

mainview

Application

- + API_KEY_FILE_NAME:String
- + myApiKey:String
- + rootUrl:String
- + main(String[] args):void

NearExpiryContractFrame

+ contracts:List<Contract>

+ show():void

+ NearExpiryContractFrame(List<Contract> contracts)

NotificationFrame

- message:String
- + NotificationFrame(String message) + show():void

AuthenticationView

- + panel:Jpanel
- + userLabel:JLabel
- + userText:JTextField
- + passwordLabel:JLabel
- + passwordText:JPasswordField
- + loginButton:JButton
- · PREFERRED_SIZE:Dimension
- MAX INPUT LENGTH:int
- + AuthenticationView(Display display)
- placeComponents():void
- + getUserName():String
- + getPassword():String

<<interface>> NavigationPane

+ showNavigationPane(Display display, JPanel parent, JButton[] buttons):void

<<interface>> Observer

+ update(EventType e):void

ContractDurationFrame

- three:JRadioButtonMenuItem
- six:JRadioButtonMenuItem
- twelve:JRadioButtonMenuItem
- twtvfour:JRadioButtonMenuItem
- longer:JRadioButtonMenuItem
- longerduration:JTextField
- duration:ButtonGroup
- selectedDuration:int
- + ContractDurationFrame()
- + show():void
- reload():void
- + getselectedduration():int
- isNumeric(String s):Boolean
- + getDuration():int

<<interface>> MouseClickListener

- + mousePressed(MouseEvent e):void
- + mouseReleased(MouseEvent e):void
- + mouseEntered(MouseEvent e):void
- + mouseExited(MouseEvent e) :void

RemovablePanel

- + RemovablePanel()
- + RemovablePanel(LayoutManager layout)
- + onRemoved():void
- + onAttached():void

Controller

<<interface>> MessageView

+ getLogArea(List<String> messageLog)
+ getChatBox()

Utils

- mapper:ObjectMapper
- + format:SimpleDateFormat
- + SUCCESS_CONTRACT_CREATION:NotificationFrame
- + SUCCESS_BID_CREATION:NotificationFrame
- + INSUFFICIENT_COMPETENCY:NotificationFrame
- $+ \ SUCCESS_MATCH_REQUEST: Notification Frame$
- + PLEASE_FILL_IN:NotificationFrame + INVALID_FIELDS:NotificationFrame
- + REACHED_CONTRACT_LIMIT:NotificationFrame
- + CONTRACT_SIGNED:NotificationFrame
- + OTHER_PARTY_PENDING:NotificationFrame
- + INVALID_USER:NotificationFrame
- + CANNOT_CREATE_BID:NotificationFrame
- + CANNOT_MODIFY_BID:NotificationFrame
- + formatDate(String datetime):Date

Display

- frame:JFrame
- + FRAME_WIDTH:int
- + FRAME_HEIGHT:int
- + Display()
- + createPanel(JComponent panel):void
- + setVisible():void
- + removePanel(JComponent panel):void
- + closeWindow():void

HomeView

- user:User
- + panel:Jpanel
- + studentButton:JButton
- + tutorButton:JButton
- + logout:JButton
- + HomeView(Display display, User user)
- placeComponents():void

<<abstract>> View

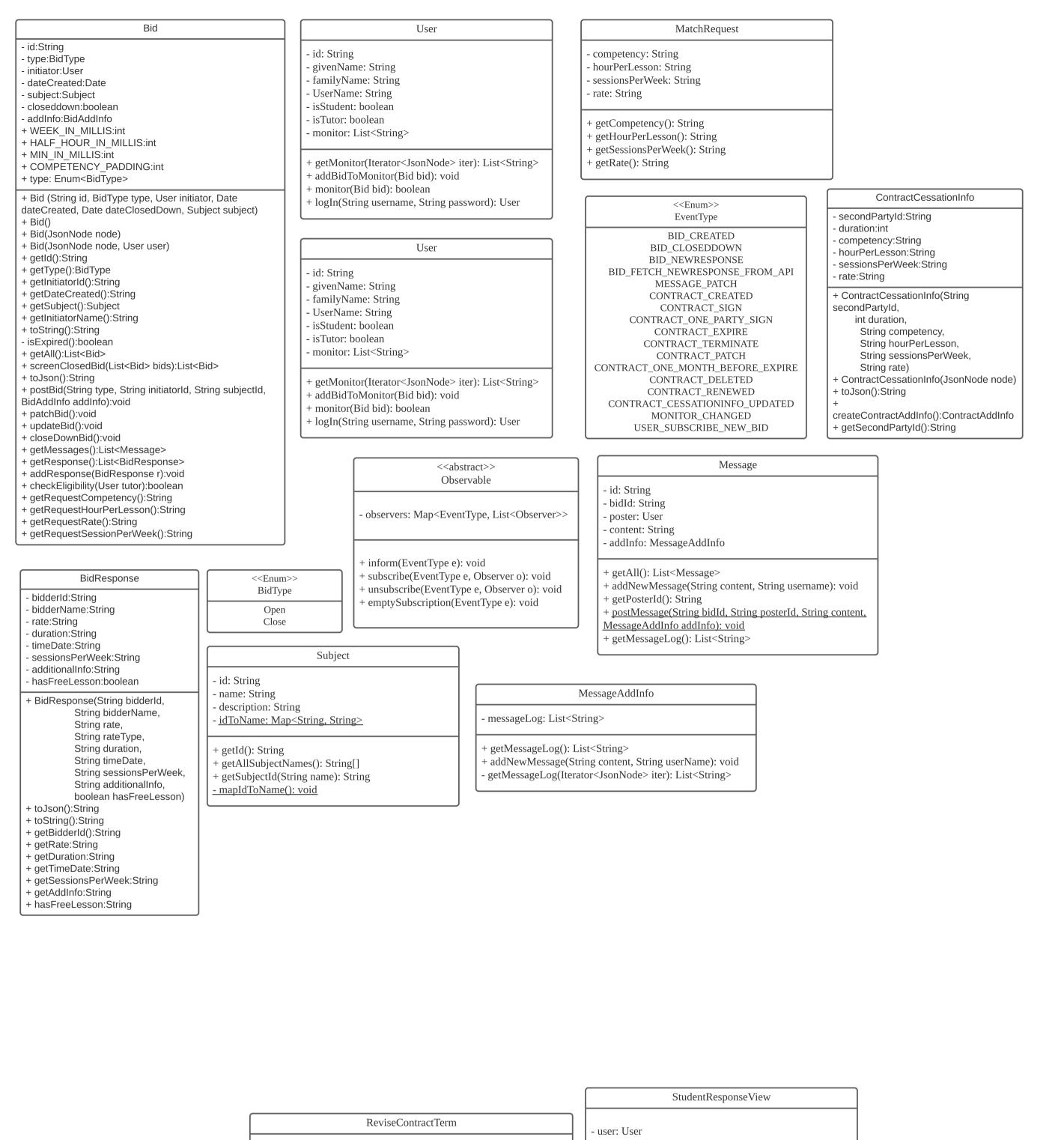
- display:Display
- homeButton:JButton
- + activePanel:RemovablePanel
- + View(Display display)
- + display():void
- createPanel(LayoutManager layout:JPanel
- createPanel():JPanel
- placeComponents():void
- setSwitchPanelListener(RemovablePanel panel, Component comp, View newView):void
- + mouseClicked(MouseEvent e):void
- setSwitchPanelListener(JPanel main, Component comp,
- RemovablePanel newPanel):void
- + mouseClicked(MouseEvent e):void
- addMouseListener(JComponent component,

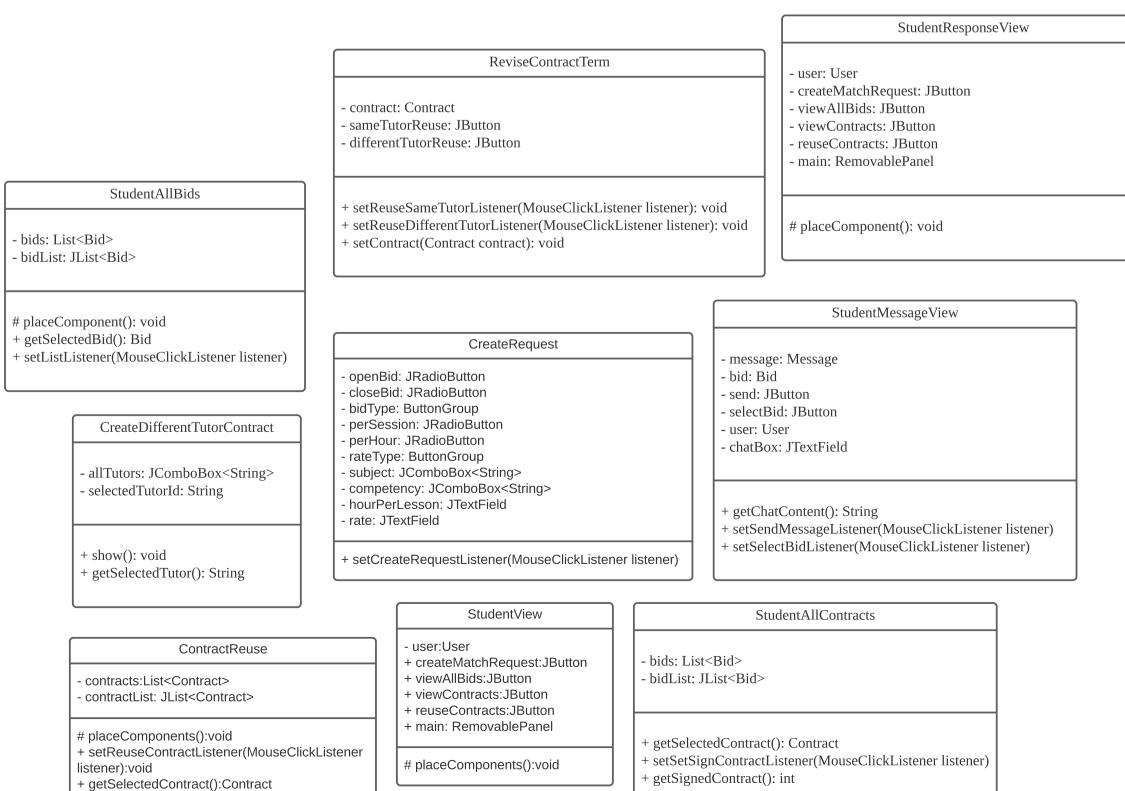
MouseClickListener listener):void

Bid <<Enum>> BidResponse BidType - id:String - bidderId:String type:BidType Open - bidderName:String initiator:User Close rate:String - dateCreated:Date duration:String Monitor BidAddInfo - subject:Subject Contract timeDate:String closeddown:boolean sessionsPerWeek:String - addInfo:BidAddInfo ContractAddInfo - additionalInfo:String + WEEK IN MILLIS:int hasFreeLesson:boolean + HALF HOUR IN MILLIS:int + BidResponse(String bidderId, + MIN IN MILLIS:int String bidderName, + COMPETENCY PADDING:int String rate, + type: Enum<BidType> String rateType, + Bid (String id, BidType type, User initiator, Date String duration, dateCreated, Date dateClosedDown, Subject subject) String timeDate, Message Subject String sessionsPerWeek, + Bid(JsonNode node) EventType String additionalInfo, <<Enum>> + Bid(JsonNode node, User user) boolean hasFreeLesson) EventType + getId():String + toJson():String BID CREATED + getType():BidType + toString():String BID CLOSEDDOWN + getInitiatorId():String + getBidderId():String BID NEWRESPONSE + getDateCreated():String + getRate:String <<interface>> BID_FETCH_NEWRESPONSE_FROM_API + getSubject():Subject Model + getDuration:String MESSAGE PATCH + getInitiatorName():String + getTimeDate:String ContractCessationInfo + get(String url):HttpResponse<String> CONTRACT_CREATED + toString():String + getSessionsPerWeek:String + getAll(String - secondPartvId:String CONTRACT SIGN - isExpired():boolean + getAddInfo:String extension):List<ObjectNode> CONTRACT_ONE_PARTY_SIGN duration:int + getAll():List<Bid> + hasFreeLesson:String + get(String extension, String CONTRACT_EXPIRE - competency:String + screenClosedBid(List<Bid> bids):List<Bid> <<abstract>> id):ObjectNode CONTRACT_TERMINATE User - hourPerLesson:String + toJson():String Observable + post(String url, String json):void - sessionsPerWeek:String CONTRACT PATCH + postBid(String type, String initiatorId, String subjectId, + delete(String url, String ison):void rate:String CONTRACT_ONE_MONTH_BEFORE_EXPIRE BidAddInfo addInfo):void + observers: Map<EventType, + patch(String url, String json):void CONTRACT DELETED + patchBid():void List<Observer>> + ContractCessationInfo(String CONTRACT_RENEWED + updateBid():void secondPartyId. CONTRACT_CESSATIONINFO_UPDATED + closeDownBid():void int duration. + inform(EventType e):void MONITOR CHANGED + getMessages():List<Message> String competency, + subscribe(EventType e, Observer o):void USER SUBSCRIBE NEW BID + getResponse():List<BidResponse> String hourPerLesson, + unsubscribe(EventType e, Observer + addResponse(BidResponse r):void MessageAddInfo String sessionsPerWeek, + checkEligibility(User tutor):boolean String rate) MatchRequest + emptySubscription(EventType e):void + getRequestCompetency():String + ContractCessationInfo(JsonNode node) + getRequestHourPerLesson():String + toJson():String + getRequestRate():String + getRequestSessionPerWeek():String createContractAddInfo():ContractAddInfo + getSecondPartyId():String

ContractReuse	StudentAllBids	ReviseContractTerm	CreateSameTutorContract	
- contracts:List <contract> - contractList: JList<contract></contract></contract>				
+ ContractReuse(List <contract> contracts) + placeComponents():void</contract>				
+ update(EventType e):void				
+ setReuseContractListener(MouseClickListener istener):void				
+ getSelectedContract():Contract				
				Charles th III Control to
				StudentAllContracts
StudentResponseView	StudentMessageView	StudentView	CreateRequest	
		- user:User		
		+ createMatchRequest:JButton + viewAllBids:JButton		
		+ viewContracts:JButton		
		+ reuseContracts:JButton + main: RemovablePanel		
		Hail. Nemovabler allel		
		+ StudentView(Display display,		
		User user) - placeComponents():void		
		pisso somponomon		

TutorMonitorView	TutorResponseView	CreateBid	
			TutorAllContracts
TutorView			
TutorView		TutorMessageView	
- user:User	TutorAllBids	TutorMessageView	
- user:User + createBid:JButton + viewAllBids:JButton	TutorAllBids	TutorMessageView	
- user:User + createBid:JButton + viewAllBids:JButton	TutorAllBids	TutorMessageView	
- user:User + createBid:JButton	TutorAllBids	TutorMessageView	
- user:User + createBid:JButton + viewAllBids:JButton + viewContracts:JButton + viewMonitor:JButton + main: RemovablePanel	TutorAllBids	TutorMessageView	
- user:User + createBid:JButton + viewAllBids:JButton + viewContracts:JButton + viewMonitor:JButton + main: RemovablePanel + TutorView(Display display, User	TutorAllBids	TutorMessageView	
- user:User + createBid:JButton + viewAllBids:JButton + viewContracts:JButton + viewMonitor:JButton + main: RemovablePanel	TutorAllBids	TutorMessageView	





Application

AuthenticationView

+ panel:Jpanel + userLabel:JLabel + userText:JTextField + passwordLabel:JLabel + passwordText:JPasswordField + loginButton:JButton PREFERRED SIZE:Dimension - MAX INPUT LENGTH:int

+ AuthenticationView(Display display) placeComponents():void + getUserName():String + getPassword():String

ContractDurationFrame

three: JRadioButtonMenuItem six: JRadioButtonMenuItem twelve: JRadioButtonMenuItem longer: JRadioButtonMenuItem longerDuration: JTextField duration: ButtonGroup selectedDuration: int

+ show(): void + getDuration(): int reload(): void getSelectedDuration(): void isNumeric(String s): boolean

- frame: JFrame FRAME WIDTH: int - FRAME_HEIGHT: int

display:Display

+ display():void

homeButton:JButton

+ View(Display display)

- activePanel:RemovablePanel

closeWindow(): void

<<abstract>>

+ API KEY FILE NAME:String + myApiKey:String rootUrl:String - main(String[] args):void

twentyFour: JRadioButtonMenuItem

Display

- createPanel(JComponent panel): void removePanel(JComponent panel): void

Main Controller display: Display user: User initiatedBids: List<Bid> - allBids: List<Bid>

> activeBid: Bid subscriberBid: Bid activeContract: Contract subscriberContract: Contract activeMessage: Message homeView: HomeView studentAllBids: StudentAllBids studentAllContracts: StudentAllContracts createRequest: CreateRequest

monitoredBids: List<Bid>

allContracts: List<Contract>

studentExpiredContracts: List<Contract>

studentView: StudentView studentMessage: StudentMessage contractReuse: ContractReuse reviseContractTerm: ReviseContractTerm

> tutorAllBids: TutorAllBids tutorAllContracts: TutorAllContracts createBid: CreateBid tutorView: TutorView

tutorResponse: TutorResponseView

tutorMonitor: TutorMonitorView tutorMessage: TutorMessageView createSameTutorContract: CreateSameTutorContract - createDifferentTutorContract: CreateDifferentTutorContract

+ start(): void · initStudent(): void · initTutor(): void fetchInitiatedBids(): void reFetchAllBids(): void fetchAllMonitoredBids(): void fetchAllContractAsFirstPartv(): void fetchAllContractAsSecondParty(): void reFetchAllContractAsFirstParty(): void - reFetchAllContractAsFirstParty:() void

fetchStudentExpiredContract:(): void

- subscribeBidNewResponse(): void · subscribeMessage(): void subscribeContractCreation(): void · showStudentMessagePanel(): void · showTutorMessagePanel(): void

- subscribeBidCreation(): void

<<interface>> Observer

NotificationFrame

- createPanel(LayoutManager layout:JPanel createPanel():JPanel + show(): void placeComponents():void setSwitchPanelListener(RemovablePanel panel, Component comp, View newView):void

RemovablePanel + mouseClicked(MouseEvent e) voice setSwitchPanelListener(JPanel main, Component comp, RemovablePanel newPanel):void + mouseClicked(MouseEvent e):void - addMouseListener(JComponent component, MouseClickListener listener):void

update(EventType e)

· message: String

+ RemovablePanel(LayoutManager + onRemoved(): void

+ onAttached(): void

getChatBox(): JTextField <<interface>>

<<interface>>

MessageView

getLogArea(List<String> messageLog): JTextArea

MouseClickListener

HomeView

+ HomeView(Display display, User user)

user:User

+ panel:Jpanel

+ logout:JButton

- studentButton:JButton

placeComponents():void

+ tutorButton:JButton

- mousePressed(MouseEvent e):void + mouseReleased(MouseEvent e):void - mouseEntered(MouseEvent e):void - mouseExited(MouseEvent e) :void

> <<interface>> NavigationPane

showNavigationPane(Display display, JPanel parent, JButton[] buttons)

NearExpiryContractFrame

List<Contract> contracts

show(): void

Utils mapper:ObjectMapper format:SimpleDateFormat

- INSUFFICIENT COMPETENCY: NotificationFrame - SUCCESS MATCH REQUEST: Notification Frame - PLEASE FILL IN:NotificationFrame - INVALID FIELDS:NotificationFrame REACHED CONTRACT LIMIT: NotificationFrame - CONTRACT SIGNED: Notification Frame + OTHER PARTY PENDING:NotificationFrame + INVALID USER:NotificationFrame + CANNOT_CREATE_BID:NotificationFrame

- SUCCESS CONTRACT CREATION: NotificationFrame

- SUCCESS BID CREATION:NotificationFrame

+ formatDate(String datetime):Date

+ CANNOT_MODIFY_BID:NotificationFrame

TutorMonitor timer: Timer - bids: List<Bid> responseList: JList<BidResponse responses: List<BidResponse>

placeComponent(): void onAttached():void

- user:User + createMatchRequest:JButton + viewAllBids:JButton + viewContracts:JButton + reuseContracts:JButton + main: RemovablePanel

TutorView

placeComponent(): void

bids: List<Bid> bidList: JList<Bid>

+ getSelectedContract(): Contract + setSetSignContractListener(MouseClickListener listener)

TutorMessageView

TutorAllContracts

- message: Message - bid: Bid - send: JButton selectBid: JButton user: User chatBox: JTextField

getChatContent(): String + setSendMessageListener(MouseClickListener listener) setSelectBidListener(MouseClickListener listener)

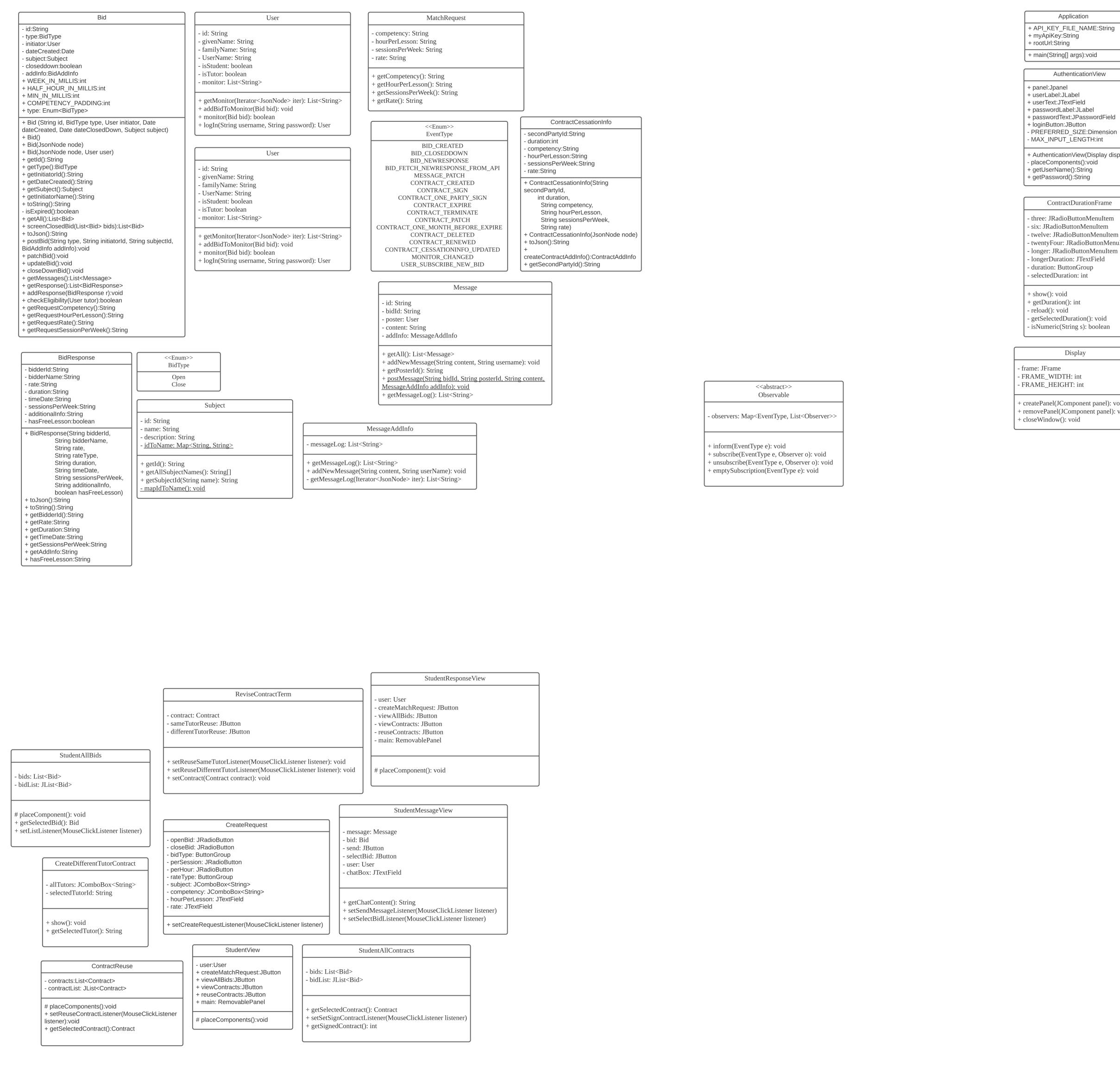
TutorAllBids · bids: List<Bid> · bidList: JList<Bid> # placeComponent(): void + getSelectedBid(): Bid - setListListener(MouseClickListener listener)

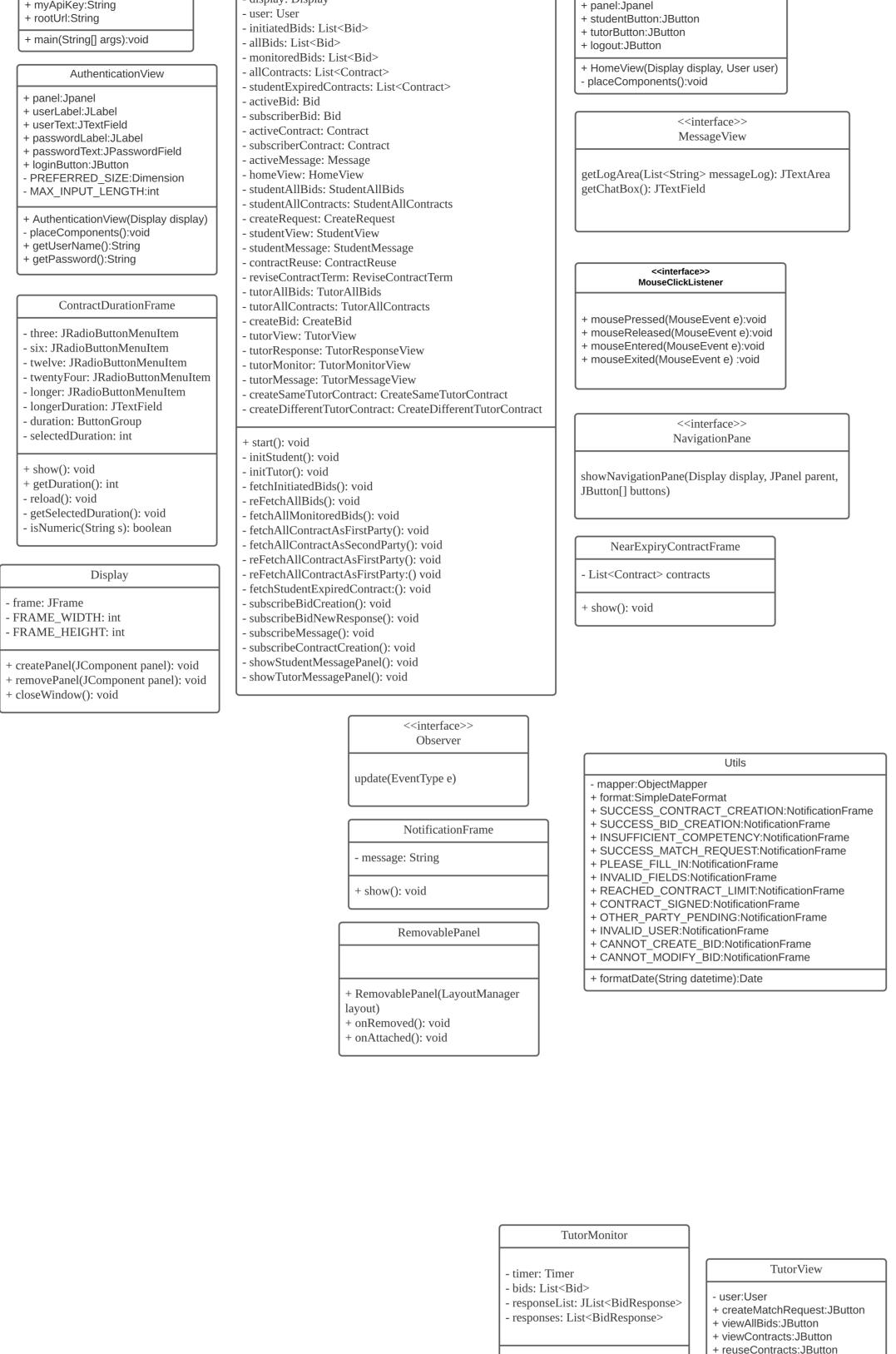
TutorResponseView

user: User createMatchRequest: JButton viewAllBids: JButton viewContracts: JButton reuseContracts: JButton main: RemovablePanel

placeComponent(): void - setCreateBidListener(MouseClickListener listener): void - setBuyOutListener(MouseClickListener listener): void setSubscribedBidListener(MouseClickListener listener):

- setBid(Bid bid): void





TutorAllBids

+ setListListener(MouseClickListener listener)

- createMatchRequest: JButton

viewAllBids: JButton

viewContracts: JButton

- reuseContracts: JButton

- main: RemovablePanel

placeComponent(): void

+ setBid(Bid bid): void

TutorResponseView

+ setCreateBidListener(MouseClickListener listener): void

+ setBuyOutListener(MouseClickListener listener): void

+ setSubscribedBidListener(MouseClickListener listener):

- bids: List<Bid>

- bidList: JList<Bid>

placeComponent(): void

- user: User

+ getSelectedBid(): Bid

placeComponent(): void

- bids: List<Bid>

- bidList: JList<Bid>

- message: Message

- selectBid: JButton

- chatBox: JTextField

+ getChatContent(): String

- bid: Bid

- send: JButton

- user: User

+ getSelectedContract(): Contract

+ onAttached():void

- display: Display

HomeView

+ main: RemovablePanel

placeComponent(): void

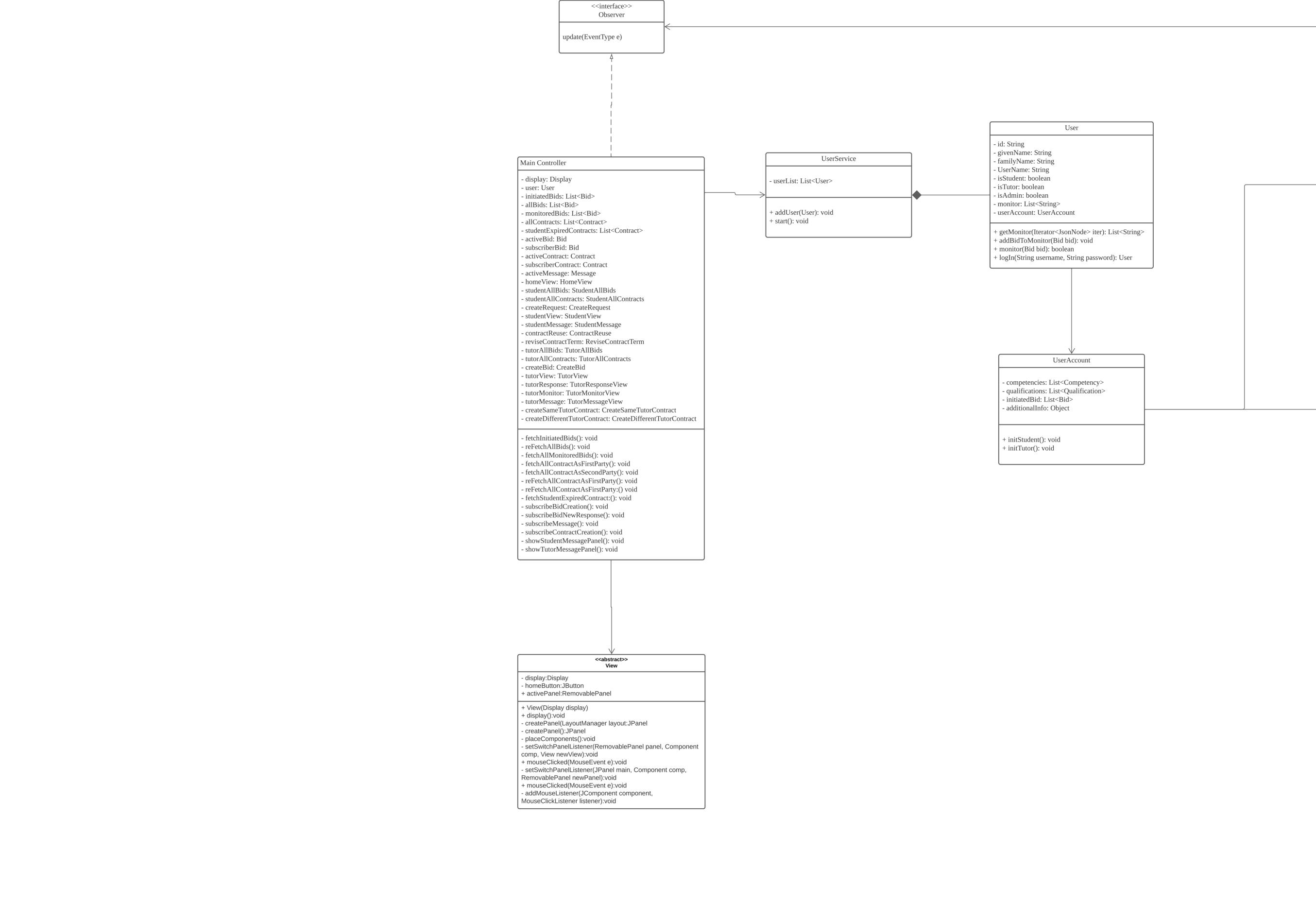
TutorAllContracts

+ setSetSignContractListener(MouseClickListener listener)

TutorMessageView

+ setSendMessageListener(MouseClickListener listener)

+ setSelectBidListener(MouseClickListener listener)



Model

<<abstract>>

Observable

- observers: Map<EventType, List<Observer

+ subscribe(EventType e, Observer o): void

+ unsubscribe(EventType e, Observer o): void

BidService

CompetencyService

competencyList: List<Competency>

+ addCompetency(): void

+ emptySubscription(EventType e): void

- initiator:User

- dateCreated:Date

subject:Subject

closeddown:boolean

addInfo:BidAddInfo

+ WEEK IN MILLIS:int

+ MIN IN MILLIS:int

+ type: Enum<BidType>

+ Bid(JsonNode node)

+ getId():String

+ getType():BidType

→ + getInitiatorId():String

+ toString():String

+ getDateCreated():String

+ getSubject():Subject

isExpired():boolean

BidAddInfo addInfo):void

+ patchBid():void

+ updateBid():void

+ closeDownBid():void

+ getMessages():List<Message>

+ getResponse():List<BidResponse

+ addResponse(BidResponse r):void

+ checkEligibility(User tutor):boolean

+ getRequestCompetency():String

+ getRequestHourPerLesson():Strin

+ getRequestSessionPerWeek():String

+ getRequestRate():String

+ getAll():List<Bid>

+ getInitiatorName():String

+ HALF HOUR IN MILLIS:int

+ COMPETENCY PADDING:int

+ Bid(JsonNode node, User user)

+ Bid (String id, BidType type, User initiator, Date

+ screenClosedBid(List<Bid> bids):List<Bid>

+ postBid(String type, String initiatorId, String subjectId,

Competency

dateCreated, Date dateClosedDown, Subject subject)

+ inform(EventType e): void

bidList: List<Bid>

+ addBid(): void

+ removeBid(): void