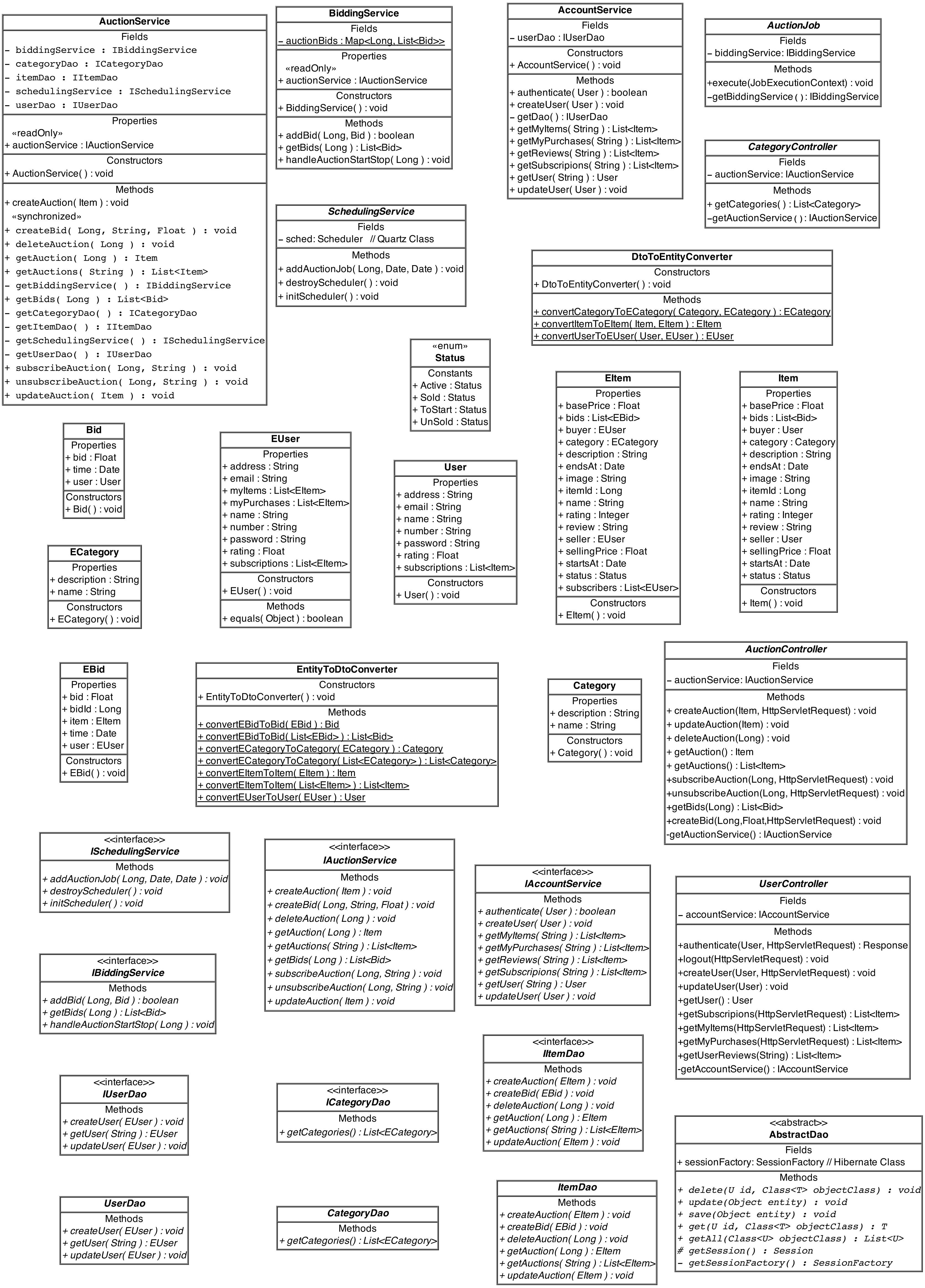

```
AuctionService
Fields
- biddingService : IBiddingService
- categoryDao : ICategoryDao
- itemDao : IItemDao
- schedulingService : ISchedulingService
- userDao : IUserDao
Properties
  «readOnly»
+ auctionService : IAuctionService
Constructors
+ AuctionService( ) : void
Methods
+ createAuction( Item ) : void
  «synchronized»
+ createBid( Long, String, Float ) : void
+ deleteAuction( Long ) : void
+ getAuction( Long ) : Item
+ getAuctions( String ) : List<Item>
- getBiddingService( ) : IBiddingService
+ getBids( Long ) : List<Bid>
- getCategoryDao( ) : ICategoryDao
- getItemDao( ) : IItemDao
- getSchedulingService( ) : ISchedulingService
- getUserDao( ) : IUserDao
+ subscribeAuction( Long, String ) : void
+ unsubscribeAuction( Long, String ) : void
+ updateAuction( Item ) : void
```

```
BiddingService
Fields
– auctionBids : Map<Long, List<Bid>>
Properties
  «readOnly»
+ auctionService : IAuctionService
Constructors
+ BiddingService( ) : void
Methods
+ addBid( Long, Bid ) : boolean
+ getBids( Long ) : List<Bid>
+ handleAuctionStartStop( Long ) : void
```

```
SchedulingService
Fields
– sched: Scheduler   // Quartz Class
Methods
+ addAuctionJob( Long, Date, Date ) : void
+ destroyScheduler( ) : void
+ initScheduler( ) : void
```

```
AccountService
Fields
– userDao : IUserDao
Constructors
+ AccountService( ) : void
Methods
+ authenticate( User ) : boolean
+ createUser( User ) : void
– getDao( ) : IUserDao
+ getMyItems( String ) : List<Item>
+ getMyPurchases( String ) : List<Item>
+ getReviews( String ) : List<Item>
+ getSubscripions( String ) : List<Item>
+ getUser( String ) : User
+ updateUser( User ) : void
```

```
AuctionJob
Fields
– biddingService: IBiddingService
Methods
+execute(JobExecutionContext) : void
–getBiddingService( ) : IBiddingService
```

```
CategoryController
Fields
– auctionService: IAuctionService
Methods
+ getCategories( ) : List<Category>
–getAuctionService( ) : IAuctionService
```

```
DtoToEntityConverter
Constructors
+ DtoToEntityConverter( ) : void
Methods
+ convertCategoryToECategory( Category, ECategory ) : ECategory
+ convertItemToEItem( Item, EItem ) : EItem
+ convertUserToEUser( User, EUser ) : EUser
```

```
«enum»
Status
Constants
+ Active : Status
+ Sold : Status
+ ToStart : Status
+ UnSold : Status
```

```
Bid
Properties
+ bid : Float
+ time : Date
+ user : User
Constructors
+ Bid( ) : void
```

```
EUser
Properties
+ address : String
+ email : String
+ myItems : List<EItem>
+ myPurchases : List<EItem>
+ name : String
+ number : String
+ password : String
+ rating : Float
+ subscriptions : List<EItem>
Constructors
+ EUser( ) : void
Methods
+ equals( Object ) : boolean
```

```
User
Properties
+ address : String
+ email : String
+ name : String
+ number : String
+ password : String
+ rating : Float
+ subscriptions : List<Item>
Constructors
+ User( ) : void
```

```
EItem
Properties
+ basePrice : Float
+ bids : List<EBid>
+ buyer : EUser
+ category : ECategory
+ description : String
+ endsAt : Date
+ image : String
+ itemId : Long
+ name : String
+ rating : Integer
+ review : String
+ seller : EUser
+ sellingPrice : Float
+ startsAt : Date
+ status : Status
+ subscribers : List<EUser>
Constructors
+ EItem( ) : void
```

```
Item
Properties
+ basePrice : Float
+ bids : List<Bid>
+ buyer : User
+ category : Category
+ description : String
+ endsAt : Date
+ image : String
+ itemId : Long
+ name : String
+ rating : Integer
+ review : String
+ seller : User
+ sellingPrice : Float
+ startsAt : Date
+ status : Status
Constructors
+ Item( ) : void
```

```
ECategory
Properties
+ description : String
+ name : String
Constructors
+ ECategory( ) : void
```

```
EBid
Properties
+ bid : Float
+ bidId : Long
+ item : EItem
+ time : Date
+ user : EUser
Constructors
+ EBid( ) : void
```

```
EntityToDtoConverter
Constructors
+ EntityToDtoConverter( ) : void
Methods
+ convertEBidToBid( EBid ) : Bid
+ convertEBidToBid( List<EBid> ) : List<Bid>
+ convertECategoryToCategory( ECategory ) : Category
+ convertECategoryToCategory( List<ECategory> ) : List<Category>
+ convertEItemToItem( EItem ) : Item
+ convertEItemToItem( List<EItem> ) : List<Item>
+ convertEUserToUser( EUser ) : User
```

```
Category
Properties
+ description : String
+ name : String
Constructors
+ Category( ) : void
```

```
AuctionController
Fields
– auctionService: IAuctionService
Methods
+ createAuction(Item, HttpServletRequest) : void
+ updateAuction(Item) : void
+ deleteAuction(Long) : void
+ getAuction() : Item
+ getAuctions() : List<Item>
+subscribeAuction(Long, HttpServletRequest) : void
+unsubscribeAuction(Long, HttpServletRequest) : void
+getBids(Long) : List<Bid>
+createBid(Long,Float,HttpServletRequest) : void
-getAuctionService() : IAuctionService
```

```
<<interface>>
ISchedulingService
Methods
+ addAuctionJob( Long, Date, Date ) : void
+ destroyScheduler( ) : void
+ initScheduler( ) : void
```

```
<<interface>>
IAuctionService
Methods
+ createAuction( Item ) : void
+ createBid( Long, String, Float ) : void
+ deleteAuction( Long ) : void
+ getAuction( Long ) : Item
+ getAuctions( String ) : List<Item>
+ getBids( Long ) : List<Bid>
+ subscribeAuction( Long, String ) : void
+ unsubscribeAuction( Long, String ) : void
+ updateAuction( Item ) : void
```

```
<<interface>>
IAccountService
Methods
+ authenticate( User ) : boolean
+ createUser( User ) : void
+ getMyItems( String ) : List<Item>
+ getMyPurchases( String ) : List<Item>
+ getReviews( String ) : List<Item>
+ getSubscripions( String ) : List<Item>
+ getUser( String ) : User
+ updateUser( User ) : void
```

```
UserController
Fields
– accountService: IAccountService
Methods
+authenticate(User, HttpServletRequest) : Response
+logout(HttpServletRequest) : void
+createUser(User, HttpServletRequest) : void
+updateUser(User) : void
+getUser() : User
+getSubscripions(HttpServletRequest) : List<Item>
+getMyItems(HttpServletRequest) : List<Item>
+getMyPurchases(HttpServletRequest) : List<Item>
+getUserReviews(String) : List<Item>
-getAccountService() : IAccountService
```

```
<<interface>>
IBiddingService
Methods
+ addBid( Long, Bid ) : boolean
+ getBids( Long ) : List<Bid>
+ handleAuctionStartStop( Long ) : void
```

```
<<interface>>
IItemDao
Methods
+ createAuction( EItem ) : void
+ createBid( EBid ) : void
+ deleteAuction( Long ) : void
+ getAuction( Long ) : EItem
+ getAuctions( String ) : List<EItem>
+ updateAuction( EItem ) : void
```

```
<<interface>>
IUserDao
Methods
+ createUser( EUser ) : void
+ getUser( String ) : EUser
+ updateUser( EUser ) : void
```

```
<<interface>>
ICategoryDao
Methods
+ getCategories() : List<ECategory>
```

```
<<abstract>>
AbstractDao
Fields
+ sessionFactory: SessionFactory // Hibernate Class
Methods
+ delete(U id, Class<T> objectClass) : void
+ update(Object entity) : void
+ save(Object entity) : void
+ get(U id, Class<T> objectClass) : T
+ getAll(Class<U> objectClass) : List<U>
# getSession() : Session
- getSessionFactory() : SessionFactory
```

```
UserDao
Methods
+ createUser( EUser ) : void
+ getUser( String ) : EUser
+ updateUser( EUser ) : void
```

```
CategoryDao
Methods
+ getCategories() : List<ECategory>
```

```
ItemDao
Methods
+ createAuction( EItem ) : void
+ createBid( EBid ) : void
+ deleteAuction( Long ) : void
+ getAuction( Long ) : EItem
+ getAuctions( String ) : List<EItem>
+ updateAuction( EItem ) : void
```