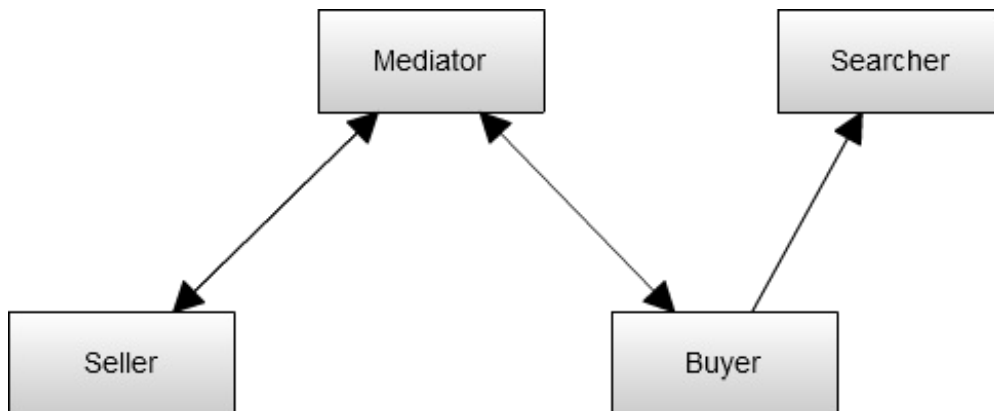


# Gaia

- Linus Oleander - F120180
- Daniel Tell - F120181

## Agents



## Roles

### Seller

#### Description

Sells items on market

#### Permissions

- changes users

#### Responsibilities

#### Liveness

```
Seller = (  
  createAuction |  
  notifyAboutEndedAuction.contactHigestBidderAction  
) *
```

### Bidder

#### Description

Bid items on market

#### Permissions

- changes users

#### Responsibilities

#### Liveness

```
Bidder = (  
  searchForItem.makeBid |  
  makeBid |  
  notifyNotHighestBidder.makeBid* |  
  notifyWinnerOfAuction.contactSellerAction  
) *
```

## Searcher

### Description

Search for items

### Permissions

- reads auctions

### Responsibilities

### Liveness

```
Searcher = (searchForItem.findItem.respondWithSearchResultAction) *
```

## Mediator

### Description

Our platform auctioneer

### Permissions

- changes auctions
- changes bids

### Responsibilities

### Safety

- exists(auction\_id)
- exists(winner\_id)
- exists(seller\_id)
- exists(bidder\_id)
- description.length > 0
- min\_price > 0
- price > Auction.highest\_bid

### Liveness

```
Mediator = (  
  checkForEndedAuctions.(  
    notifyWinnerOfAuction |  
    notifyLoserOfAuction |  
    notifyNotHighestBidder |  
    notifyAboutEndedAuction  
  ) |  
  makeBid.createBidAction.respondWithBidStatusAction |  
  createAuction.createAuctionAction.respondWithAuctionStatusAction  
) *
```

## Protocol

- notifyWinnerOfAuction()
  - Description: Called when bidder wins an auction
  - Initiator: Mediator
  - Partner: Bidder
  - Input: {auction\_id: Integer, price: Integer}
  - Output: Void
- notifyLoserOfAuction()
  - Description: Called when bidder loses an auction
  - Initiator: Mediator

- Partner: Bidder
  - Input: *{auctionid: Integer, price: Integer, winnerid: Integer}*
  - Output: Void
- notifyNotHighestBidder()
  - Description: Called when bidder doesn't have the highest bid
  - Initiator: Mediator
  - Partner: Bidder
  - Input: *{auction\_id: Integer, price: Integer}*
  - Output: Void
- makeBid()
  - Description: Bid on an item
  - Initiator: Bidder
  - Partner: Mediator
  - Input: *{auctionid: Integer, price: Integer, bidderid: Integer}*
  - Output: *{status: STATUS}*
- searchForItem()
  - Description: Search for an item
  - Initiator: Bidder
  - Partner: Searcher
  - Input: *{query: String}*
  - Output: *{item\_id: Integer}*
- notifyAboutEndedAuction()
  - Description: Information about an ended auction
  - Initiator: Mediator
  - Partner: Seller
  - Input: *{auctionid: Integer, winnerid: Integer, price: Integer}*
  - Output: Void
- notifySellerAboutNewBid()
  - Description: Notify seller about new bids on auction
  - Initiator: Mediator
  - Partner: Seller
  - Input: *{auctionid: Integer, bidderid: Integer, price: Integer}*
  - Output: Void
- createAuction()
  - Description: Create auction for item
  - Initiator: Seller
  - Partner: Mediator
  - Input: *{description: String, sellerid: Integer, minprice: Integer}*
  - Output: Void