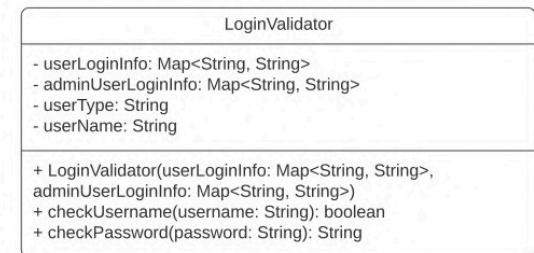
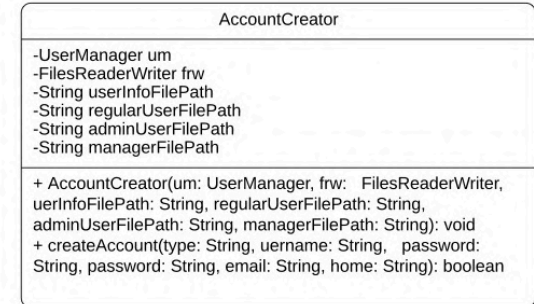
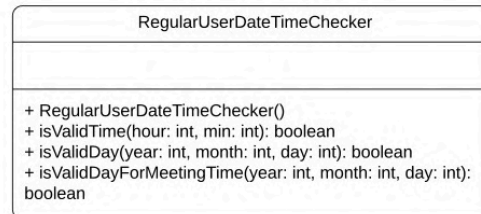
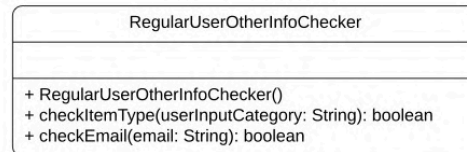
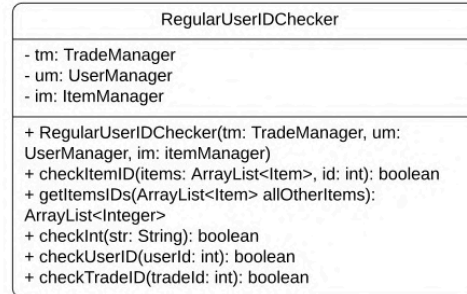


UML for Regular User Controllers



UML for Gateway

FilesReaderWriter
+ FilesReaderWriter() + readFromMenu(filePath: String): String + readThresholdValuesFromCSVFile(filePath: String): List<Integer> + saveThresholdValuesToCSVFile(thresholdValues: List<Integer>, filePath: String): void + saveUserInfoToCSVFile(filePath: String, username: String, password: String , email: String): void + readUserInfoFromCSVFile(filePath: String): Map<String, String> + helper_check_file_empty_or_not(filePath: String): boolean + readManagersFromFile(filePath: String): Object + saveManagersToFile(manager: Object, filePath: String): void + MenuLength(filePath: String): int # check_file_empty_or_not(filePath: String): boolean

SystemMessage
- pom: PrintObjectMessage - uam: UserAlertMessage - asm: AccountSystemMessage - tsm: TradeSystemMessage - msm: MeetingSystemMessage - thsm: ThresholdSystemMessage - gsm: GeneralSystemMessage
+ SystemMessage() + msgForNothing(): String + invalidInput(): String + invalidNumber(): String + tryAgainMsgForWrongDateTime(): String + printResult(obj: ArrayList<Object>): String + printItemResult(obj: ArrayList<Item>): String + regUserAlerts(um: UserManager, tc: RegularUserThresholdController, username: String, menuPartOfAlert: String, thresholdValues: ArrayList<Integer>): String - activeAlerts(notification: StringBuilder, um: UserManager, tc: RegularUserThresholdController, username: String, thresholdValues: ArrayList<Integer>): void + lockMessageForThreshold(maxNumTransactionAWeek: int): String + msgForThresholdValue(currentVal: int): String + msgForGuest(): String + lockMessageForFrozen(): String + lockMessageForVacation(): String + msgForResult(validator: boolean): String + msgForStatusChangeResult(validator: boolean): String + msgForRequestProcess(validator: boolean): String + msgForRequestResult(validator: boolean): String + failMessageForFrozen(): String + lockMessageForTPLimit(): String + msgForFriendRequest(validator: boolean, userTold: int): String + msgForNothing(string: String): String + printListUser(listOfUser: ArrayList<TradableUser>): String + printFriendRequest(requests: HashMap<TradableUser, String>): String + printListObject(objects: ArrayList<Object>): String + printListNumberedObject(objects: ArrayList<Object>): String + tryAgainMsgForWrongInput(): String + tryAgainMsgForWrongFormatInput(): String + msgForTPcannotConfirm(): String + msgForNotYourTurn(): String + msgForFollowResult(validator: boolean): String + msgForNoTradeSuggestion(): String + msgForMeetingTookPlaceResult(): String + msgForNo(string: String): String + printAllMessages(messages: ArrayList<Messages>): String + msgForTradeCompletedOrNot(result: boolean): String + msgForSetTradable(validator: boolean, status: boolean): String + printHistoricalAction(listOfAction: ArrayList<Action>): String + printObject(object: Object): String + msgTradeRequestFail(): String + msgForRatingReview(rating: double, reviews: ArrayList<Review>): String + msgForReview(reviews: ArrayList<Review>): String

UML for Presenter

AccountSystemMessage
~ AccountSystemMessage() # msgForFriendRequest(validator: boolean, userToID: int): String # printFriendRequest(request: HashMap<TradadableUser, String>): String # msgForFollowResult(validator: boolean):String # printAllMessages(messages: ArrayList<Message>):String # msgForSetTradable(validator: boolean, status: boolean): String # msgForRatingReview(rating:double, reviews: ArrayList<Review>, pom: PrintObjectMessage): String # msgForReview(reviews: ArrayList<Review>, pom: PrintObjectMessage): String

ThresholdSystemMessage
~ ThresholdSystemMessage() # lockMessageForThreshold(maxNumTransactionsAWeek: int): String # msgForThresholdValue(currentVal: int): String # lockMessageForTPLimit(): String

MeetingSystemMessage
~ MeetingSystemMessage() # tryAgainMsgForWrongDatetime(): String # msgForTPcannotConfirm(): String # msgForNotYourTurn(): String # msgForMeetingTookPlaceResult(validator: boolean): String

TradeSystemMessage
~ TradeSystemMessage() # msgForNoTradeSuggestion(): String # msgForTradeCompletedOrNot(result: boolean): String # msgTradeRequestFail(): String

GeneralSystemMessage
~ GeneralSystemMessage() # msgForNothing(): String # invalidInput(): String # invalidNumber(): String # msgForGuest(): String # lockMessageForFrozen(): String # lockMessageForVacation(): String # msgForResult(validator: boolean): String # msgForStatusChangeResult(validator: boolean): String # msgForRequestSuccess(validator: boolean): String # msgForRequestResult(validator: boolean): String # failMessageForFrozen(): String # msgForNothing(string: String): String # tryAgainMsgForWrongInput(): String # tryAgainMsgForWrongFormatInput(): String # msgForNo(string: String): String

PrintObjectMessage
~ PrintObjectMessage() # printResult(obj: ArrayList<Object>): String # printItemResult(obj: ArrayList<Item>): String # printListUser(listOfUser: ArrayList<TradableUser>): String # printListObject(objects: ArrayList<Object>): String # printListNumberedObject(objects: ArrayList<Object>): String # printHistoricalAction(listOfAction: ArrayList<Action>): String # printObject(object: Object): String

UserAlertMessage
~ UserAlertMessage() # regUserAlerts(um: UserManager, tc: RegularUserThresholdController, username: String, menuPartOfAlert: String, thresholdValues: ArrayList<Integer>): String - activeAlerts(notification: StringBuilder, um: UserManager, tc: RegularUserThresholdController, username: String, thresholdValues: ArrayList<Integer>): void

AdminUserActionMessage
+ AdminUserActionMessage() + adminUserAction(action: Action): String # adminUserManageUsersAction(action: Action, subMenuOption: int): String # adminUserEditThresholdsAction(action: Action, subMenuOption: int): String # adminUserActionAction(action: Action, subMenuOption: int): String # adminUserOtherAction(action: Action, subMenuOption: int): String # helper_admin_action_prefix(action: Action): String

RegularUserActionMessage
+ RegularUserActionMessage() + regularUserAction(action: Action): String # regularUserAccountAction(action: Action, subMenuOption: int, subSubMenuOption: int): String # regularUserManagelItemsAction(action: Action, subSubMenuOption: int): String # regularUserAccountSettingsAction(action: Action, subSubMenuOption: int): String # regularUserFollowAction(action: Action, subSubMenuOption: int): String # regularUserTradeAction(action: Action, subMenuOption: int): String # regularUserMeetingAction(action: Action, subMenuOption: int): String # regularUserSearchingAction(action: Action, subMenuOption: int, subSubMenuOption: int): String # regularUserSearchingItemsAction(action: Action, subSubMenuOption: int): String # regularUserSearchingUsersAction(action: Action, subSubMenuOption: int): String # regularUserSearchingMeetingsAction(action: Action, subSubMenuOption: int): String # regularUserSearchingTradesAction(action: Action, subSubMenuOption: int): String # regularUserCommunityAction(action: Action, subMenuOption: int): String # helper_regular_action_prefix(action: Action): String