Launcher
main()

TemporaryTrade
- meetingDateTime2 : LocalDateTime
- meetingLocation2 : String
- transactionConfirmed2 : boolean[]
- TIME_LIMIT : int

+ getMeetingDateTime2() : LocalDateTime
+ getMeetingLocation2() : String
+ getFinalMeetingDateTime() : LocalDate
+ confirmTransaction1 (username : String) : void
+ confirmTransaction2(username : String) : void
+ getUserTransactionConfirmation2(username : String) : boolean
+ hasSecondMeeting() : boolean
+ toString(currentUsername : String) : String
+ compareTo(trade : Trade) : int

<<Abstract>> Trade
- involvedUsernames: String[]
- involvedItemIDs: long[]
- isComplete: boolean
- isCancelled: boolean
- hasAgreedMeeting1: boolean
- numEdits1: int[]
- lastEditor: String
- meetingDateTime1: LocalDateTime
- meetingLocation1: String
- transactionConfirmed1: boolean[]

+ getInvolvedUsernames() : String[]
+ setIsCancelled() : void
+ getInvolvedItemIDs() : long[]
+ isInvolved(username: String) : boolean
+ getLentItemID(username: String) : long
+ getOtherUsername(username: String) : String
+ getIsComplete() : boolean
+ getIsCancelled() : boolean
+ getHasAgreedMeeting1() : boolean
+ confirmAgreedMeeting1() : void
+ getMeetingDateTime1() : LocalDateTime
+ setMeetingDateTime1(dateTime: LocalDateTime) : void
+ getMeetingLocation1() : String
+ setMeetingLocation1(location: String) : void
+ getLastEditor() : String
+ getUserTransactionConfirmation1(username: String) : boolean
+ confirmTransaction1 (username: String) : void
+ closeTransaction() : void
+ addUserEditCount1(username: String) : void
+ getUserEditCount1(username: String) : int
+ setLastEditor(username: String) : void
+ *getFinalMeetingDateTime() : LocalDateTime*
+ *toString(currentUsername: String) : String*
+ *compareTo(Trade) : int*

StartMenu
- userInput : int

+ getUserInput() : int

LoginSystem
- username : String
- email : String
- validPw : String
- user : User
- userManager : UserManager
- isAdmin : boolean
- optionInput : String

+ getUser() : user

SystemContoller
- ug : UserGateway
- ig : ItemGateway
- tg : TradeGateway
- userManager : UserManager
- itemManager : ItemManager
- tradeManager : TradeManager
- userManagerPath : String
- itemManagerPath : String
- tradeManagerPath : String
- sp : SystemPresenter

- tryRead() : void
- tryWrite() : void

ItemGateway
+ readFromFile(filePath: String) : ItemManager
+ saveToFile(filePath: String, im: ItemManager) : void

PermanentTrade
+ confirmTransaction1 (String) : void
+ getFinalMeetingDateTime() : LocalDateTime
+ toString(String) : String
+ compareTo(Trade) : int

AdminDashboard
- currentAdmin : AdminUser
- input : int
- itemManager : ItemManager
- userManager : userManager

NormalDashboard
- currentUser : NormalUser
- input : int
- itemManager : ItemManager
- userManager : UserManager
- tradeManager : TradeManager

See design2.pdf

See design3.pdf

TradeManager
- allTrades : List<Trade>
- cancelledUsers : List<String>

+ TradeManager()
+ getAllTrades() : List<Trade>
+ getAllTempTrades() : List<TemporaryTrade>
- getAllOngoingNotCancelledTrades() : List<Trade>
+ getTimesLent(username: String) : int
+ getTimesBorrowed(username: String) : int
+ cancelAllUnconfirmedTrades() : void
+ addCancelledUsers(users: String[]) : void
+ getCancelledUsers(): List<String>
+ clearCancelledUsers() : void
+ getItemInCancelledTrade(item: Item) : boolean
+ getAllPermTrades() : List<PermanentTrade>
+ createTempTrade(usernames: String[], itemIDs: long[], firstDateTime: LocalDateTime, firstLocation: String): TemporaryTrade
+ createPermTrade(usernames: String[], itemIDs: long[], firstDateTime: LocalDateTime, firstLocation: String): PermanentTrade
+ removeTrade(tradeToRemove: Trade): void
+ getOngoingTrades(username: String) : List<Trade>
- getCompletedTrades(username: String) : List<Trade>
+ getAllTrades(username: String) : List<Trade>
+ getRecentThreeTrades(username: String) : Trade[]
+ getNumMeetingsThisWeek(username: String, date: LocalDate) : int
+ getFrequentTradePartners(username: String) : String[]

UserGateway
+ readFromFile(filePath : String) : UserManager
+ saveToFile(filePath : String, um : UserManager) : void

ItemManager
- approvedItems : List<Item>
- pendingItems : List<Item>

+ ItemManager()
+ createItem(name: String, description: String, owner: String) : long
+ getApprovedItems() : List<Item>
+ getApprovedItems(username: String) : List<Item>
+ getPendingItems : List<Item>
+ getAllItems() : List<Item>
+ getNumApprovedItems() : int
+ getNumApprovedItems(username: String): int
+ getNumPendingItems() : int
+ getApprovedItem(itemID: long) : Item
+ getApprovedItem(index: int) : Item
+ getApprovedItem(username: String, index: int): Item
+ getAvailableItems(items: List<Item>) : List<Item>
+ getPendingItem(itemID: long) : Item
+ getPendingItem(index: int) : Item
+ idGetItem(items: List<Item>, itemID: long) : Item
+ getApprovedItemsByIDs(itemIDs: List<Long>) : List<Item>
+ getPendingItemsByIDs(itemIDs: List<Long>) : List<Item>
+ approveItem(itemToApprove: Item) : void
+ rejectItem(itemToReject: Item) : void

TradeGateway
+ readFromFile(filePath: String) : TradeManager
+ saveToFile(filePath: String, tm: TradeManager) : void

UserManager
- allNormals : List<NormalUser>
- allAdmins : List<AdminUser>
- usernamesToFreeze : List<String>
- unfreezeRequests : List<String>

+ UserManager()
+ createNormalUser(username: String, email: String, password: String): void
+ createAdminUser(username: String, email: String, password: String): void
+ getNormalByUsername(username: String) : NormalUser
+ getAdminByUsername(username: String) : AdminUser
+ getNormalByEmail(email: String) : NormalUser
+ getAdminByEmail(email: String) : AdminUser
+ getAllUsers() : List<Users>
+ getAllNormals() : List<NormalUser>
+ getAllAdmins() : List<AdminUser>
+ adminUsernamePassword(username: String) : String
+ normalUsernamePassword(username: String) : String
+ usernamePassword(username: String) : String
+ emailPassword(email: String) : String
+ adminEmailPassword(email: String) : String
+ normalEmailPassword(email: String) : String
+ emailExists(email: String) : boolean
+ emailExists(email: String, isAdmin: boolean) : boolean
+ usernameExists(username: String) : boolean
+ usernameNotExists(username: String, isAdmin: boolean) : boolean
+ getUsernamesToFreeze() : List<String>
+ addUsernamesToFreeze() : void
+ clearUsernamesToFreeze() : void
+ getNextAdminID() : int
+ getUnfreezeRequests() : List<NormalUser>
+ getNumUnfreezeRequest() : int
+ removeUnfreezeRequests(index: int) : void
+ addUnfreezeRequest(username: String) : void

Item
- id : long
- name : String
- description : String
- ownerUsername : String
- isAvailable : boolean
- isRemoved : boolean

+ Item(name: String, description: String, ownerUsername: String)
+ assignID() : void
+ getID() : long
+ getName() : String
+ getDescription() : String
+ getOwnerUsername() : String
+ getAvailability() : boolean
+ setAvailability(newAvailability: boolean) : void
+ getIsRemoved() : boolean
+ setIsRemoved(status: boolean) : void
+ toString() : String

NormalUser
- inventory : List<Long>
- pendingInventory : List<Long>
- wishlist : List<Long>
- tradeRequests : Map<String [], long []>
- isFrozen : boolean
- numIncomplete : int
- weeklyTradeMax : int
- meetingEditMax : int
- lendMinimum : int
- incompleteMax : int

+ getTimesBorrowed() : int
+ removeTradeRequests(key: String[]) : void
+ increaseNumIncomplete() : void
+ getNumIncomplete() : int
+ getInventory() : List<Long>
+ addInventory(itemToAdd: long) : void
+ removeInventory(itemToRemove: long) : void
+ getPendingInventory() : List<Long>
+ addPendingInventory(itemToAdd: long) : void
+ removePendingInventory(itemToRemove: long) : void
+ getWishlist() : List<Long>
+ addWishlist(itemToAdd: long) : void
+ removeWishlist(itemToRemove: long) : void
+ getIsFrozen() : boolean
+ freeze() : void
+ unfreeze() : void
+ addTradeRequests(usernames: String[], itemIDs: long[]) : void
+ getTradeRequest() : Map<String[], long[]>
+ getWeeklyTradeMax() : int
+ setWeeklyTradeMax(newMax: int) : void
+ getMeetingEditMax() : int
+ setMeetingEditMax(newMax: int) : void
+ getLendMinimum() : int
+ setLendMinimum(newMin: int) : void
+ getIncompleteMax() : int
+ setIncompleteMax(newMax: int) : void
+ isRequestedInTrade(itemID: long): boolean

AdminUser
- adminID

+ getAdminID() : int

<<Abstract>> User
- username : String
- email : String
- password : String

+ getUsername() : String
+ getEmail() : String
+ getPassword() : String

SignUpSystem
- username : String
- email : String
- password : String
- userManager : UserManager
- sp : SystemPresenter

+ createNewNormal() : NormalUser
+ createNewAdmin() : void
- inputProcess() : void