Work summary for Sprint-2

We planned these next 4 functionalities for Sprint-2:
The first sprint level includes the following functionality implementation:

1. Functional: Countdown Timers
2. Functional: Real-Time Bidding
3. Functional: Auction Configuration by Sellers
4. Functional: Outbid Notification

Summary:

Work Completed

1. Functional: Countdown Timers

- Implemented real-time countdown timer using WebSocket
- Added timer display in auction details page
- Auto-refresh functionality when auction ends
- Server synchronization for accurate timing

```
// CountdownService.java
@Service
public class CountdownService
{
   private final WebSocketService webSocketService;
   private final AuctionRepository auctionRepository;

   public void startAuctionTimer(Long auctionId)
 {
      Auction auction = auctionRepository.findById(auctionId)
         .orElseThrow(() -> new RuntimeException("Auction not found"));

      Duration timeLeft = Duration.between(LocalDateTime.now(),
auction.getEndTime());
      if (timeLeft.isNegative())
 {
         endAuction(auctionId);
         return;
      }
```

```
    webSocketService.sendMessage("/topic/auction/" + auctionId + "/timer",
      new TimerUpdate(timeLeft.getSeconds()));
  }
}
```

2. Functional: Real-Time Bidding

- Implemented WebSocket endpoints for real-time bid updates
- Added bid validation and processing
- Integrated with countdown timer
- Implemented bid history tracking

```
// BiddingController.java
@Controller
public class BiddingController
{
  @MessageMapping("/auction/{auctionId}/bid")
  @SendTo("/topic/auction/{auctionId}/bids")
  public BidResponse processBid(@DestinationVariable Long auctionId,
                @Payload BidRequest bidRequest,
                Principal principal)
{
    return biddingService.processBid(auctionId, bidRequest,
principal.getName());
  }
}
```

3. Functional: Auction Configuration by Sellers

- Implemented auction creation with customizable settings
- Added validation for auction parameters
- Created seller dashboard for auction management
- Implemented auction editing functionality

```
// AuctionService.java
@Service
public class AuctionService
 {
  public Auction createAuction(AuctionRequest request, String sellerId)
 {
    validateAuctionRequest(request);

    Auction auction = new Auction();
```

```
        auction.setStartTime(request.getStartTime());
        auction.setEndTime(request.getEndTime());
        auction.setStartingPrice(request.getStartingPrice());
        auction.setReservePrice(request.getReservePrice());
        auction.setMinimumBidIncrement(request.getMinBidIncrement());

        return auctionRepository.save(auction);
    }
}
```

4. Functional: Outbid Notification

- Implemented real-time notification system using WebSocket
- Added email notifications for outbid events
- Implemented notification preferences
- Created notification history tracking

```
// NotificationService.java
@Service
public class NotificationService
 {
    public void sendOutbidNotification(Long auctionId, String outbidUserId,
                                     BigDecimal newBidAmount)
 {
        NotificationMessage notification = new NotificationMessage(
            "You have been outbid on auction #" + auctionId,
            "A new bid of $" + newBidAmount + " has been placed"
        );
        webSocketService.sendMessage(
            "/topic/user/" + outbidUserId + "/notifications",
            notification
        );

        if (isEmailNotificationEnabled(outbidUserId))
 {
            emailService.sendOutbidEmail(outbidUserId, auctionId, newBidAmount);
        }
    }
}
```

Additional Work Brought In

1. Added WebSocket security configuration

2. Implemented connection retry mechanism
3. Added bid validation rules
4. Created comprehensive notification preferences
5. Implemented auction state management

Work Not Completed

Real-Time Bidding, implemented using Kafka, was not fully functional due to some errors we are resolving while learning. Thus, it is taking a little bit longer.

Challenges

The primary challenge faced during this sprint was shifting the server on Ubuntu VM to make it compatible with the Operating System. We encountered a lot of errors because of that.
Real-Time Bidding also had errors we are resolving in the next sprint.