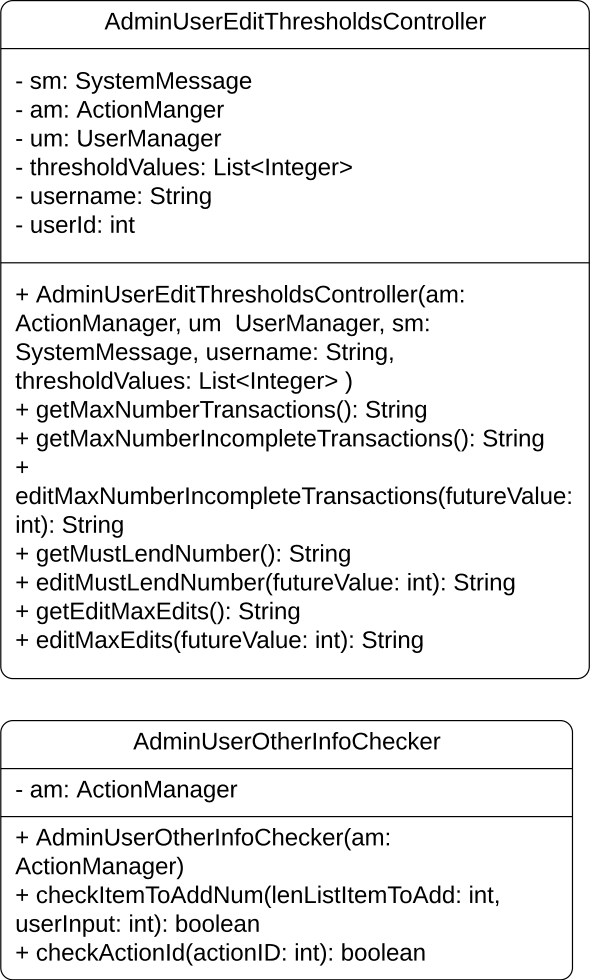
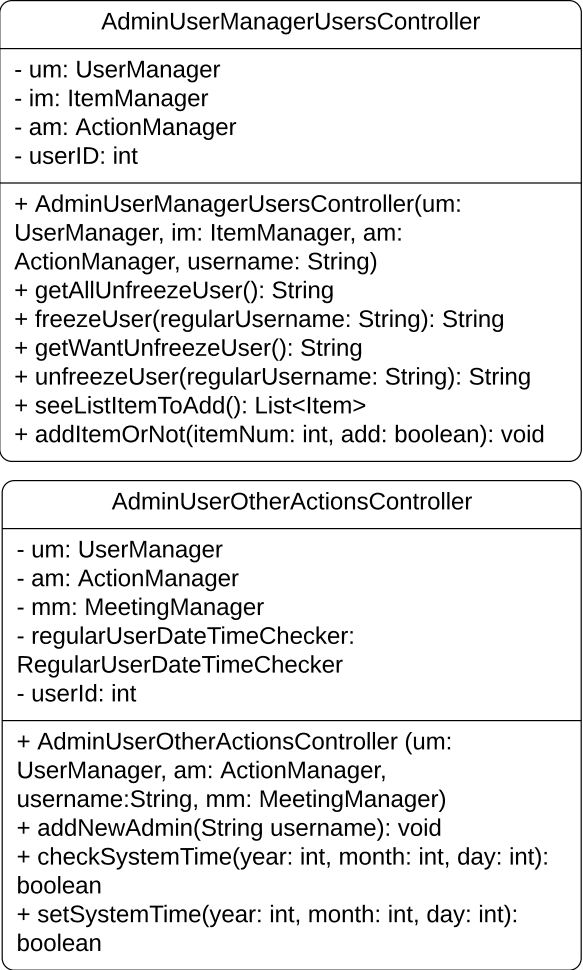


UML for AdminUser Controllers



UML for Regular User Controllers

RegularUserSearchingMenuController
<div>- tm: TradeManager -mm: MeetingManager -um: UserManager -im:ItemManager -fm: FeedBackManager -am:ActionManager -userId:int</div>
<div>+RegularUserSearchingMenuController(tm:Trademanager, mm:MeetingManager, am:ActionManager, um:UserManager, im:ItemManager, username:String) sortRating():List<TradableUser> recentThreePartner():List<Integer> filterCompleteTrade():List<Trade> sortAllTradedPartner():List<Integer> filterIncompleteTrade():List<Trade> allMeetingSortByDate():List<Meeting> completeMeetingSortByDate():List<Meeting> unCompleteMeetingSortByDate():List<Meeting> filterByCategory(category: Category):ArrayList<Item> searchItemByName(name:String):ArrayList<Item> getItemById(itemId: int): String getItemObjectById(id: int): Item sortItemByFollows(): ArrayList<Item></div>

RegularUserMeetingMenuController
<div>- tm: TradeManager - mm: MeetingManager - am: ActionManager -userId: int</div>
<div>+ RegularUserMeetingMenuController(tm: TradeManager, mm: MeetingManager, am: ActionManager, userId: int) + getUnconfirmedMeeting(): List<Meeting> + getCompleteMeetings(): List<Meeting> + isEmpty(meetings: List<Meeting>): boolean + checkValidMeeting(tradeld: int, numMeeting: int): boolean + confirmMeetingTookPlace(tradeld: int, numMeeting: int, maxMeetingTimePlaceEdit: int): boolean + getUnconfirmTimePlace(): List<Meeting> + confirmMettingTandP(tradeld: int, numMeeting: int, maxMeetingTimePlaceEdit: int): boolean + editMeetingTandP(tradeld :int, numMeeting: int, time: List<Integer>, place: String, maxMeetingTimePlaceEdit:int): boolean + checkOverEdit(tradeld: int, numMeeting: int, maxMeetingTimePlaceEdit: int) : String</div>

RegularUserCommunityMenuController
<div>- um: UserManager - am: ActionManager - fm: FeedbackManager - messageManager: MessageManager - username:String - userId: int</div>
<div>+ RegularUserCommunityMenuController(um: UserManager, am: ActionManger, fm: FeedbackManager, messageManager: MessengerManager, username: String) + getUserId(): int + reviewUser(userToReview: int, rating: int, reason: String): boolean + reportUser(userToReport: int, readon: String): boolean + findRatingForUser(id: int): double + getAllReviews(userId: int): ArrayList<Review> + checkUserId(id: int): boolean + seeUsersInSameHC(): ArrayList<TradableUser> + getFriends(): ArrayList<TradableUser> + getNotFriends(): ArrayList<TradableUser> + sendFriendRequest(userToId: int, msg: String):boolean + getFriendRequest(): HashMap<TradableUser, String> + checkIdInRequest(userId1): boolean + addFriend(id1: int): boolean + unfriendUser(id: int): boolean + checkIsFriend(userId1: int): boolean + sendMessage(receiverId: int, message: String): boolean + getAllMessages(): ArrayList<Message></div>

RegularUserThresholdController
<div>- mm: MeetingManager - um: UserManager - username: String - userId: int</div>
<div>+ public RegularUserThresholdController(mm: MeetingManager, um: UserManager, username: String) + reassessNumTransactionsLeftForTheWeek(maxNum TransactionAllowedAWeek: int) : void # changeNumTradesLeftForTheWeek(): void # isFirstDayOfTheWeek(): boolean + freezeUserOrNot(maxNumTransactionIncomplete: int): boolean - numUncompletedTransactions(): int</div>

RegularUserAccountMenuController
<div>- sm: SystemMessage - tm: TradeManager - um: UserManager - im: ItemManager - am: ActionManager - username: String - userId: int</div>
<div>+ RegularUserAccountMenuController(tm: TradeManager, um: UserManager, im: ItemManager, am: ActionManager, username: String, userId: int) + getTradables(): ArrayList<Item> + getUserId(): int + getAllTradableFromOtherNotInWishlist(): ArrayList<Item> + addToWishList(templtemId: int): boolean + removeFromWishlist(templtemId: int): boolean + removeFromInventory(templtemId: int): boolean + requestAddItem(itemName: String, description: String, category: Category): void + seeMostRecentThreeItems(): ArrayList<Item> + getWishList(): ArrayList<Item> + getInventory(): ArrayList<Item> + getTradableItems(): ArrayList<Item> + getNotTradableItems(): ArrayList<Item> + setTradableBasedOnResponses(itemId: int, option: boolean): boolean + getSuggestItemtoLend(lendToUserId: int): ArrayList<Item> + getRandomSuggestion(lendtoUserId: int): ArrayList<Item> + requestToUnfreeze(msg: String): boolean + setOnVacationStatus(newStatus: boolean): void + changeUserHC(newHC: String): String + followAnUser(userToFollowUserId: int): boolean + followAnItem(itemToFollowId: int): boolean + seeRecentStatusOfFollowedUser(): ArrayList<String> + seeRecentStatusOfFollowedItem(): ArrayList<String> + seeOwnRevocableAction(): ArrayList<Action> + requestUndoARevocableAction(actionId: int): void + selfFrozen(): boolean + selfOnVacation(): boolean</div>