Online Auction Based Book Store

Software Requirements Specification Document

Software Engineering Project (UCS503)

Report Evaluation

Submitted To:

Mrs. Deep Mann

Prepared By:

Nandan 101916066

Nitin 101916063

Rachit 101916046

Prince 101916056

13/06/2021

		Software BID
1. Project Options	1	Design a front-end for Google Search, Google Image Search, and Google Advanced Search
	2	Design a Wikipedia like online encyclopaedia.
	3	**Design an eBay-like e-commerce bidding site for books that allows users to sell and buy books.**
	4	Design a front-end for an email client that makes API calls to send and receive emails.
	5	Design a Twitter-like social network website for making posts and following users.
2. Project Chosen		Design an eBay-like e-commerce bidding site for books that allows users to sell and buy books.
3. Project Experience		Prince: HTML, CSS, Python, SQLite3, Django Nandan: HTML, CSS, Python, SQLite Rachit: HTML, CSS, Python, SQLite, Django Nitin: Python, SQLite, Django
4. Languages Used		HTML, CSS, Python, SQLite3, Framework: Django

TABLE OF CONTENTS

Revision History

Chapter Topic

Revision History

Name	Date	Reason For Changes	Version
Use Case Diagram	22/05/2021	Incorrect Diagram	
Data Flow Diagram	3/06/2021	Incorrect Flow of Data	

1.Introduction

1.1 Purpose

Since the internet has become a popular platform to buy and sell goods, online auctioning services have made their way into most homes. Online auction systems are web-based applications, so the main advantage is that there is no more system compatibility requirement problem. **Our main purpose is to make people invest their time and input to get maximum profit and knowledge about our online bidding system**. Here all types of users can go and analyze the data of different fields and get maximum profit for future investment.

1.2 Document Conventions

Typeface	**Indicates**
Font	Open Sans
Bold	Mainly for headings
Italics	Mainly URLs in References
Blue-Underline	Used for URLs

1.3 Intended Audience and Reading Suggestions

This document is intended for developers, project managers, marketing staff, users, testers, and documentation writers. This SRS contains information related to the Functional requirements and non- functional requirements. You can start reading from the purpose heading and proceed as further with no specific path needed.

1.4 Project Scope

This web application system will be an online auction based book store which consists of seller(vendor), buyer of books. The admin web application will allow the online auction administrator to sell and buy the products.

The **main purpose** of this software is that one can find good products at best prices and no middleman is involved in the transaction. Basically, the goal of this software is to expand it for the worldwide sales in order to achieve the long-term goal.

1.5 References

Scope of the development project

https://www.quora.com/What-are-the-benefits-of-online-bidding-auctions

https://www.researchgate.net/publication/310615150_An_efective_online_auction_system

https://www.iitg.ac.in/gb/papers/auct_paper.pdf
https://ukdiss.com/examples/0410591.php

https://ukdiss.com/examples/0410591.php

2. Overall Description

2.1 Product Perspective

The online auction based book store is an independent system. This system involves two users i.e., buyer and vendor(seller). In this the seller will post the product with the help of images and description of the product. The buyer has to select the product and bid accordingly. And, the bidding will have a specific time limit which will be set by the seller of the product. The buyer with the highest bid, the product will be sold to the bidder.

2.2 Product Features

- Customer has a valid User ID and password to login into the system.
- If there is a new user, he has to register for this auction process.
- User can search the desired book he wants and also in the particular price he wants i.e. - maximum and minimum price.
- Administrators can take a backup of the database for every auction that is happening, periodically.
- All users are authenticated to avail the services.

2.3 User Classes and Characteristics

The **user (seller and buyer)** may have the basic knowledge about computer & internet so that they can use this Online auction-based Book Store. The **admin** should have all the control over the website be it database or the back-end services so that any change made can be accessed with administrator's permission.

2.4 Operating Environment

The web application will run over all Kind of Operating Systems (Windows, Linux, Mac etc.)

2.5 Design and Implementation Constraints

- Considering the cost involved in 3rd party validation of Credit card numbers the validation is done by a simple validation mechanism integrated into the system.
- In the view of maintenance and investment factors involved in having a separate server for live auction, the same web browser will be used for live auction.

2.6 User Documentation

A customer support system will always be available for support during auction process.

2.7 Assumptions and Dependencies

The system can be used only if the user logs in with correct credentials and is already registered on the web application.

The further transactions can be carried on only with the permission of the administrator. The web application will not be responsible for shipping and transport of articles (books). It is left to the buyer and seller of the articles.

The web application does not deal with advertisement charges and its transaction and any subsequent legal issues out of it.

3 SYSTEM FEATURES

System features for the online auction management system have been developed to make sure that the functionalities & functional aspects of the system are met.

3.1 Bidder/Seller Registration

3.1.1 Description and Priority

This allows a user to be a member of e-bid and enables him to sell or bid on items and participate in the community forum. This is an essential part of the system of high priority as it adds new customers to the buyers /seller's community.

3.1.2 Response Sequences (Functional requirement)

- The user selects the ‘sign up ‘link to get the registration form and after following the instruction fills in the required information to create his/her profile.
- Client-side validation will be done for checking provided information.
- The user will be asked for a login and password.
- After client-side validation for uniqueness an activation link will be sent by mail.
- On activation the user is now registered to an online auction-based bookstore.

3.2 Database Search

3.2.1 Description and priority

Any user can search for articles based on category, cost and other parameters. This will be the top priority module as it is very essential for a prospective buyer to find the article, he/she is

looking for. The search will be of two type basic and advanced searches.

3.2.2 Stimulus/Response Sequences (Functional requirement)

- Any user will be selecting either basic or advance search.
- If the user selects basic search, the search will be performed on the product and category. This is for users who are not very particular about the product features.
- If the user selects advanced search, he/she will be presented with a form to fill in the advanced features of the product like vendor, model, color etc. The search will be searched on these preferences and search results will be displayed.

3.3 Sell Product

3.3.1 Description and Priority

Here the registered users can post their articles for bidding. It's an essential part of the project. Software Requirements Specification for online auction-based bookstore.

3.3.2 Stimulus/Response Sequences (Functional requirement)

- The seller is authenticated.
- Article description form is provided to the seller. A snapshot of the article may also be uploaded. The category of the article needs to be selected. The description may include minimum bid value, time for bid and other article characteristics.
- On client-side validation the database of items for bidding will be updated.
- RSS feed for the category will also be updated.

3.4 Bidding and Buying

3.4.1 Description and Priority

The buyers will be able to bid on any item available for bidding. This is the core part of this auction web application.

3.4.2 Stimulus/Response Sequences (Functional requirement)

- The buyer searches the item to bid from the search option.
- The buyer is authenticated for his first bid on an item and the item is added to his bidding list.
- For subsequent bids, he selects the item he wants to bid from the list of items he is currently bidding and updates his bid.

3.5 Live Auction

3.5.1 Description and priority

This module will allow seller to set a start time of the auction and will be a simulation of real auction scenario. The seller will be interacting with the bidders. This is an optional component and left to the sellers' discretion.

3.5.2 Stimulus/Response Sequences (Functional requirement)

- The seller selects the live auction option while selling an item
- The live auction time will be set.
- Buyers enter the live auction using ‘live auction link’.
- Buyers select the live auction he is interested in from the currently running live auctions.
- The buyer is authenticated before entering the live auction.
- Now they bid their values along with interaction with the seller

3.6 Authentication of buyers/sellers

3.6.1 Description and priority

The seller is authenticated before each article is posted for selling. The buyer is authenticated before he bids. The authentication is done through credit card or e-mail/mobile confirmation.

3.6.2 Stimulus/Response Sequences (Functional requirement)

- The buyer or seller has to select the mode of authentication.

- If the mode is credit card number its verified using suitable technique.
- If the mode is email/mobile then an email is sent with a secret code which the user has to message back to the server in a particular format
- On receipt of the message the user is authenticated.

3.7 24/7 Customer Support

3.7.1 Description and priority

This will allow the users to send in their queries to the customer support system. Designated staff for the Customer support system will answer the queries. This is optional part and is dependent on the availability of extra staff.

3.7.2 Stimulus/Response Sequences (Functional requirement)

- The user opens the customer support link to send in their query.
- The operator on the customer support system replies to the queries.

4 EXTERNAL INTERFACE REQUIREMENTS

The interface ought to be web based.

System features should be improved for better execution.

4.1 User Interface

It is designed to be functional and minimal in its styling. All products will be displayed in an image and description format. HTML and CSS will be used to setup the page layout & add minimal styling to make the interface user friendly.

- Simple Graphical user interface (GUI) for easy navigating through the program
- Easy to update profile and items
- Dynamically configurable interface
- Search functions

- Appealing to the eye through coloration and pictorial presentation

4.2 Hardware Interface

The System must run over the internet, all the hardware shall require to connect internet will be hardware interface for the system. As for e.g., Modem, WAN – LAN, Ethernet Cross-Cable.

- Processor speed of 0.5 GHz or more for mobile gadgets
- Processor speed of 1.5 GHz or more for desktop and computer gadgets
- Ram of 500mb and above for all devices
- Free storage memory capacity of more than 100mb

4.3 Software Interface

The system is on server so it requires the any scripting language like PHP, VBScript etc. The system requires Data Base also for the store the any transaction of the system like MYSQL etc. system also require DNS (domain name space) for the naming on the internet. At the last user need web browser for interact with the system.

4.4 Communication Interface

Internet connectivity

- Client on Internet will be using HTTP/HTTPS protocol.
- Client on Intranet will be using TCP/IP protocol.
- A Web Browser such as IE 6.0 or equivalent.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The device to be used to open the online auction website needs to have internet access in order to connect to the database. This software will not assume that a code scanner hardware is available on the machine, and so the USER_ID input will be done via keyboard and mouse. user need internet access on their devices as well, since all the data will be stored on the server database which the software will need to connect to.

5.2 Safety Requirements

The detail of the buyers and seller is kept secret. The details of the competitor bidders are also kept secret to provide all safety to the users of the software.

5.3 Security Requirements

The online auction system uses secure authentication for the online auction system administrator. Login Id & Password is associated with the System Administrator to provide security over the system.

5.4 Software Quality Attributes

· **Reliability**: The system is completely reliable as it has been checked for several trial runs before it is made available to the users.

· **Adequacy**: The input required of the user is limited to only what is necessary.

· **Maintainability:** The system is completely suitable for debugging (localization and correction of errors) and for modification and extension of functionality.

· **Readability:** The system is made in such a way that it can be easily used by everyone.

· **Extensibility**: Required modifications at the appropriate locations can be made without undesirable side effects.

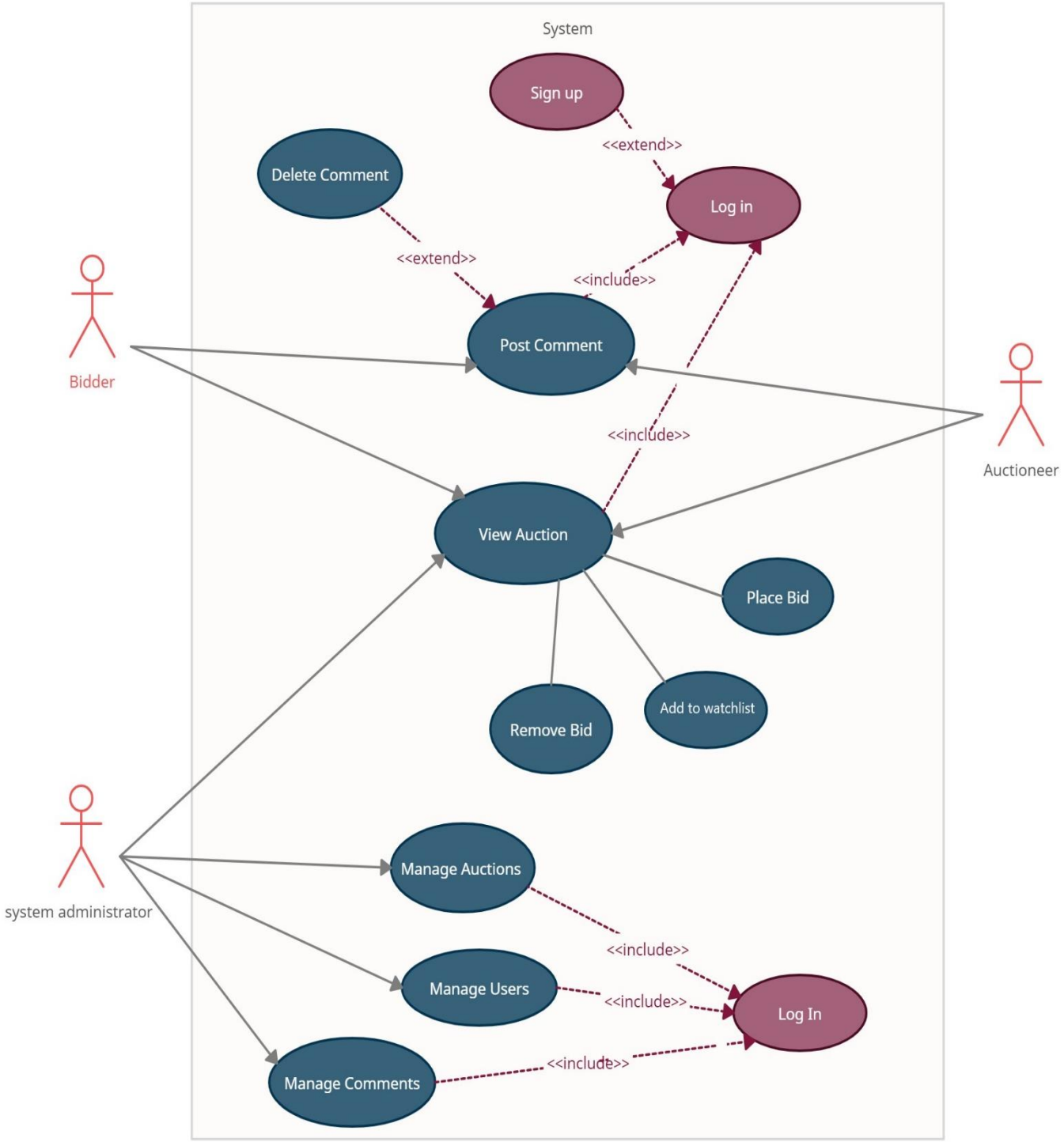
- **Efficiency**: ability of a software system to fulfill its purpose with the best possible utilization of all necessary resources (time, storage, transmission channels, and peripherals).
- **Portability:** the ease with which a software system can be adapted

Appendix A: Glossary

TERM	FULL FORM
HTML	Hypertext markup language
QA	Quality Assurance
SCMP	Software Configuration Management Plan
SDD	Software Design Document
SRS	Software Requirements Specification
TBD	To be decided
MySQL	Open-Source Database Server

Appendix B: Analysis Models

Appendix B.1 USE CASE DIAGRAM: -



DESCRIPTION- This diagram here depicts it as a graphical depiction of a user's possible interactions with the system. This use case diagram shows various use cases and different types of users the system has and will often be accomplished by other types of diagrams as well. In this use case the bidder and the auctioneer, both are interacting with our website to view the auction and to post comments on the item. Before that they have to login into the system and if they don't have an account, the other use case will come into action and that is sign up which has an extended relationship with login. Further, View auction has a generalization relationship with other three use cases Add to watchlist, Place Bid and Remove Bid. A user can delete comment which is other use case which is related to post comment with extended relationship. Now for System Administrator, he also required to log in to interact with system for managing the auctions, the users and the comments.

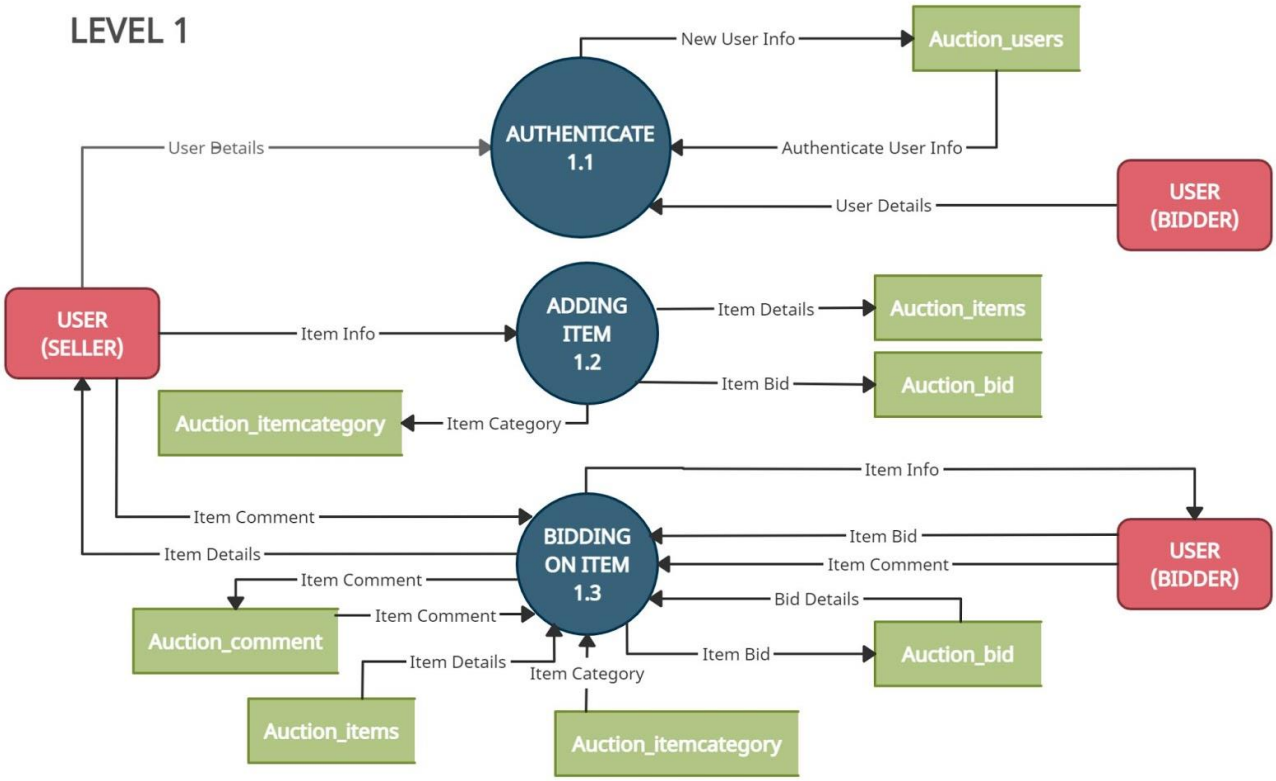
Appendix B.2 USE CASE TEMPLATE: -

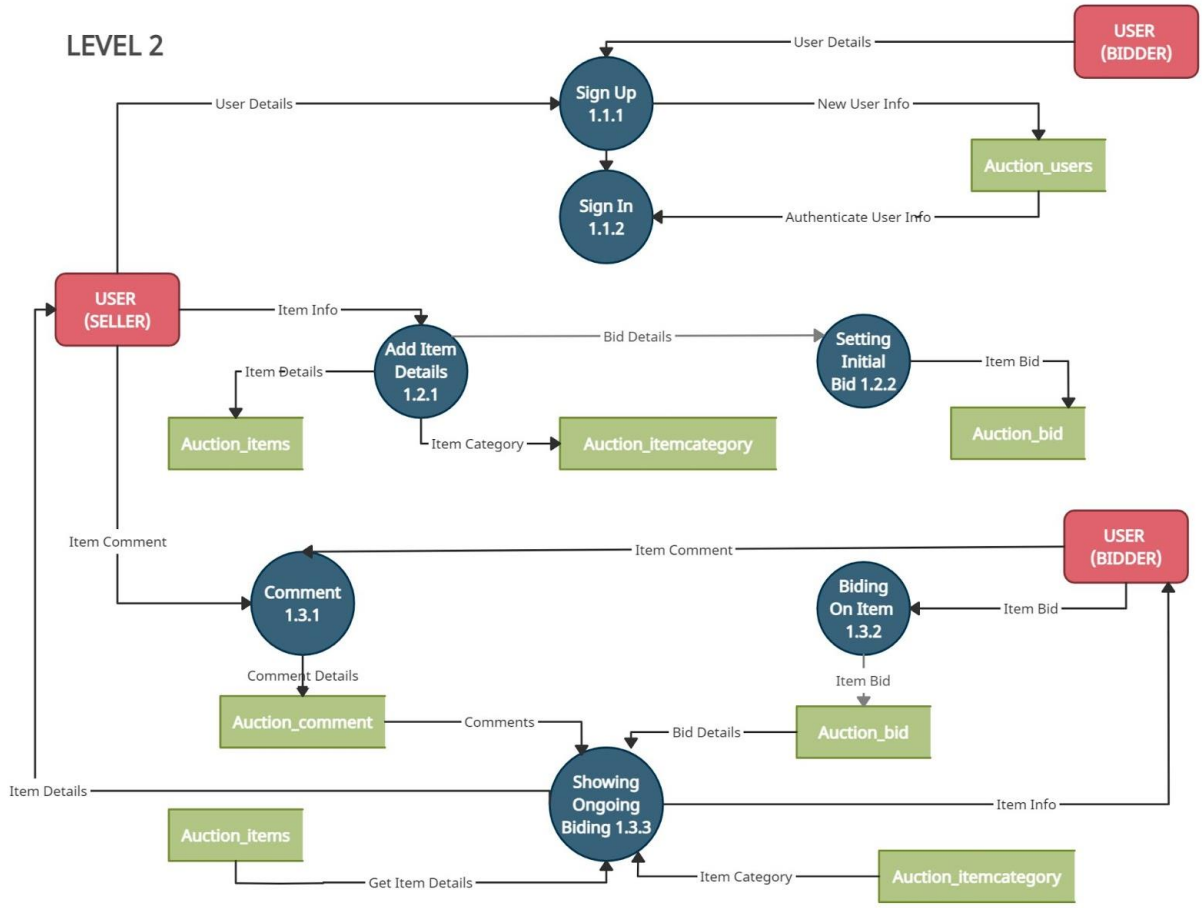
ID:	Ucs_oas 1
Use Case Title:	Adding the book
Description:	Seller add the book in active listing with details.
Primary Actor:	Seller
Stakeholders and Interests:	Seller : Want to sell his books at the best price possibly given by any bidder.
PreConditions:	Seller already has a login id. Seller should be logged in.
Success Guarantee(PostConditions):	Seller added items to active listings. Sellers can comment on it.
Main Success Scenario:	1) Seller arrives at the portal with books to sell 2) Me sign in to portal 3) Seller adds a book by choosing category & initial price of book . 4) Sellers comment on the comment section for each book. 5) Seller sets times for items when he gets the desired price.
Extensions	If the system fails, the seller restarts the system & tries to log in again to go to his previous state. If anomalies in preventing recovery : • Enters a clean state. • Seller checks if the items are added correctly else. • Seller added the same product again which was not added due to system failure.

ID:	Ucs_oas 2
Use Case Title:	Bidding & commenting on book
Primary Actor:	Bidder comes to specific book start bidding & commenting on it.
Primary Actor:	Bidder
Stakeholders and Interests:	Bidder : Want to buy book at low price possible.
PreConditions:	Bidder already has a login id. Bidder should be logged in.
Success Guarantee(PostConditions):	Bidder if winner, able to get book in his auction history. Bidder successfully commented & placed bid on it.
Main Success Scenario:	1) Bidder go to specific book.. 2) Bidder comment on it if needed. 3) Bidder placed bid on book . 4) If times up, winner is declared.
Extensions	In case the bidder is unable to comment, place a bid or remove comment due to some technical error. • Restarts the page • If not resolved, bider log out and log in again and check the status.

Appendix B.3 DATA FLOW DIAGRAM: -

LEVEL 0:





DESCRIPTION-Data flow diagrams are used to graphically represent the flow of data.

Level 0 shows two external entities that are the user – seller and the user – bidder, and one process called auctions.

In Level 1, we have divided the process auctions into three smaller processes namely, Authenticate, Adding item and Bidding on item.

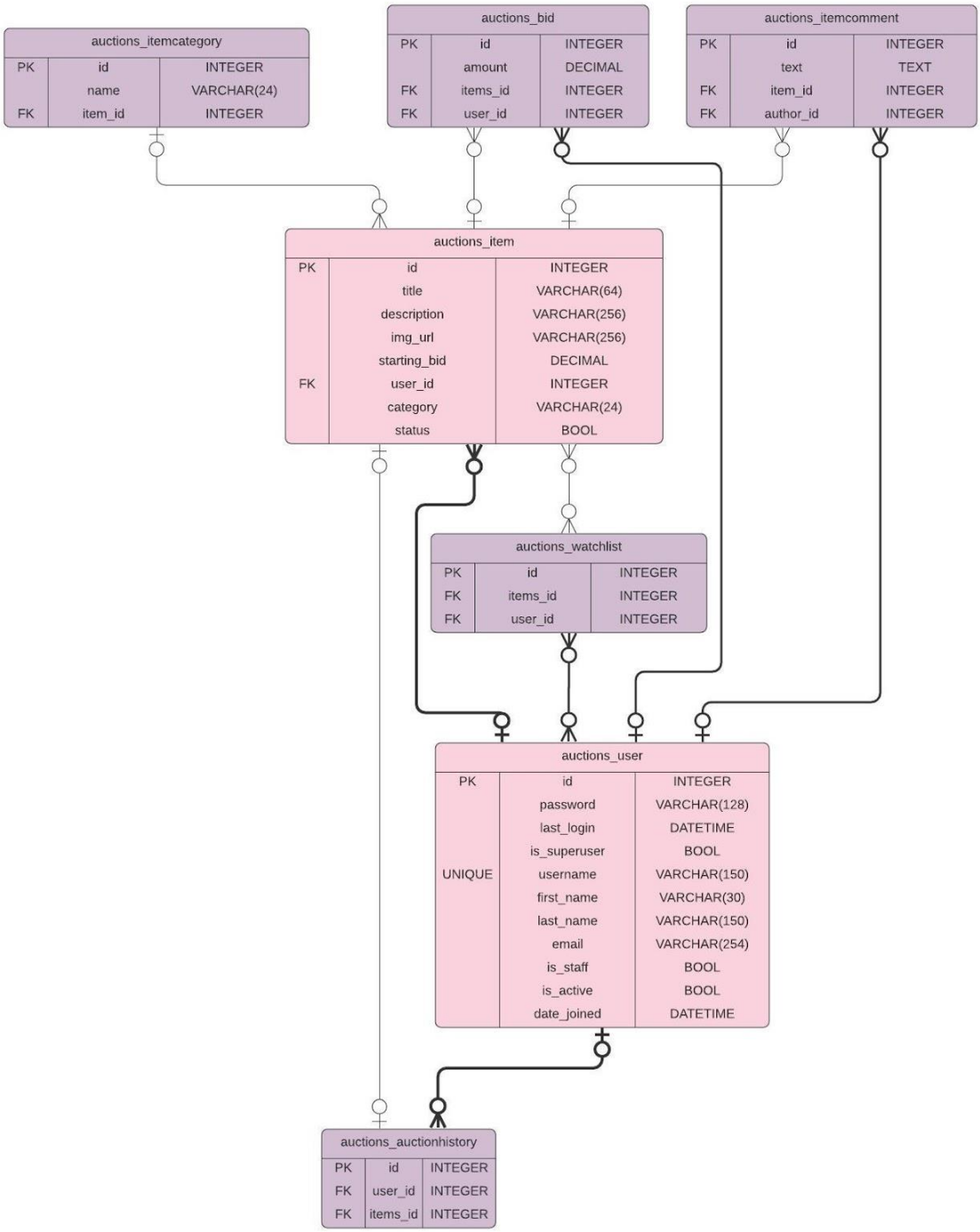
In Level 2, we have further divided the Authenticate process into Sign Up and Sign In, Adding Item into two process adding item details and setting initial bid, and bidding on item into three processes namely comment, bidding on item and showing ongoing bidding.

Appendix B.4 Data Dictionary

	Data flow	Subdivision	Data Type	Description
1.	User details	Username + Password		Get user details for Sign in
2.	Item info	Item details + Bid details + item category		Information of book containing details, bid details and book category.
3.	New User info	Username + Password		Get new user info as username and password.
4.	Authenticate User Info	Username + Password		Authenticate the user from database.
5.	Bid details	Item bid		Setting bid for the book by bidder
6.	Item details	Title + Description + Image URL + Starting Bid		Details of book, seller wants to add.
7.	Comments	Item comment		Comments on book, with current bidding.
8.	Item comment		Text	Comments on book, which is for bidding.
9.	Item category		Varchar (24)	Category of book, which is for bidding.
10.	Item bid		Decimal	Bid value of ongoing bidding book.
11.	Username		Varchar (150)	Username of user for sign in and sign up.
12.	Password		Varchar (128)	Password of user for sign in and sign up.
13.	Title		Varchar (64)	Title of the book added by seller.
14.	Description		Varchar (256)	Description of the book added by seller.
15.	Image URL		Varchar (256)	Image URL of the book added by seller.
16.	Starting Bid		Decimal	Starting bid of the book added by seller.

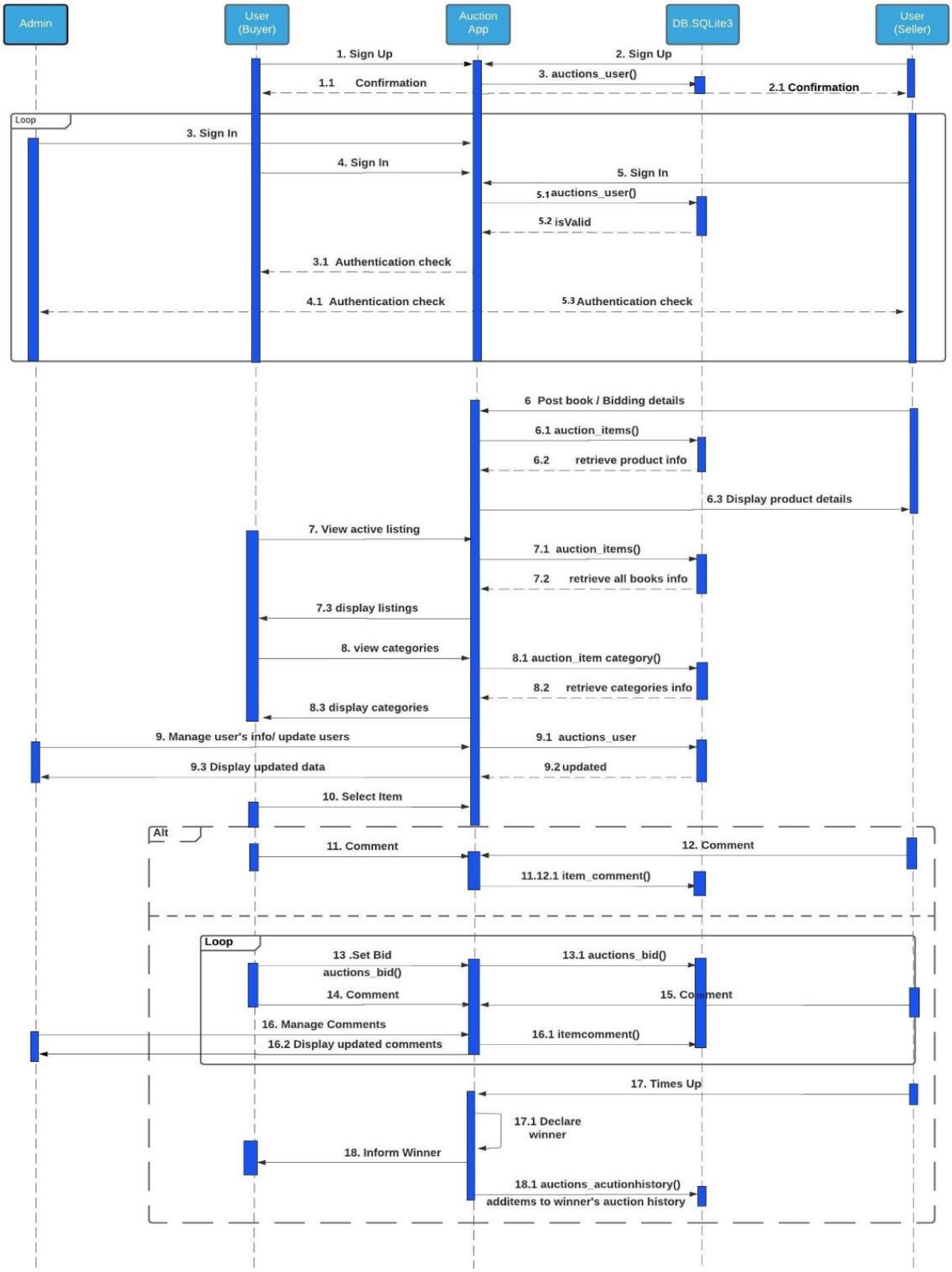
DESCRIPTION - Data dictionary shows the different data flows and different data variables used along with their subdivision or data type and description.

Appendix B.5 Database Schema



DESCRIPTION- We have seven database tables namely, auctions_ bid, auctions_ itemcomment, auctions_ item, auctions_ itemcategory, auctions_ user, auctions_ watchlist, and auctions_history. Database schema diagram show the different columns in these tables and information about the keys (primary and foreign) and constrains.

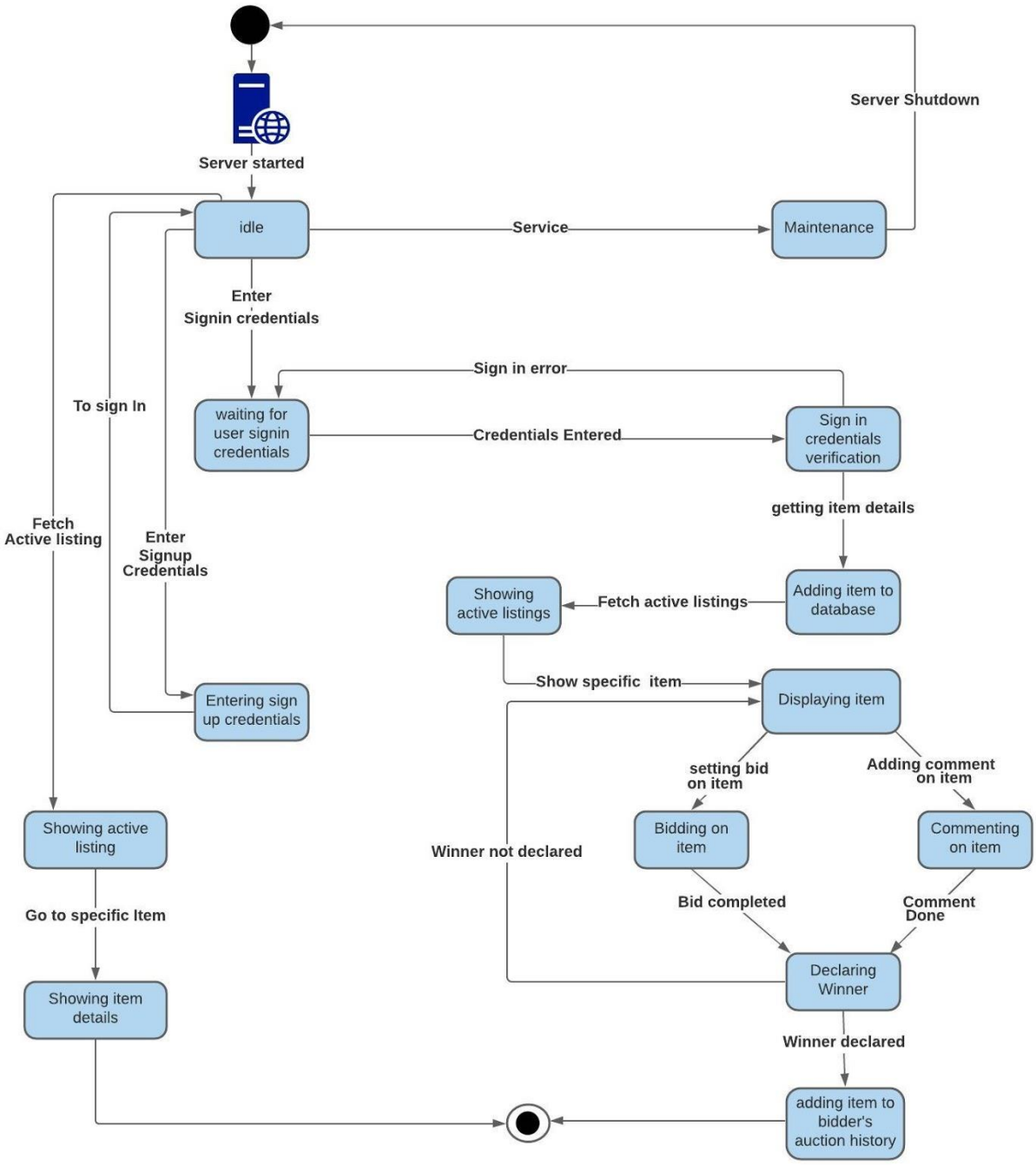
Appendix B.6 SEQUENCE DIAGRAM: -



DESCRIPTION- Sequence diagram shows the sequence of messages flowing from one object to another. We have five objects namely, the admin, the user - bidder(buyer), the auction system, the database server (SQLite3), and the user – seller. The diagram shows

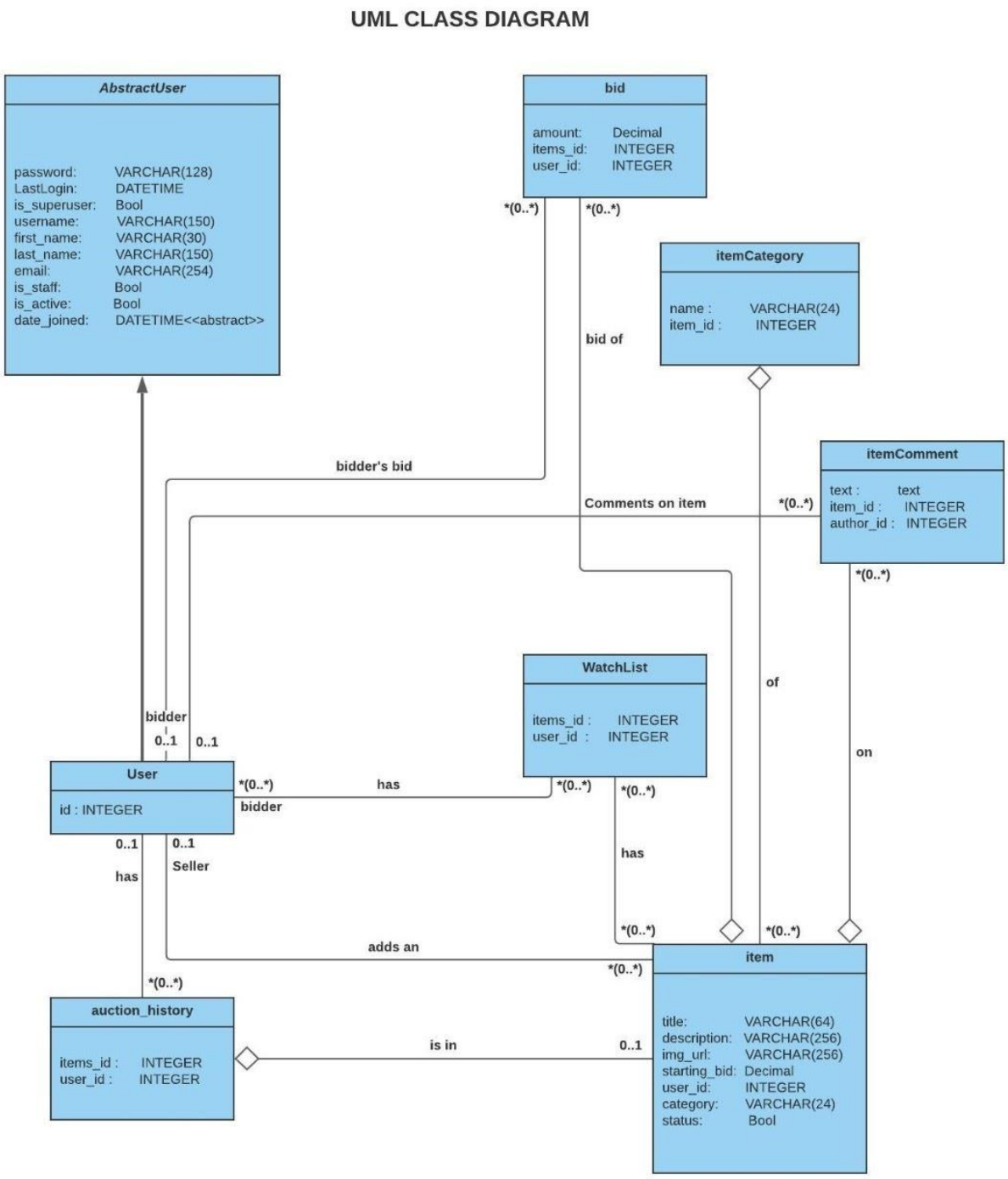
the order in which the operations take place with the use of proper symbols and numbering.

Appendix B.7 STATE DIAGRAM:



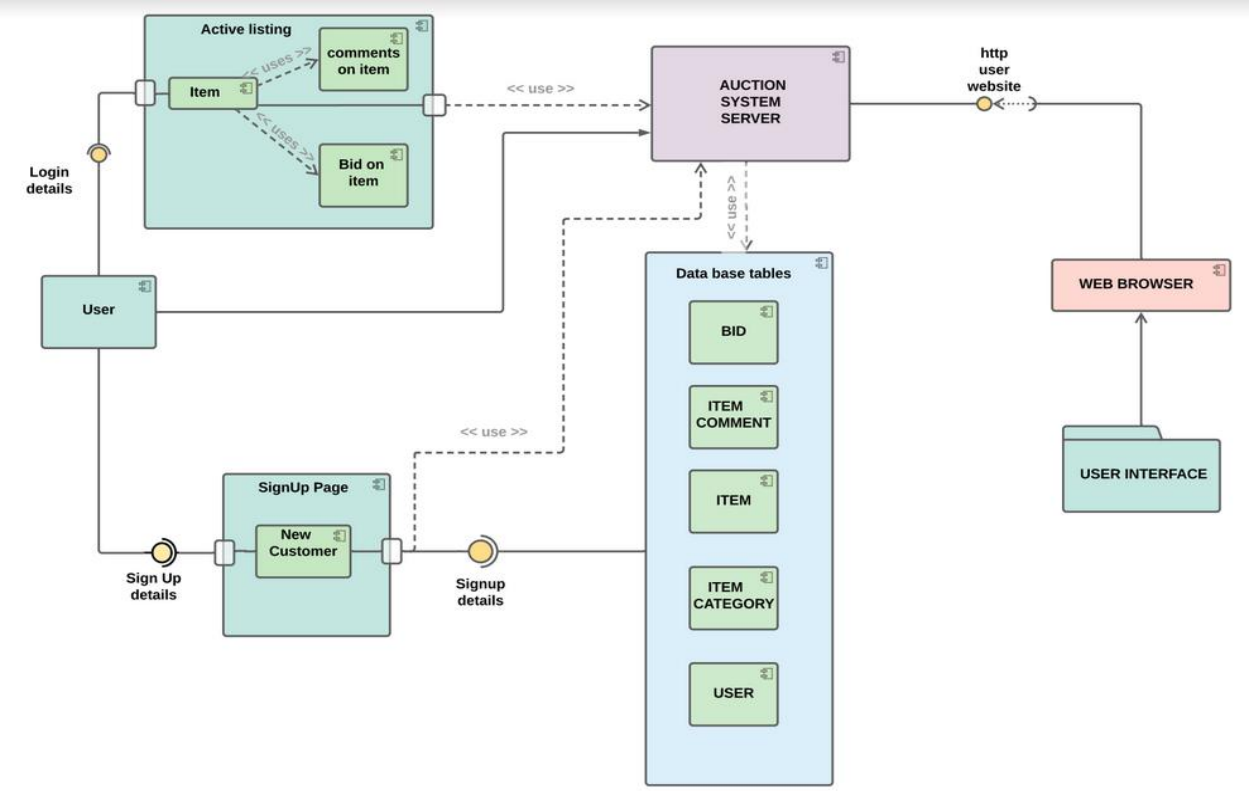
DESCRIPTION- State diagram is used to state the events responsible for change in state. It shows that when user wants to enter sign in details, the system will be waiting for the credentials. When the details will be entered it will go for the verification of those details. It only shows the behavior of the system on some events like entering the sign in details without showing the process that causes these events.

Appendix B.8 CLASS DIAGRAM



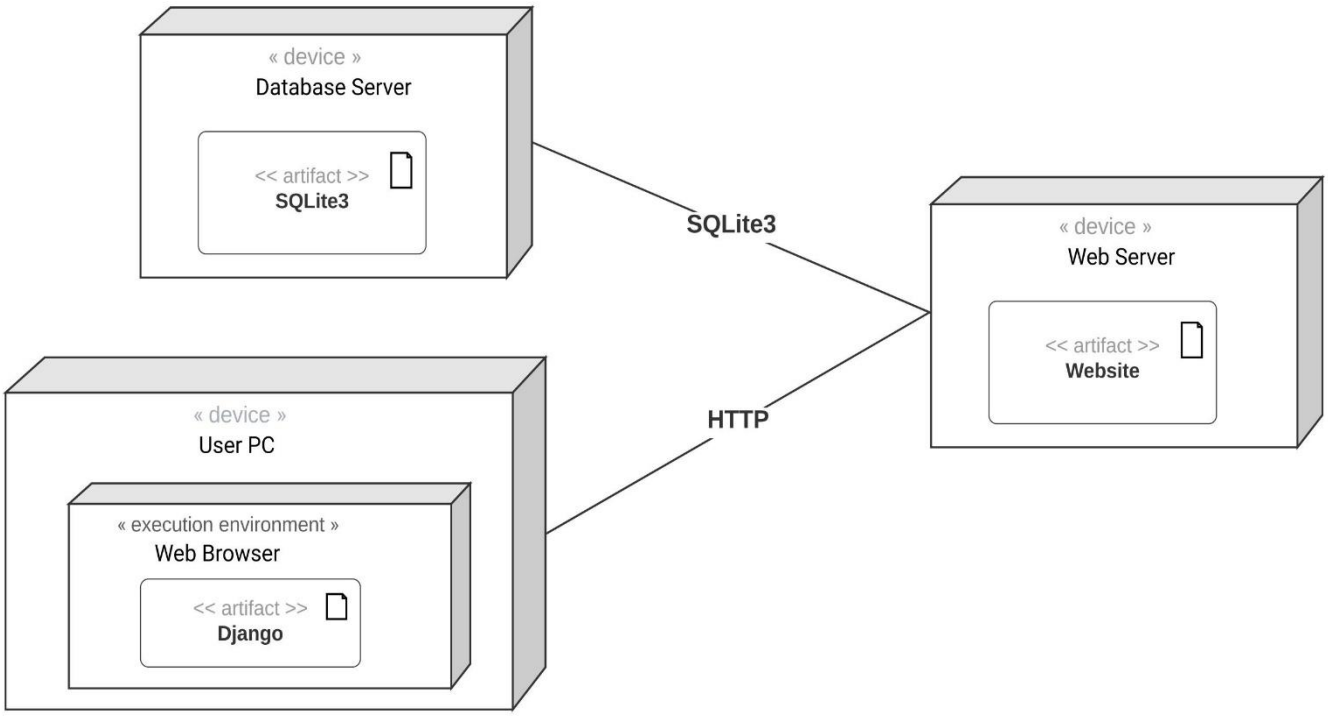
DESCRIPTION- Class diagram is a static diagram. It represents the static view of an application The top row contains the name of the class, the middle row contains the attributes (variable name and its data type) of the class, and the bottom section is empty because we have not used any methods (used multiple python codes instead of methods) so, it only shows the class name and attributes.

Appendix B.9 COMPONENT DIAGRAM



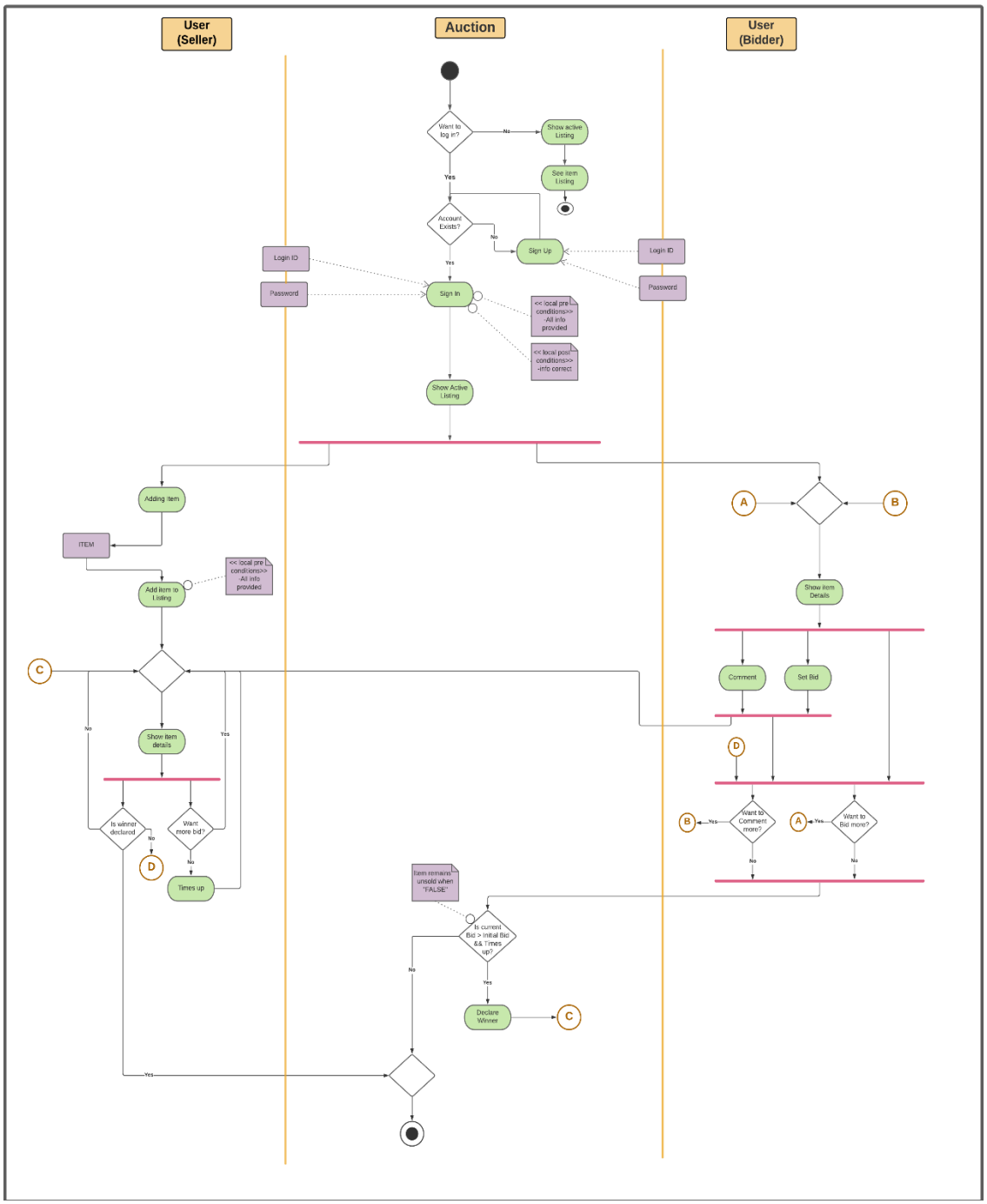
DESCRIPTION- Component diagram shows the physical components of the system. The user must provide login in detail to see the active listings or provide sign up details which is then stored in our database table. The user can then see the item and place a bid or comment on the item which is managed by the Auction System Server. The sign-up page and the active listing page use the auction system server which itself is uses the Database tables.

Appendix B.10 DEPLOYMENT DIAGRAM



DESCRIPTION- Deployment diagrams are used for describing the hardware components, where software components are deployed. In our project we have three external nodes namely, the database server, the web server, and the user PC. The database uses SQLite3 as artifact, the webserver has the website artifact written 'mainly' using HTTP and CSS, and the user PC contains the internal software node that is the web browser which serves as the execution environment for the project.

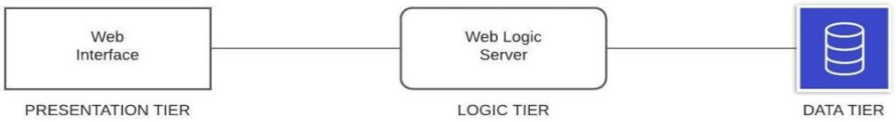
Appendix B.11 ACTIVITY DIAGRAM



DESCRIPTION- Activity diagram shows the flow of control. The user can see the active listing without even logging in. If the user wants to login, he can log in by providing the log in ID and password (if account already exists). If account does not exist, he can provide the required details and then log in again. After logging in the diagram splits into two, based on the type of user (seller or bidder). The seller can add an item to the item database with the local precondition that all information must be provided. Then the seller can see the current item details where he can either choose to end the bid when he is satisfied with the current bid and the winner will be declared. In the bidder section, the bidder can either bid or comment after seeing the item details and he can do it until the seller hasn't ended the bid (by using times up condition). The bidder can bid and comment multiple times until item is sold. When the time is up and all conditions are satisfied the winner is declared.

Appendix B.12 ARCHITECTURAL STYLE

The implemented online web auction system follows a three-tier architecture. It has a client component (front end application), middle tier component, and backend tier component. The three-tier architecture is as shown in the figure below.



Client Component (Presentation Tier)

The client component usually contains programs executed by users, including Web browsers and network-capable application programs. For the current system's client interface HTML pages act as the front end. For the user interface we have deployed servlets and JavaScript pages in SQLite3 Server.

Middle Tier component (Logic Tier)

The middle tier of the web auction system consists of SQLite3 Server. For the implementation of the project only a single SQLite3 server has been chosen. But for the purpose of scalability, additional SQLite3 servers can be added. By having the SQLite3 Server cluster option we can distribute client requests and back-end services among multiple SQLite3 Servers. The cluster uses a selectable load-balancing algorithm to choose a SQLite3 Server in the cluster that is capable of handling the request. Applications can be made easily scalable by having a middle-tier server. So, reliability, scalability, and high performance can be easily achieved.

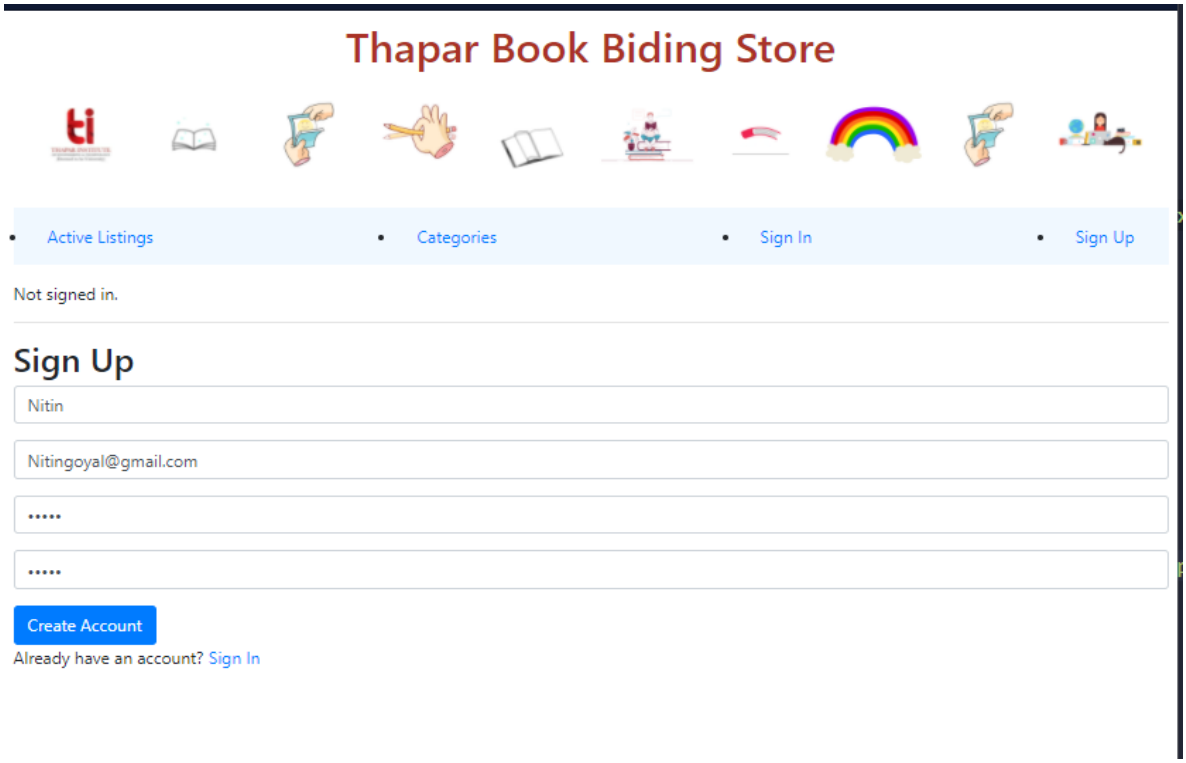
Backend Tier component (Data Tier)

Database is the backend tier component in the three-tier configuration where information processed by the application is processed and managed.We have chosen SQLite3 for the database.After the installation of the database components, an auction user is created and appropriate tables that store the user, bidding, and transaction data are created.

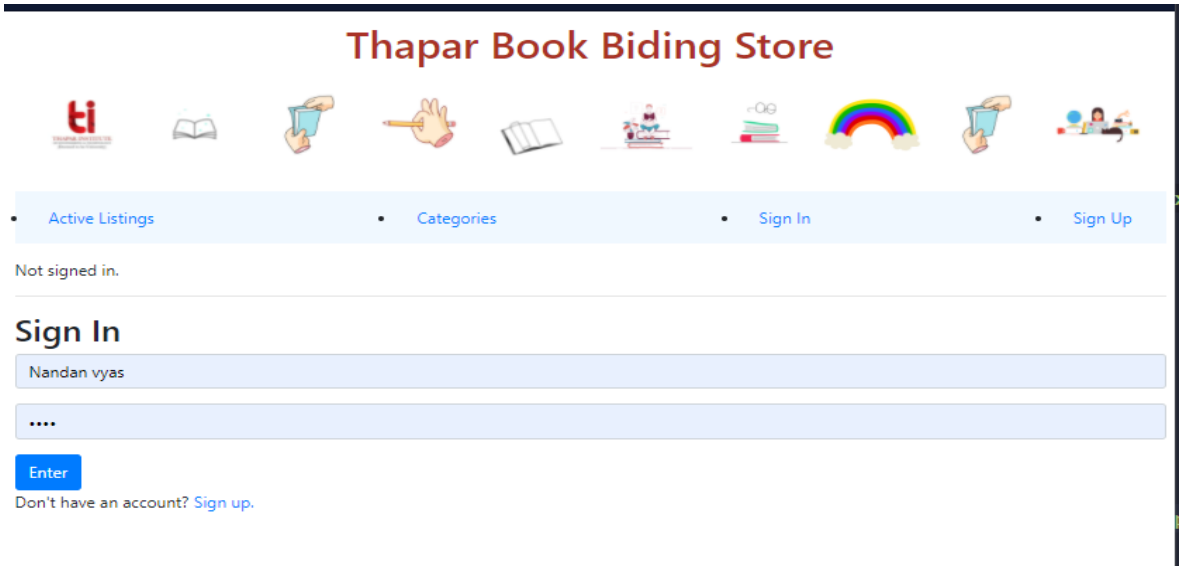
DESCRIPTION- We have implemented a three-tier architecture. The front-end component or presentation tier consists of HTML pages. The logic tier consists of a single SQLite3 server and the data tier consists of the database which contains multiple tables namely, bid table, item comment table, item table, item category table, and the USER information table.

Appendix B.13 USER INTERFACE DESIGN (UID) SNAPSHOTS

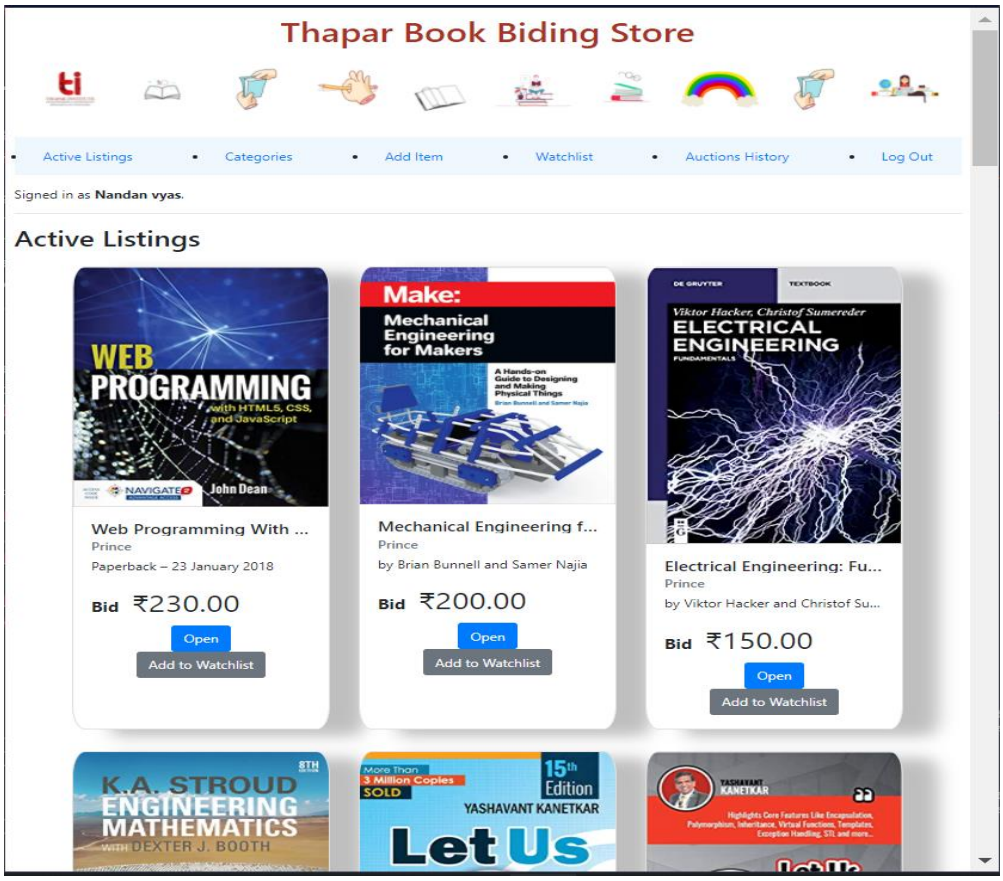
Appendix B.13.1 Sign Up



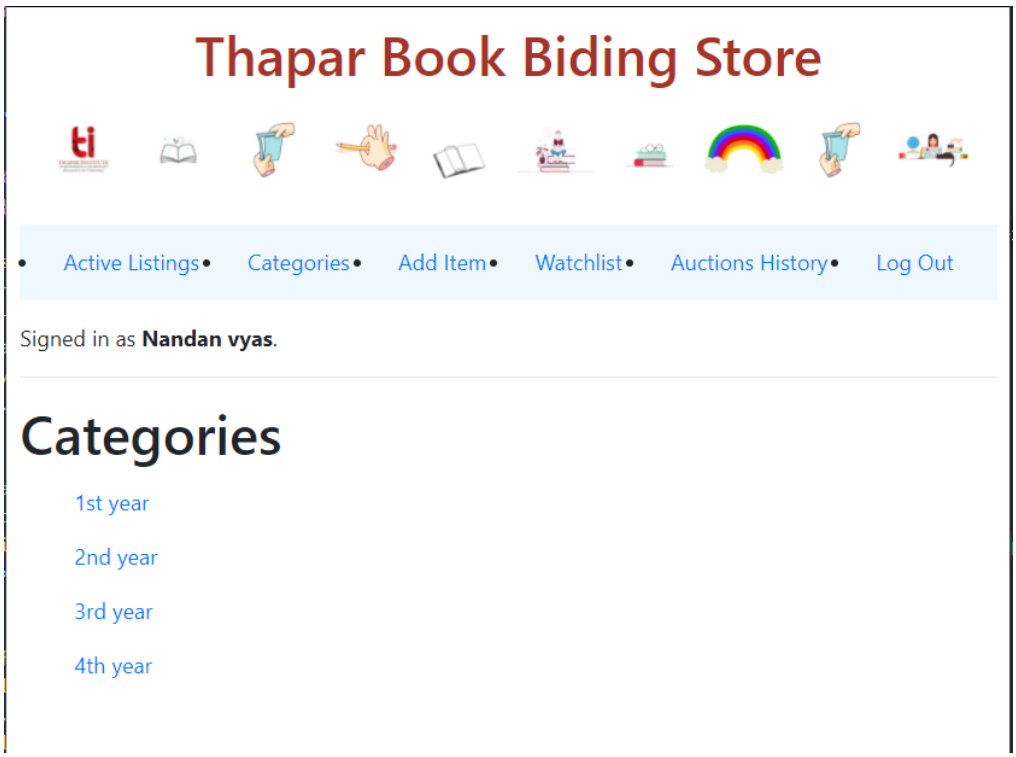
Appendix B.13.2 Sign In



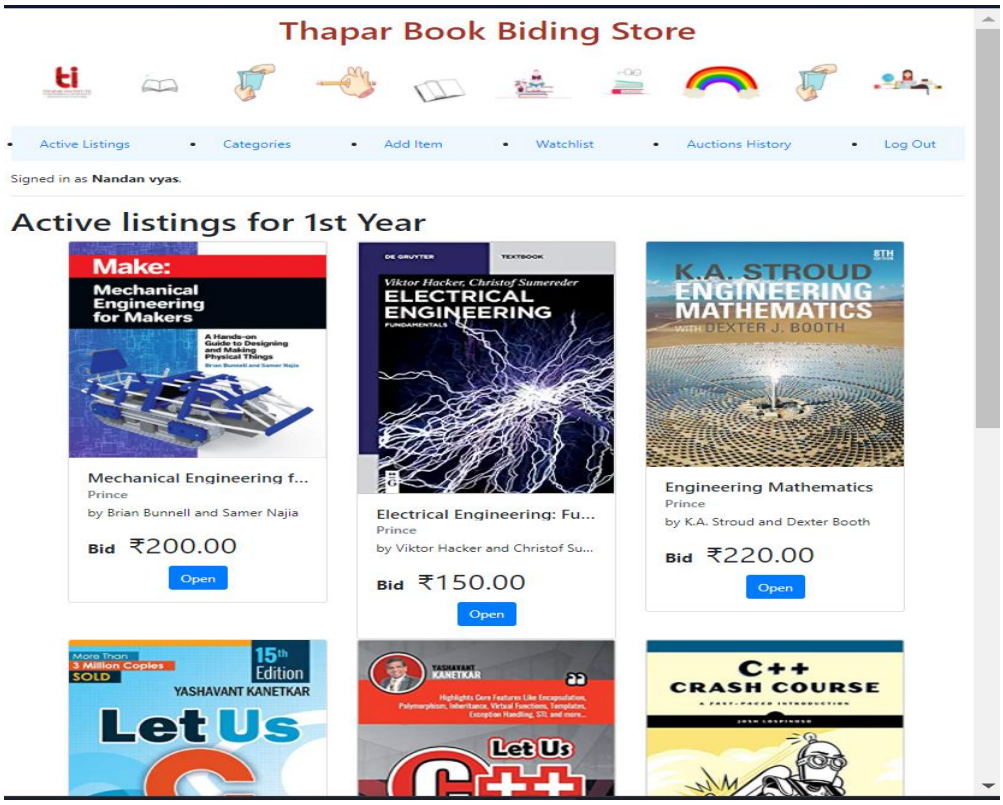
Appendix B.13.3 Active Listing



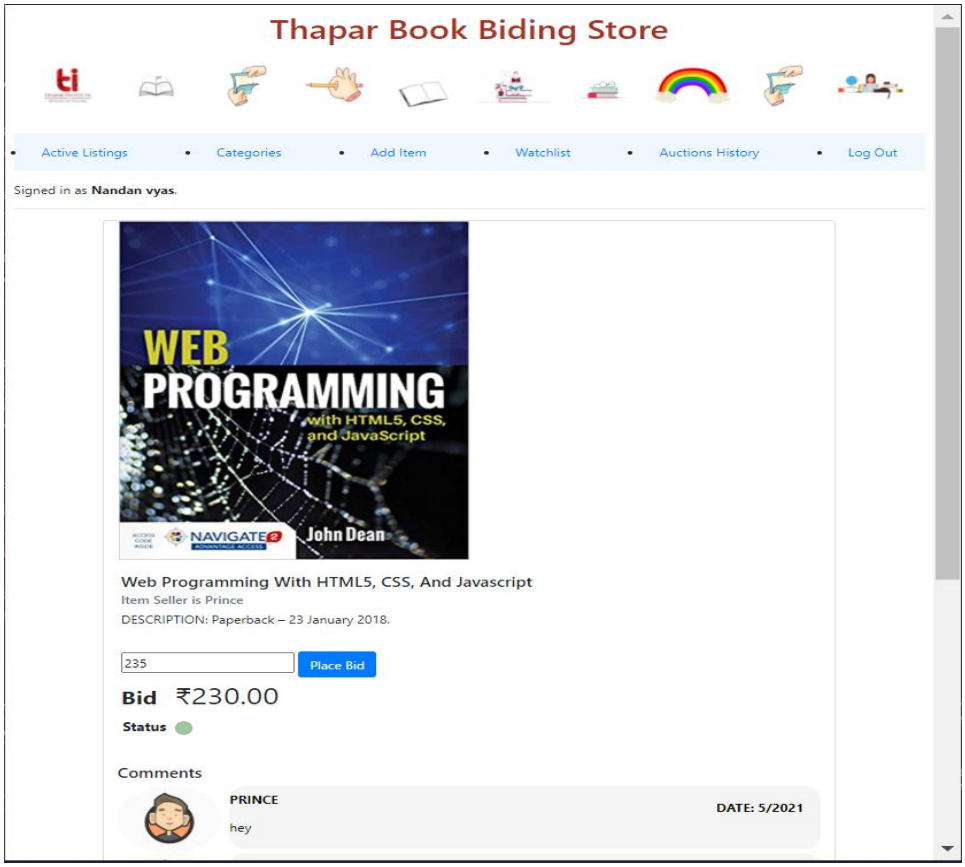
Appendix B.13.4 Categories



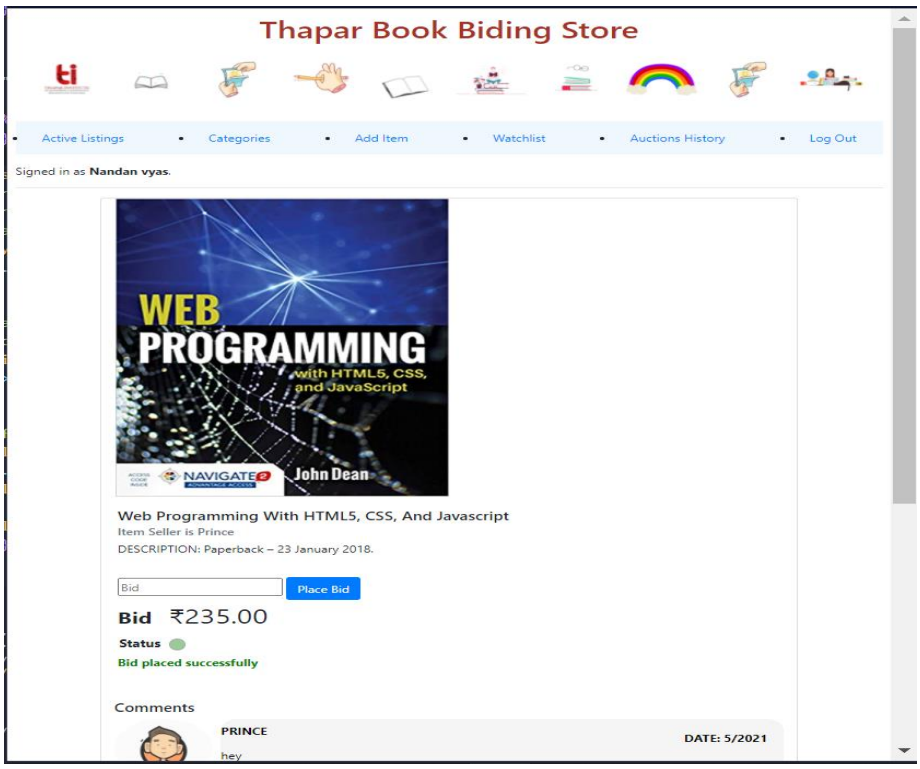
Appendix B.13.5 1st Year Active Listing



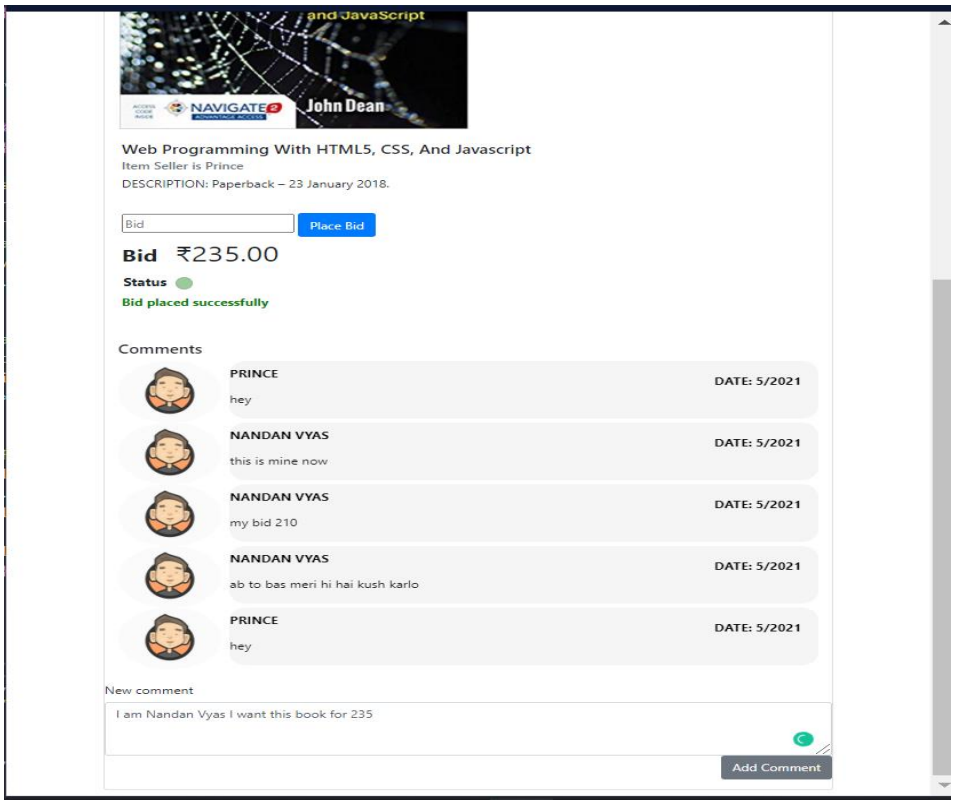
Appendix B.13.6 Specific item Page



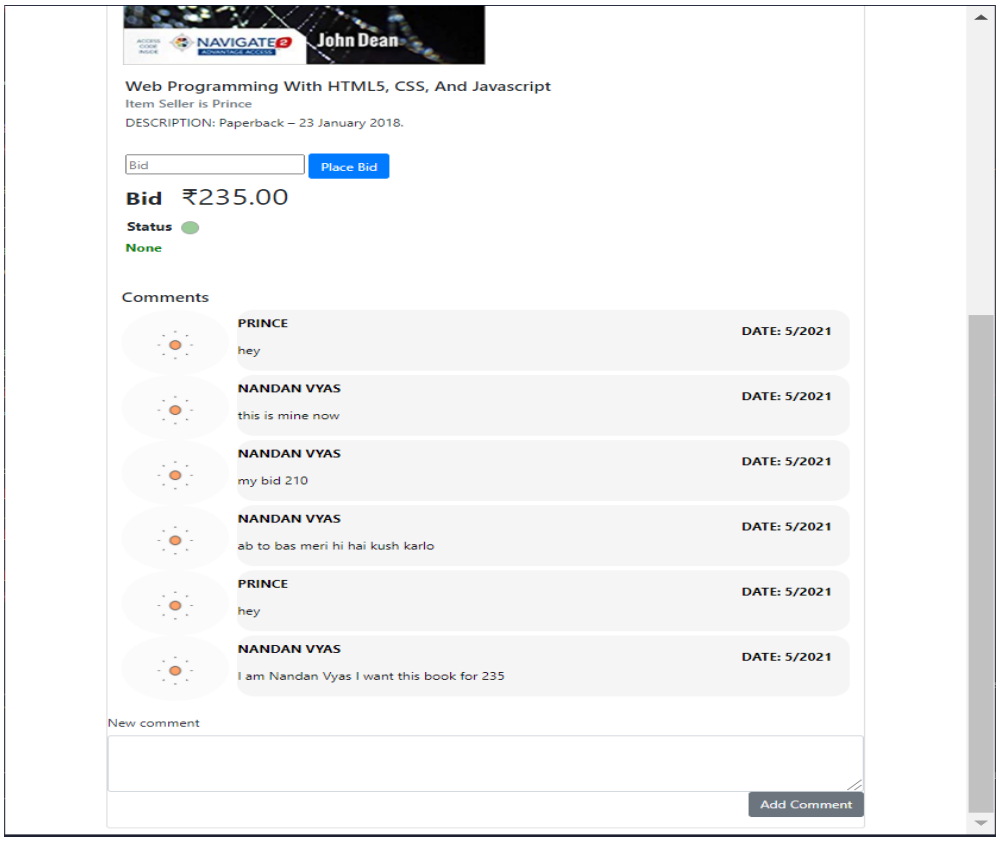
Appendix B.13.7 Place Bid on Item



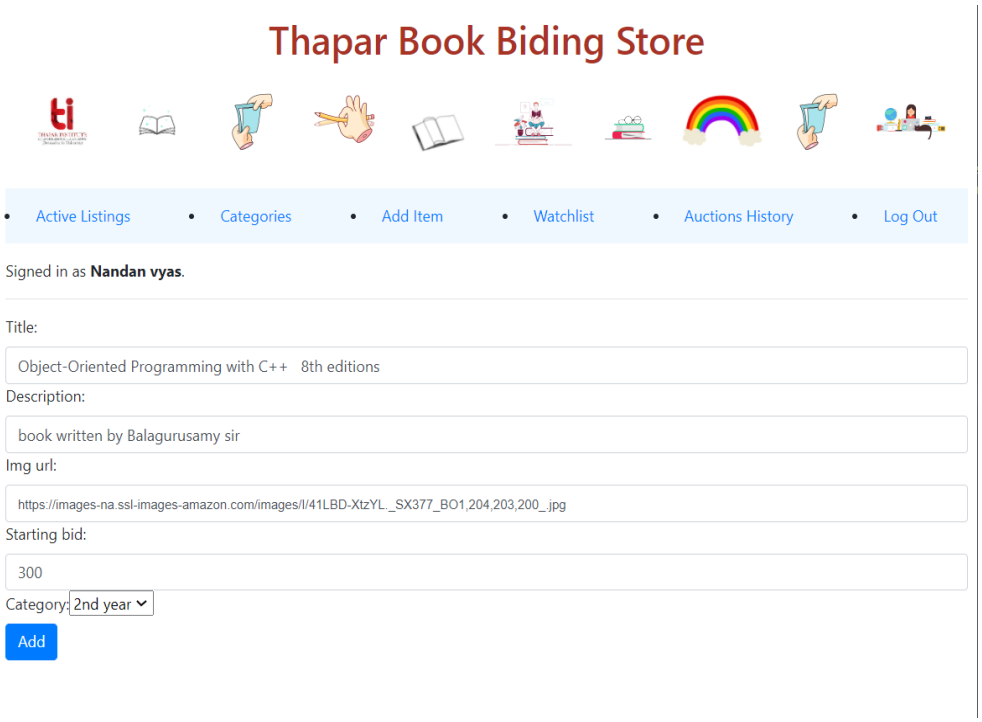
Appendix B.13.8 Adding Comments



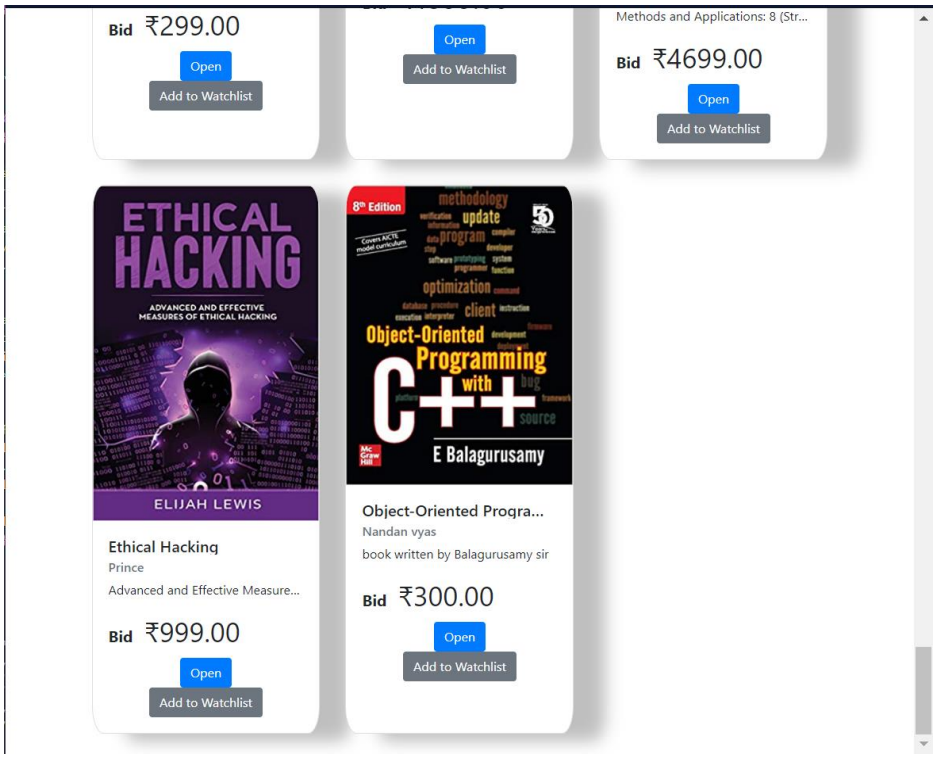
Appendix B.13.9 Comment added Successfully



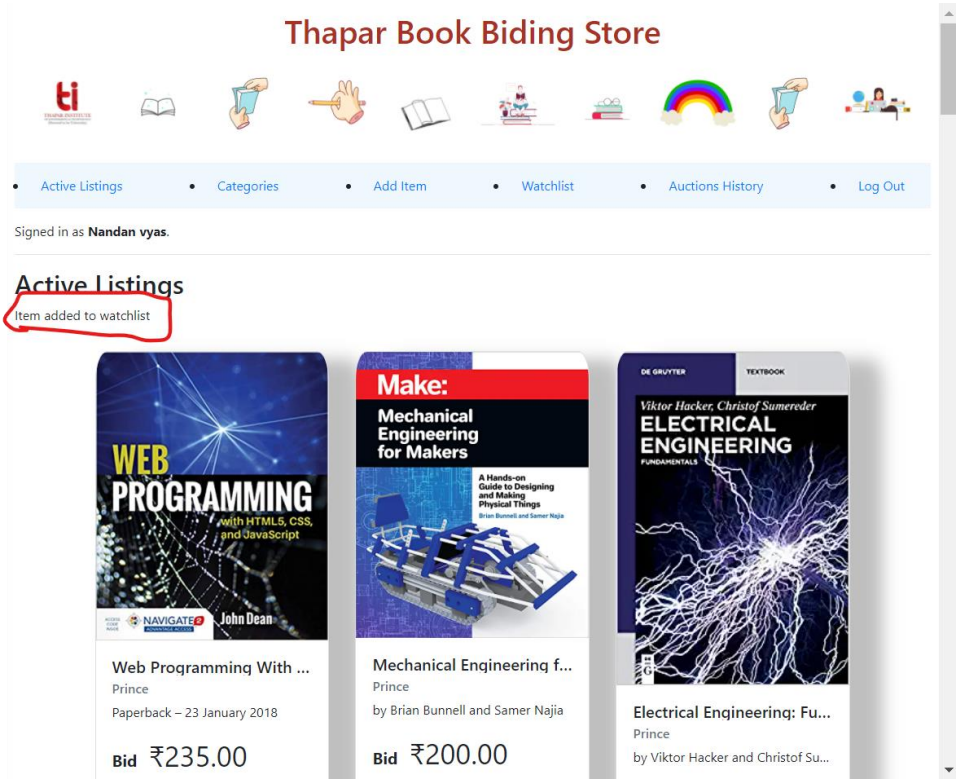
Appendix B.13.10 Add Item



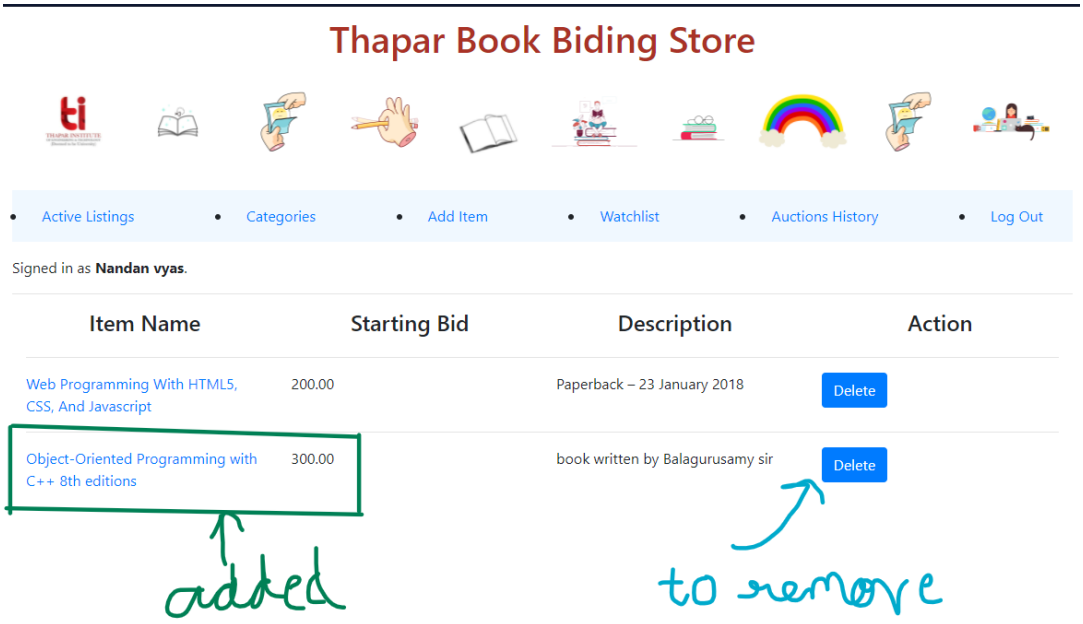
Appendix B.13.11 Item added Successfully



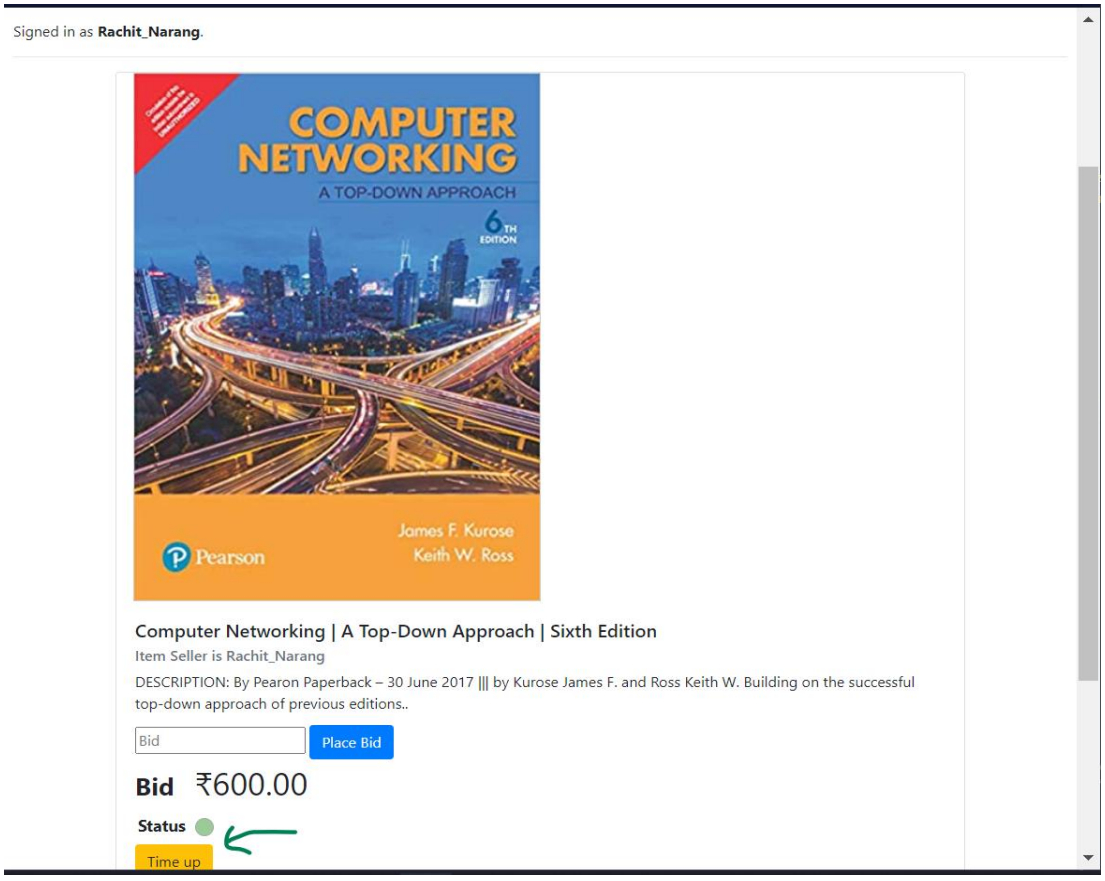
Appendix B.13.12 Adding Item to Watchlist



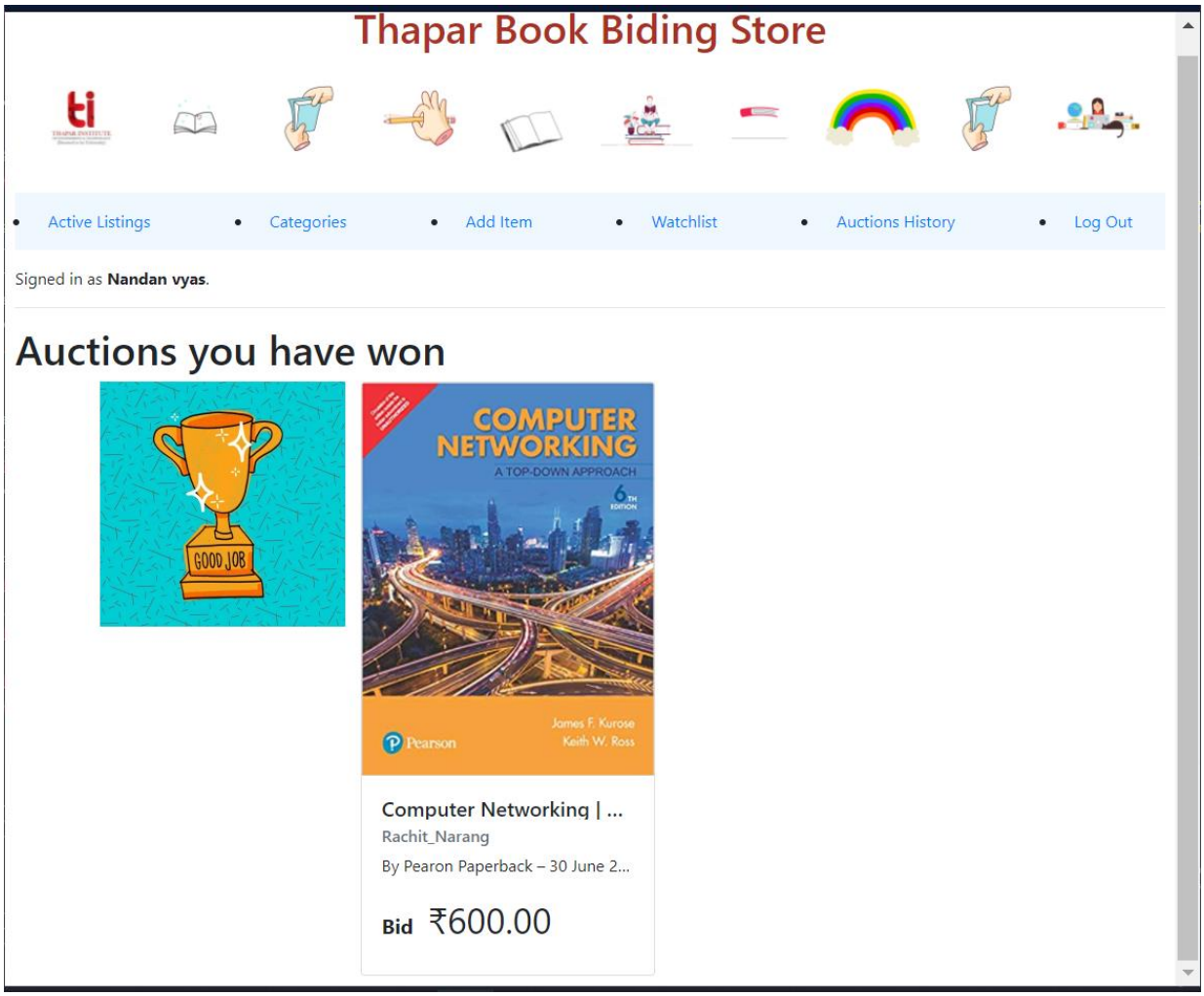
Appendix B.13.13 User's Watchlist



Appendix B.13.14 Time's Up by Seller



Appendix B.13.15 Show Auction Won History



7. Testing

S no.	Check Point	Status
1	Sign Up	Passed
2	Sign in	Passed
3	Active Listing	Passed
4	Categories	Passed
5	Year wise active listing	Passed
6	Specific Item Page	Passed
7	Place Bid on item	Passed
8	Writing Comment	Passed
9	Successful Comment	Passed
10	Add item	Passed
11	Item Added Successfully	Passed
12	Adding to Watchlist	Passed
13	User's Watchlist	Passed
14	Times up by seller	Passed
15	Adding to watchlist	Passed