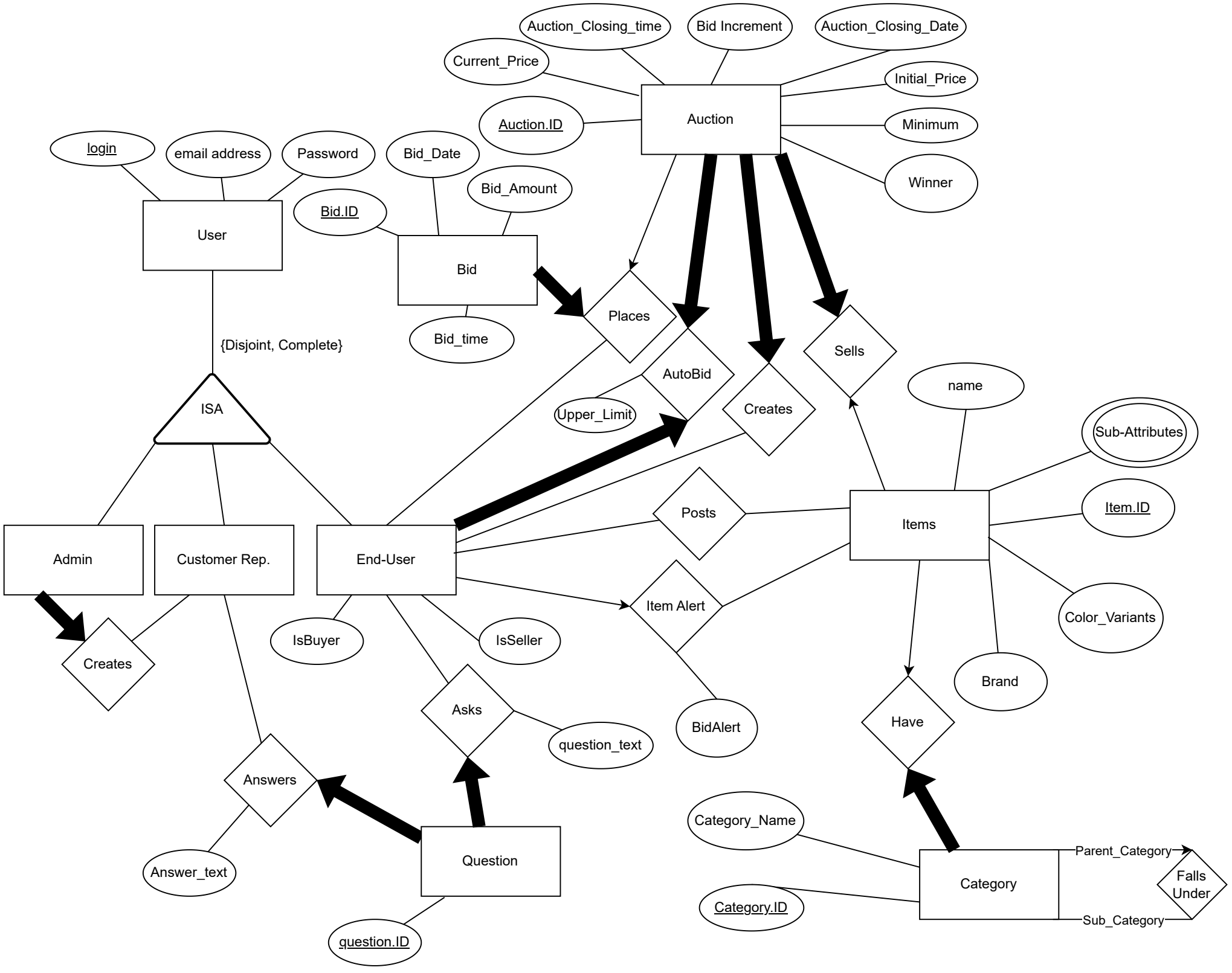
Auction_Closing_time
Bid Increment
Auction_Closing_Date
Current_Price
Initial_Price
Auction
Auction.ID
Minimum
Winner
login
email address
Password
Bid_Date
Bid_Amount
Bid.ID
User
Bid
Places
Bid_time
{Disjoint, Complete}
AutoBid
Sells
Upper_Limit
Creates
name
ISA
Sub-Attributes
Item.ID
Posts
Items
Admin
Customer Rep.
End-User
Item Alert
Color_Variants
Creates
IsBuyer
IsSeller
Brand
Asks
BidAlert
Have
question_text
Answers
Category_Name
Parent_Category
Answer_text
Question
Category
Falls Under
Category.ID
Sub_Category
question.ID

## Assumptions/Design Explanation:

1. For the 'Question' Entity, we decided this should be entity set rather than simply a relationships as per these issues:
    a. If End-User wants to ask question, they will be forces to have to set a customer Representative if 'Question' is kept as a relationship.
    b. If End-User's question is important, and may want to be stored for a Customer Rep. to use the answer to that question for another End-User, then it is useful to store these text's in a table.

2. For 'Categories' Entity using self-referencing to represent hierarchy of Parent to Sub Category.
    a. This helps in keeping generality with what names the parent and subcategories might be. If we were to use something like 'Manifestation' we would have to specify dependent Weak Entity sets and this would mean we would have to essentially 'force'/'hard define' the names of the categories being sold in 'Items'.

3. Item Alert, Auto Bid, are kept as relationships as we may need more information from Item's table or End-User table (Item Alert)/ Auction's table or End-User table (Auto Bid). This means that for query purpose, we will want foreign key relationships which keeping these as relationships enables us to do.

4. End-User --> IsBuyer, IsSeller. Reason for this being just attributes rather than another ISA relationship under End-User is that it simplifies the difficulty for this project scope rather than specifying more sub user types. If we needed to maybe do aggregation operations or more complex information for specifically Seller or Buyer, then making them into entities in a ISA relationship with End-User might be justified.

5. Sale Entity: Originally, we were thinking of adding a Sales entity with attributes for sales being made and sales report the Admin creates. The relationships to this Sales entity would've been 'Items GENERATE Sales' and 'Admin CREATES SALES REPORT Sales.' Reason for not doing so is:
    a. adding a Sales entity would be pointless since these stuff can simply be queried from the database using the other tables and relationships. If for any reason Sales Reports needs to be stored, this can be cached or stored in the runtime of the application as anyways this is a local project and when the Tomcat server dies, the application lifetime scope dies with it.

6. Originally had a Customer Rep. REMOVES Item and a Customer Rep. DELETES Bid but we decided there isn't a need to add these as essentially these are delete queries that don't need to be specified in as a relationship and these can be kept at log level (like a simple console.log/print statement). The only time we could see this being necessary is we want to store deleted bid information or item information which is not a use case in this project.

7. The increment of the AutoBid will be set to the bid increment of what the seller places acceptable for that Auction. We felt this made sense to the concept of an auction as if a user decided to make their increment whatever they wanted, it wouldn't make much sense to go above a certain increment as if increment happens to be less than the current bid, it would be useless.

8. There should be only 1 Parent Category for every Sub Category. This is set as say if a sub category was of a car or a plane, this doesn't make much sense as there is not vehicle that is a car and a plane.

9. Decided End-User will include both sellers and buyers so kept isBuyer, isSeller attributes

10. Seller can create Auction and post item listings. Buyer can place bid on Auction for item listing.

11. Kept POSTS relationship participation constraints between EndUsers and Items as many-to-many both sides as a Seller can post an item but they don't necessarily have to for creating their account.