E-Bidding : Software Requirements Specification (SRS)

1.Introduction

1.1 Purpose

To build a secure, feature-rich, and real-time online auction platform enabling users to buy and sell products through competitive bidding, fixed pricing, and seamless communication.

1.2 Scope

1. Real-time bidding interface with countdown
2. Auto-bidding (proxy bidding)
3. Easy registration with multi-factor authentication
4. Fixed-price instant purchase option
5. Product detail and media upload
6. Secure payments with escrow
7. Alerts and notifications
8. Mobile and desktop compatibility

1.3 Definitions

1. Auto-Bidding: System auto-increments bid on behalf of user up to a set limit
2. Escrow: Third-party holding payment until product delivery
3. Proxy Bidding: Same as Auto-Bidding

2.Overall Description

2.1 Product Perspective

A cloud-native web and mobile auction system for general users with roles:

1. Buyers: Browse, bid, pay
2. Sellers: Add items, receive bids, manage listings
3. Admins: Monitor system activity, flag issues

2.2 User Classes

1. Guests: View auctions
2. Registered Users: Bid, add products
3. Verified Sellers: Host auctions
4. Admins: Manage platform

2.3 Assumptions

1. Users have basic internet access
2. Reliable API support for OTP, payments, notifications

3.Functional Requirements

3.1 Registration & Authentication

1. Quick signup with email/phone
2. Multi-factor authentication (OTP/email)

3.2 Add Product

Sellers will be able to add new products for auction or fixed-price sale, providing the following details:

1. Item Header
2. Description
3. Category
4. Photos/Videos
5. Base Price (with flexible price range option)
6. Quantity
7. Buy Now (fixed price)
8. Documents Upload (PDF/Images)

3.3 Auctions Page

1. Real-time display of all ongoing auctions
2. Sorting and filtering options

3.4 Real-Time Bidding

1. Live bid updates
2. Highest bid visibility
3. Countdown timer
4. Anti-sniping time extension

3.5 Auto-Bidding

1. Set maximum bid
2. System increases bid only when needed

3.6 Notifications

1. Outbid alert
2. Auction ending
3. New item
4. Winning notification

3.7 Messaging & Contact

Contact options after auction close:

1. WhatsApp
2. Internal message (sends email)

3.8 User Profile

1. Transaction history
2. Rewards tracking
3. Bought/Sold items
4. Payment and account info
5. Help & Support

3.9 Wishlists/Watchlists

1. Save items without bidding
2. Track auctions

3.10 Payment System

1. Multiple gateways (UPI, NetBanking, Cards, Wallets)
2. Escrow model
3. Digital receipts
4. Refund mechanism

3.11 Admin Features

1. Monitor suspicious activities
2. Flag/block fake sellers
3. Manage help desk

4.Non-Functional Requirements

4.1 Security

1. HTTPS encryption
2. Data encryption (DB)
3. OTP and 2FA

4.2 Performance

1. Real-time updates
2. Scalable infrastructure for traffic

4.3 Usability

1. Simple UI for all users
2. Mobile-first responsive design

4.4 Availability

1. 99.9% uptime
2. Backup and recovery features

5.Pages

Page Name	Description
Home	Hero redirect to Add Product / Auctions, trending items, categories, stats
Add Product	Complete seller options and listing creation form
Auctions	All auction items shown with filters and live updates
About	Info about the platform and vision
Notifications	All alerts and messages (bids, wins, payments)
User Profile	History, rewards, payments, help, watchlist

6.Future Enhancements

1. Mobile App (iOS/Android)
2. Voice-based bidding support
3. Video intro by sellers
4. AI recommendations engine

Technology Stack (Layered View)

Layer	Technology / Tools	Purpose

Frontend	React.js / Next.js	Interactive UI, SPA, real-time updates
Styling	Tailwind CSS / Bootstrap / SCSS	Responsive design and component styling
Backend	Node.js (Express) / Python (FastAPI, Django)	APIs, business logic, bid handling
Database	MongoDB (NoSQL) / PostgreSQL (SQL)	Store user, product, bid, and transaction data
Authentication	Firebase Auth / Auth0 / Custom JWT + OTP via Twilio	Secure login with 2FA/OTP
Real-Time Updates	Socket.IO / WebSockets	Live bidding updates, countdown timers
Payment Gateway	Razorpay / Stripe / Paytm / Cashfree	Secure payments + escrow handling
Notifications	Firebase Cloud Messaging (FCM) / Twilio / SendGrid	SMS/email alerts for bids, wins, etc.
Cloud Hosting	AWS / Vercel / Heroku / Render / DigitalOcean	App deployment and scalability
Storage	AWS S3 / Cloudinary / Firebase Storage	Media storage: images, videos, documents
Version Control	Git + GitHub / GitLab	Codebase management and collaboration
CI/CD	GitHub Actions / Jenkins / Vercel	Continuous integration and deployment

Design Issues and Considerations

Area	Design Issue	Recommendation
Real-Time Bidding	Race conditions, last-second bids (sniping)	Use WebSockets + Anti-sniping logic (extend timer if last-minute)

Auto-Bidding	Priority of bids with same max price	Use timestamps to resolve tie-breakers
Scalability	Spike in users during hot auctions	Use load balancers, horizontal scaling
Security	Payment fraud, data breaches	Use HTTPS, JWT, OAuth, encrypted DB fields, role-based access
Escrow System	When to release or refund payments	Define clear business logic and dispute handling
Data Consistency	Race condition between bid updates and user session	Use transactions or real-time locks in DB where needed
Mobile Compatibility	Touch input issues, screen size adaptation	Use responsive UI frameworks like Tailwind / Bootstrap
User Communication	Spam or abuse in messages after auction	Implement rate-limiting, report/block features
Notification Overload	Too many emails/SMS can annoy users	Allow user settings to control alert preferences
Search Performance	Slow filtering on large product lists	Use indexed search (e.g., Elasticsearch) or efficient DB queries
Auction Integrity	Fake bidders or bots	CAPTCHA, verification for bidders, bid logs

Function Point (FP) Explanation for E-Bidding Platform

What is Function Point (FP)?

Function Point (FP) is a software measurement metric used to quantify the amount of business functionality a system provides to its users. It helps measure the size, complexity, and effort required to develop or maintain a software application based on its functional requirements, independent of the technology used.

Instead of counting lines of code, FP counts distinct user-facing functionalities like:

- **Inputs (data entry screens):** Such as "Register," "Login," "Add Product," "Place Bid,"

"Upload Docs," and "Message."
- **Outputs (reports, notifications):** Including "Bid status," "alerts," "receipts," and "confirmations."
- **Inquiries (queries or searches):** Like "Product search," "filter," "bid history," and "item status."
- **Internal files (databases or data groups maintained by the system):** Examples include "User data," "Products," "Bids," "Payments," and "Feedback."
- **External interfaces (connections with other systems):** Such as "Payment Gateway API," "Auth API," and "Messaging API."

Each function type is assessed for complexity (low, medium, high) and assigned a weight. The total function points for the project are calculated by summing all weighted counts, which helps estimate the project size, timeline, and cost.

2. Calculating Function Points for E-Bidding Platform

Here's a simplified Function Point calculation based on the main functional components described in the document:

Function Type	Description & Example	Count	Complexity	Weight	FP (Count × Weight)
External Inputs (EI)	Screens/forms: Register, Login, Add Product, Place Bid, Upload Docs, Message	10	Medium	4	40
External Outputs (EO)	System outputs: Bid status, alerts, receipts, confirmations	7	Medium	5	35
External Inquiries (EQ)	Product search, filter, bid history, item status	5	Low	3	15
Internal Logical Files (ILF)	User data, Products, Bids, Payments,	6	Medium	7	42

	Feedback				
External Interface Files (EIF)	Payment Gateway API, Auth API, Messaging API	4	Medium	5	20

3. Sample Function Point Calculation:

- Inputs (EI): 10 × 4 = 40
- Outputs (EO): 7 × 5 = 35
- Inquiries (EQ): 5 × 3 = 15
- Internal Files (ILF): 6 × 7 = 42
- External Interfaces (EIF): 4 × 5 = 20

Unadjusted Function Points (UFP) = 40 + 35 + 15 + 42 + 20 = 152

4. Adjustments (Value Adjustment Factor - VAF):

Adjustments are made based on 14 general system characteristics like performance, security, complexity, reusability, etc. Each is rated from 0 to 5.

Assuming an average adjustment score of 3 for each:

- Sum of factor scores = 14 × 3 = 42
- VAF = 0.65 + (0.01 × 42) = 0.65 + 0.42 = 1.07

5. Final Adjusted Function Points (AFP):

- AFP = UFP × VAF = 152 × 1.07 = 162.64 ≈ **163 FP**