite.com/auctionsitewatch.php. [Accessed: 22-Jan-2016]

[16] R. Breu, R. Grosu, E. Huber, B. Rumpe, and W. Schwerin, *Systems, views and models of uml,* in The Unified Modeling Language, pp. 93–108, 1999.