

Modules			
Module Name	Description	Exposed Interface Names	Interface Description
DisplayModule			
Authentication	This module provides authentication services like verifying use credentials or reset passwords.	UserAuthenticationService	This interface exposes services related to authentication i.e reads from user Database, the attempts to find a matching username and password, resetting password.
UserManagement	This module provides services related to managing current users of the auction system such as creating new users, login etc.	UserService	This interface exposes functionality related to creating new users, logging in.
CatalogServer	This module provides services regarding the management of the items/ inventory such as getting information about the items, status of an item etc.	CatalogService	This interface exposes functionality related to retrieving, creating, updating, and deleting products and categories.
AuctionServer	This module provides services related to bidding i.e placing bid, highest bid	ForwardBiddingService DutchBiddingService	ForwardBiddingService: This interface exposes functionality related to forward bidding operations. Users can create, update, and delete forward bidding sessions. Users can search for forward bidding sessions based on criteria, place bids, cancel bids, retrieve bid details, and get the

			<p>winning bid for a session.</p> <p>DutchBiddingService: This interface exposes functionality related to Dutch auction operations. Users can create, update, and delete Dutch auctions. Users can search for Dutch auctions based on criteria, place bids, and accept prices for auctions</p>
PaymentServer	This module provides services that allow users to make payments	PaymentService	This interface exposes functionality related to payment related operations
ShippingServer	This module provides services that allow users to have purchased items to be shipped to them	ShippingService	<p>This interface exposes functionality related to shipping services. Users can create shipments for bids, retrieve shipment details by receipt, and get a list of shipments associated with a user. It also includes functionality for tracking shipment status.</p>

Interfaces		
Interface Name	Operations	Operation Descriptions
UserAuthenticationService	<boolean> authenticateUser(String userName, String password)	Verifies the user's credentials and returns a boolean indicating successful authentication.

	<User> getUserByAuthToken(String authToken)	Retrieves user information based on the provided authentication token.
	<boolean> resetPassword(String userName, String email)	Allows users to reset their passwords, often through a password reset email.
UserService	<User> createUser(UserDTO userDTO)	Creates a new user account based on the provided user data. It gets the arguments from db schema.
	<boolean> handleUser(UserDTO userDTO)	Handles user login and returns a boolean indicating success.
	<boolean> isAuthorized(UserDTO userDTO)	Checks whether a user is authorized to perform specific actions.
	<User> updateUserProfile(UserDTO userDTO)	Allows users to update their profile information.
CatalogService	<Product> getProductById(Long productId)	Gets product details by its unique identifier.
	<boolean> removeItemFromAuction(Long itemId)	Allows sellers to remove an item from auction.
	void createBidSession()	Initiates a bid session for a selected item.
	<List<Product>> searchProducts(String query)	Searches for products based on a query string – keyword.
	<boolean> deleteCategory(Long categoryId)	Deletes a category from the catalog.
	<Category> updateCategory(Long categoryId, CategoryInput input)	Allows updating category information.
	<boolean> deleteProduct(Long productId)	Deletes a product from the catalog.
	<Product> updateProduct(Long productId, ProductInput input)	Allows updating product information.
	<Product> createProduct(ProductInput input)	Adds a new product to the catalog – get attributes from db schema
ForwardBiddingService	<ForwardBidding> createForwardBidding(ForwardBid	Allows users to create a new forward bidding session with the specified input

	dingInput input)	parameters. The method returns the created forward bidding session.
	<Bid> getWinningBid(String biddingId)	Retrieves the winning bid for a forward bidding session based on the bidding ID. It returns the winning bid details.
	<Bid> getBid(String biddingId, String bidId)	Retrieves the details of a specific bid within a forward bidding session based on the bidding ID and bid ID. It returns the bid information.
	<boolean> cancelBid(String biddingId, String bidId)	Allows users to cancel a previously placed bid in a forward bidding session. Returns true if the bid cancellation is successful, or false if it fails.
	<Bid> placeBid(String biddingId, BidInput input)	Permits users to place a bid on a specific forward bidding session using the provided bid input. It returns the bid that was placed.
	<ForwardBidding> getForwardBidding(String biddingId)	Retrieves the details of a forward bidding session based on the provided bidding ID and returns the corresponding forward bidding session.
	<boolean> deleteForwardBidding(String biddingId)	Permits users to delete a forward bidding session identified by its unique bidding ID. Returns true if the deletion is successful, or false if it fails.
	<ForwardBidding> updateForwardBidding(String biddingId, ForwardBiddingInput input)	Enables users to update an existing forward bidding session using the provided input parameters. It returns the updated forward bidding session.
DutchBiddingService	<DutchAuction> createDutchAuction(DutchAuctionInput input)	Allows users to create a new Dutch auction with the specified input parameters. It returns the created Dutch auction.
	<boolean> acceptPrice(Long auctionId, String bidderId)	Allows the auctioneer or seller to accept a bid price for a Dutch auction. Returns true if the bid price is accepted, or false if it fails.
	<Bid> placeBid(Long auctionId, BidInput input)	Permits users to place a bid on a specific Dutch auction using the provided bid input. It returns the bid that was placed.
	<boolean>	Permits users to delete a Dutch auction

	deleteDutchAuction(Long auctionId)	identified by its unique auction ID. Returns true if the deletion is successful, or false if it fails.
PaymentService	<Payment> createPayment(int bidId, double paymentAmount, PaymentMethod paymentMethod, PaymentDetails paymentDetails)	Enables users to create a payment for a bid with the specified payment amount, payment method, and payment details. It returns the payment confirmation.
	<Payment> getPaymentByReceipt(Receipt receipt)	Retrieves payment details based on the provided receipt and returns the corresponding payment information.
	<boolean> handlePaymentStatus(int paymentId, PaymentStatus status)	Allows handling and updating the status of a payment identified by its payment ID. Returns true if the status update is successful, or false if it fails.
ShipmentService	<Shipment> createShipment(int bidId, ShipmentMethod method)	Allows users to create a shipment for a bid using the specified shipment method. It returns the shipment details.
	<Shipment> getShipmentByReceipt(Receipt receipt)	Retrieves shipment details based on the provided receipt and returns the corresponding shipment information.