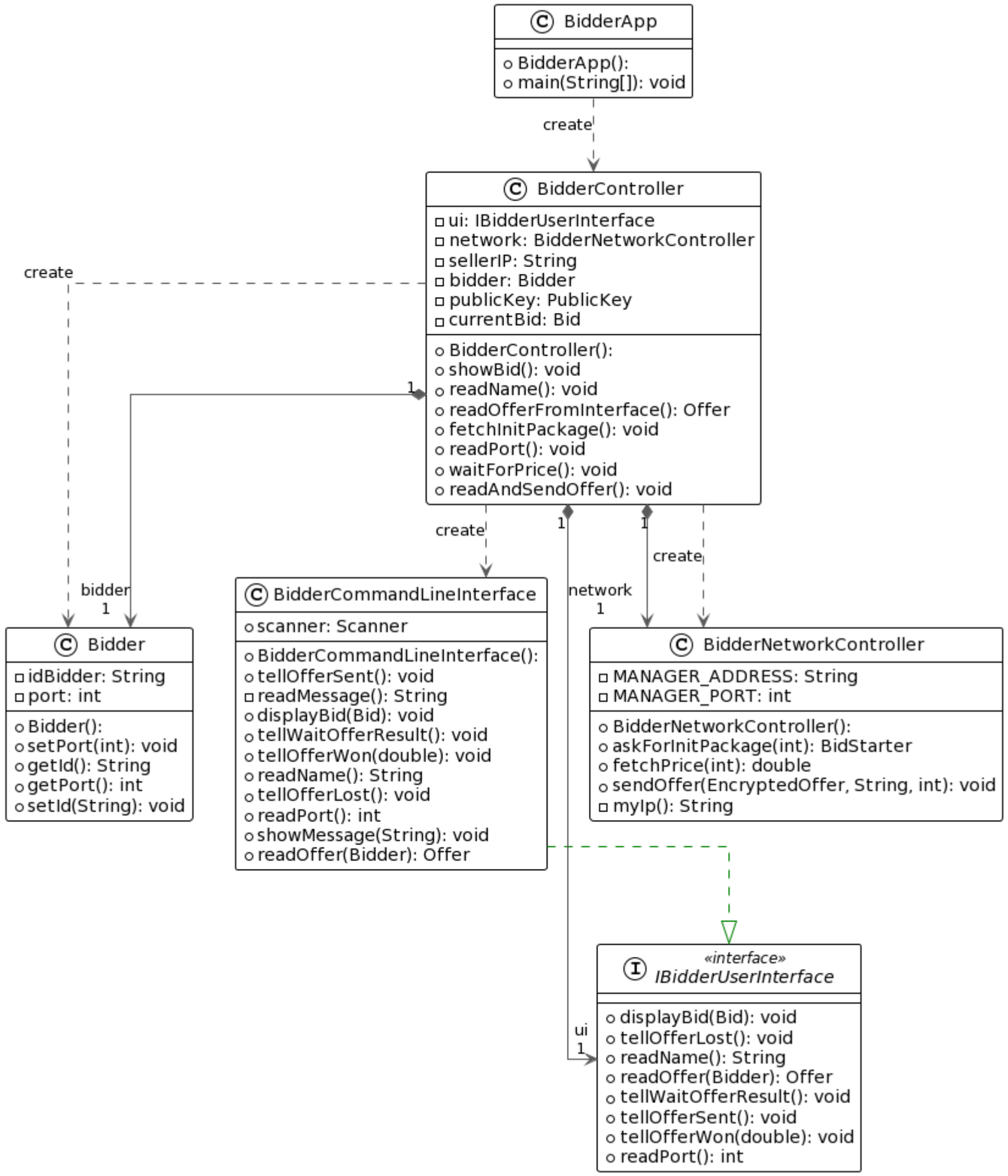
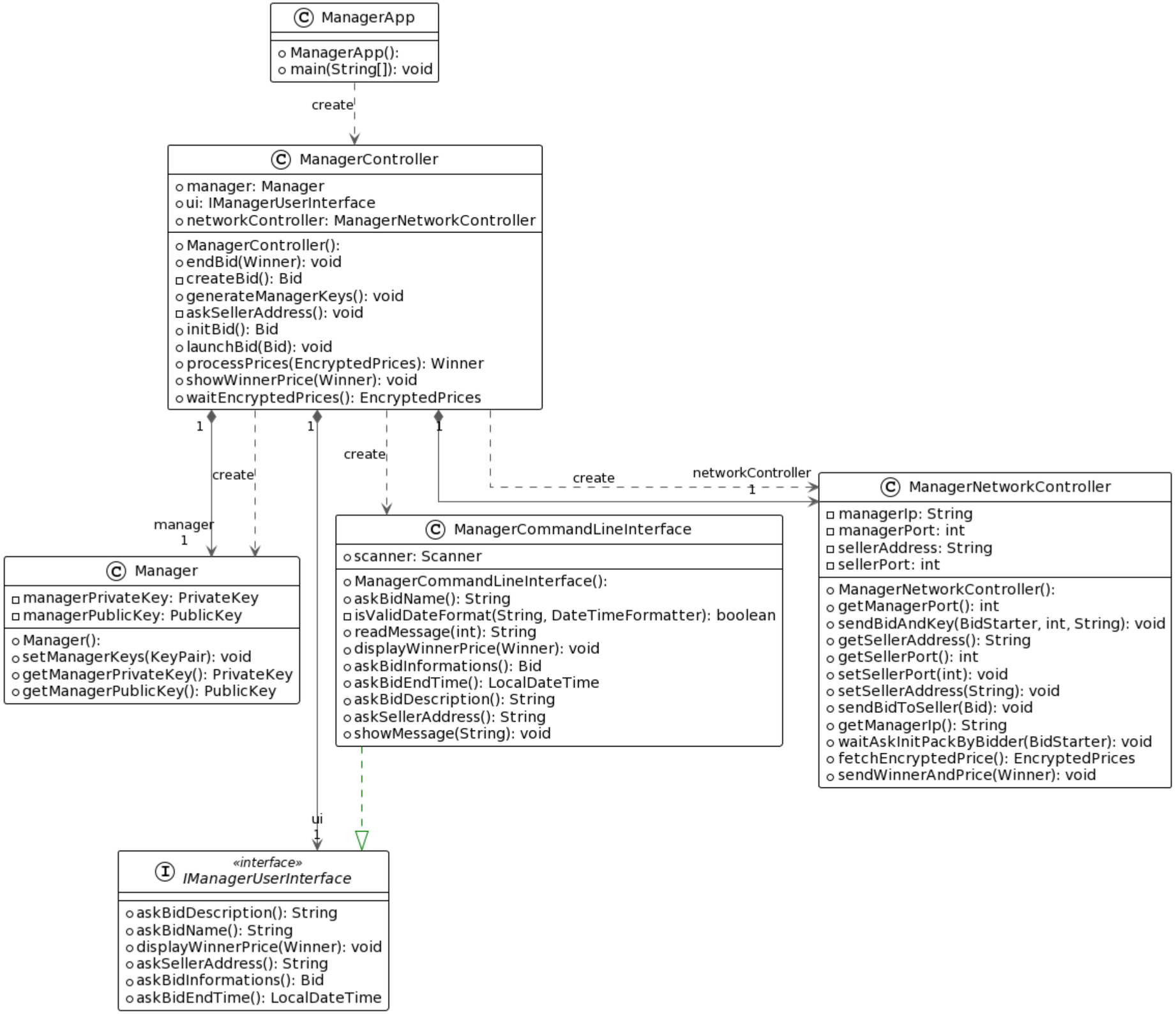
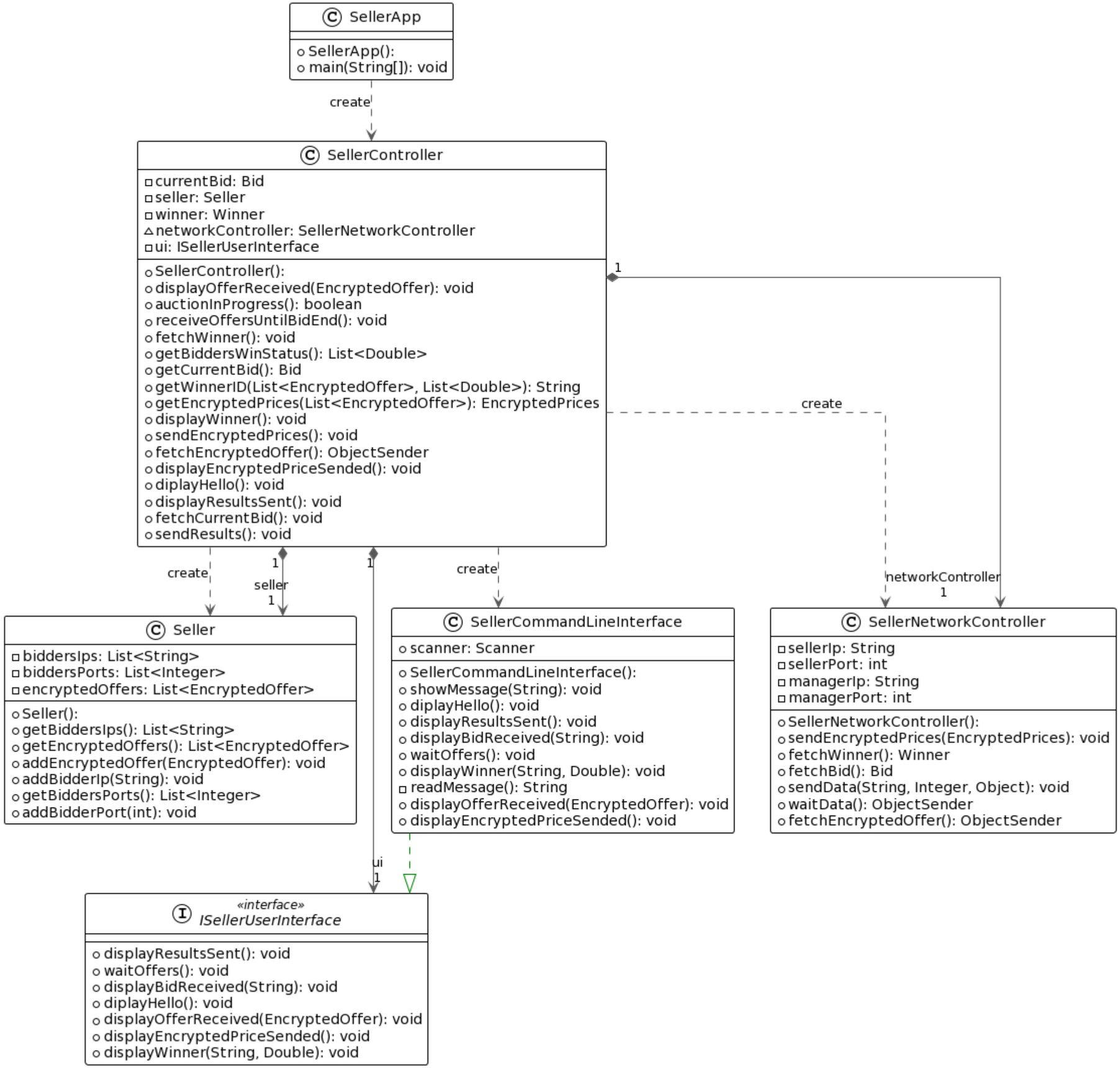

BidderApp
BidderApp():
main(String[]): void
create
BidderController
ui: IBidderUserInterface
network: BidderNetworkController
sellerIP: String
bidder: Bidder
publicKey: PublicKey
currentBid: Bid
BidderController():
showBid(): void
readName(): void
readOfferFromInterface(): Offer
fetchInitPackage(): void
readPort(): void
waitForPrice(): void
readAndSendOffer(): void
create
1
bidder
1
Bidder
idBidder: String
port: int
Bidder():
setPort(int): void
getId(): String
getPort(): int
setId(String): void
create
BidderCommandLineInterface
scanner: Scanner
BidderCommandLineInterface():
tellOfferSent(): void
readMessage(): String
displayBid(Bid): void
tellWaitOfferResult(): void
tellOfferWon(double): void
readName(): String
tellOfferLost(): void
readPort(): int
showMessage(String): void
readOffer(Bidder): Offer
1
network
1
1
create
BidderNetworkController
MANAGER_ADDRESS: String
MANAGER_PORT: int
BidderNetworkController():
askForInitPackage(int): BidStarter
fetchPrice(int): double
sendOffer(EncryptedOffer, String, int): void
myIp(): String
«interface»
IBidderUserInterface
displayBid(Bid): void
tellOfferLost(): void
readName(): String
readOffer(Bidder): Offer
tellWaitOfferResult(): void
tellOfferSent(): void
tellOfferWon(double): void
readPort(): int
ui
1


ManagerApp
ManagerApp():
main(String[]): void
create
ManagerController
manager: Manager
ui: IManagerUserInterface
networkController: ManagerNetworkController
ManagerController():
endBid(Winner): void
createBid(): Bid
generateManagerKeys(): void
askSellerAddress(): void
initBid(): Bid
launchBid(Bid): void
processPrices(EncryptedPrices): Winner
showWinnerPrice(Winner): void
waitEncryptedPrices(): EncryptedPrices
1
1
1
create
create
create
networkController
1
manager
1
ui
1
Manager
managerPrivateKey: PrivateKey
managerPublicKey: PublicKey
Manager():
setManagerKeys(KeyPair): void
getManagerPrivateKey(): PrivateKey
getManagerPublicKey(): PublicKey
ManagerCommandLineInterface
scanner: Scanner
ManagerCommandLineInterface():
askBidName(): String
isValidDateFormat(String, DateTimeFormatter): boolean
readMessage(int): String
displayWinnerPrice(Winner): void
askBidInformations(): Bid
askBidEndTime(): LocalDateTime
askBidDescription(): String
askSellerAddress(): String
showMessage(String): void
ManagerNetworkController
managerIp: String
managerPort: int
sellerAddress: String
sellerPort: int
ManagerNetworkController():
getManagerPort(): int
sendBidAndKey(BidStarter, int, String): void
getSellerAddress(): String
getSellerPort(): int
setSellerPort(int): void
setSellerAddress(String): void
sendBidToSeller(Bid): void
getManagerIp(): String
waitAskInitPackByBidder(BidStarter): void
fetchEncryptedPrice(): EncryptedPrices
sendWinnerAndPrice(Winner): void
«interface»
IManagerUserInterface
askBidDescription(): String
askBidName(): String
displayWinnerPrice(Winner): void
askSellerAddress(): String
askBidInformations(): Bid
askBidEndTime(): LocalDateTime


SellerApp
SellerApp():
main(String[]): void
create
SellerController
currentBid: Bid
seller: Seller
winner: Winner
networkController: SellerNetworkController
ui: ISellerUserInterface
SellerController():
displayOfferReceived(EncryptedOffer): void
auctionInProgress(): boolean
receiveOffersUntilBidEnd(): void
fetchWinner(): void
getBiddersWinStatus(): List<Double>
getCurrentBid(): Bid
getWinnerID(List<EncryptedOffer>, List<Double>): String
getEncryptedPrices(List<EncryptedOffer>): EncryptedPrices
displayWinner(): void
sendEncryptedPrices(): void
fetchEncryptedOffer(): ObjectSender
displayEncryptedPriceSended(): void
diplayHello(): void
displayResultsSent(): void
fetchCurrentBid(): void
sendResults(): void
1
create
networkController
1
create
1
seller
1
1
ui
1
Seller
biddersIps: List<String>
biddersPorts: List<Integer>
encryptedOffers: List<EncryptedOffer>
Seller():
getBiddersIps(): List<String>
getEncryptedOffers(): List<EncryptedOffer>
addEncryptedOffer(EncryptedOffer): void
addBidderIp(String): void
getBiddersPorts(): List<Integer>
addBidderPort(int): void
SellerCommandLineInterface
scanner: Scanner
SellerCommandLineInterface():
showMessage(String): void
diplayHello(): void
displayResultsSent(): void
displayBidReceived(String): void
waitOffers(): void
displayWinner(String, Double): void
readMessage(): String
displayOfferReceived(EncryptedOffer): void
displayEncryptedPriceSended(): void
SellerNetworkController
sellerIp: String
sellerPort: int
managerIp: String
managerPort: int
SellerNetworkController():
sendEncryptedPrices(EncryptedPrices): void
fetchWinner(): Winner
fetchBid(): Bid
sendData(String, Integer, Object): void
waitData(): ObjectSender
fetchEncryptedOffer(): ObjectSender
«interface»
ISellerUserInterface
displayResultsSent(): void
waitOffers(): void
displayBidReceived(String): void
diplayHello(): void
displayOfferReceived(EncryptedOffer): void
displayEncryptedPriceSended(): void
displayWinner(String, Double): void