

Application

+ API_KEY_FILE_NAME:String
+ myApiKey:String
+ rootUrl:String

+ main(String[] args):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

ContractDurationFrame

- three:JRadioButtonMenuItem
- six:JRadioButtonMenuItem
- twelve:JRadioButtonMenuItem
- twtyfour:JRadioButtonMenuItem
- longer:JRadioButtonMenuItem
- longerduration:JTextField
- duration:ButtonGroup
- selectedDuration:int

+ ContractDurationFrame()
+ show():void
- reload():void
+ getselecteddduration():int
- isNumeric(String s):Boolean
+ getDuration():int

Controller

Display

- frame:JFrame
+ FRAME_WIDTH:int
+ FRAME_HEIGHT:int

+ Display()
+ createPanel(JComponent panel):void
+ setVisible():void
+ removePanel(JComponent panel):void
+ closeWindow():void

NearExpiryContractFrame

+ contracts:List<Contract>

+ NearExpiryContractFrame(List<Contract> contracts)
+ show():void

<<interface>>
MessageView

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

HomeView

- user:User
+ panel:Jpanel
+ studentButton:JButton
+ tutorButton:JButton
+ logout:JButton

+ HomeView(Display display, User user)
- placeComponents():void

NotificationFrame

- message:String

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

<<interface>>
NavigationPane

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

<<interface>>
MouseClickedListener

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

Utils

- mapper:ObjectMapper
+ format:SimpleDateFormat
+ SUCCESS_CONTRACT_CREATION:NotificationFrame
+ SUCCESS_BID_CREATION:NotificationFrame
+ INSUFFICIENT_COMPETENCY:NotificationFrame
+ SUCCESS_MATCH_REQUEST:NotificationFrame
+ 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

<<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

RemovablePanel

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

<<interface>>
Observer

+ update(EventType e):void

ContractReuse
- contracts:List<Contract> - contractList: JList<Contract>
+ ContractReuse(List<Contract> contracts) + placeComponents():void + update(EventType e):void + setReuseContractListener(MouseClickListener listener):void + getSelectedContract():Contract

StudentAllBids

ReviseContractTerm

CreateSameTutorContract

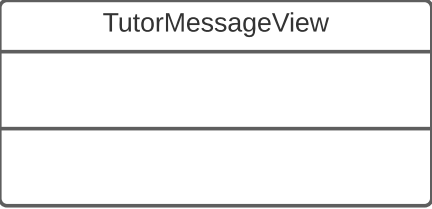
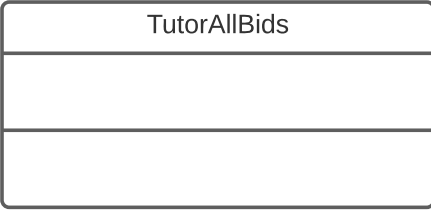
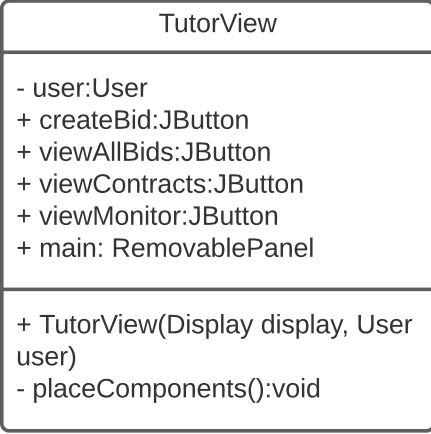
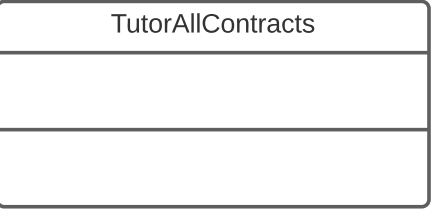
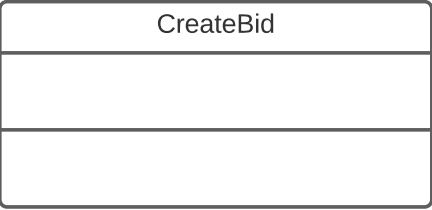
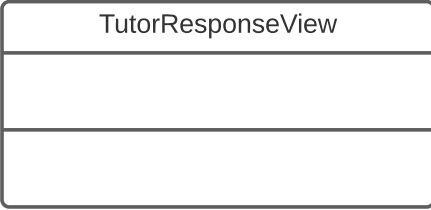
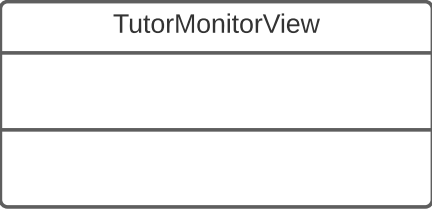
StudentAllContracts

StudentResponseView

StudentMessageView

StudentView
- user:User + createMatchRequest:JButton + viewAllBids:JButton + viewContracts:JButton + reuseContracts:JButton + main: RemovablePanel
+ StudentView(Display display, User user) - placeComponents():void

CreateRequest



Bid
- id:String - type:BidType - initiator:User - dateCreated:Date - subject:Subject - closeddown:boolean + addInfo:BidAddInfo + WEEK_IN_MILLIS:int + HALF_HOUR_IN_MILLIS:int + MIN_IN_MILLIS:int + COMPETENCY_PADDING:int + type: Enum<BidType>
+ Bid (String id, BidType type, User initiator, Date dateCreated, Date dateClosedDown, Subject subject) + Bid() + Bid(JsonNode node) + Bid(JsonNode node, User user) + getId():String + getType():BidType + getInitiatorId():String + getDateCreated():String + getSubject():Subject + getInitiatorName():String + toString():String + isExpired():boolean + getAll():List<Bid> + screenClosedBid(List<Bid> bids):List<Bid> + toJson():String + postBid(String type, String initiatorId, String subjectId, 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():String + getRequestRate():String + getRequestSessionPerWeek():String

User
- id: String - givenName: String - familyName: String - userName: String - isStudent: boolean - isTutor: boolean - monitor: List<String>
+ getMonitor(Iterator<JsonNode> iter): List<String> + addBidToMonitor(Bid bid): void + monitor(Bid bid): boolean + login(String username, String password): User

User
- id: String - givenName: String - familyName: String - userName: String - isStudent: boolean - isTutor: boolean - monitor: List<String>
+ getMonitor(Iterator<JsonNode> iter): List<String> + addBidToMonitor(Bid bid): void + monitor(Bid bid): boolean + login(String username, String password): User

MatchRequest
- competency: String - hourPerLesson: String - sessionsPerWeek: String - rate: String
+ getCompetency(): String + getHourPerLesson(): String + getSessionsPerWeek(): String + getRate(): String

<<Enum>> EventType
BID_CREATED BID_CLOSEDDOWN BID_NEWRESPONSE BID_FETCH_NEWRESPONSE_FROM_API MESSAGE_PATCH CONTRACT_CREATED CONTRACT_SIGN CONTRACT_ONE_PARTY_SIGN CONTRACT_EXPIRE CONTRACT_TERMINATE CONTRACT_PATCH CONTRACT_ONE_MONTH_BEFORE_EXPIRE CONTRACT_DELETED CONTRACT_RENEWED CONTRACT_CESSATIONINFO_UPDATED MONITOR_CHANGED USER_SUBSCRIBE_NEW_BID

Message
- id: String - bidId: String - poster: User - content: String - addInfo: MessageAddInfo
+ getAll(): List<Message> + addNewMessage(String content, String username): void + getPosterId(): String + postMessage(String bidId, String posterId, String content, MessageAddInfo addInfo): void + getMessageLog(): List<String>

<<abstract>> Observable
- observers: Map<EventType, List<Observer>>
+ inform(EventType e): void + subscribe(EventType e, Observer o): void + unsubscribe(EventType e, Observer o): void + emptySubscription(EventType e): void

MessageAddInfo
- messageLog: List<String>
+ getMessageLo + addNewMessage(String content, String userName): void + getMessageLog(Iterator<JsonNode> iter): List<String>

<<Enum>> Bid Type
Open Close

BidResponse
- bidderId:String - bidderName:String - rate:String - duration:String - timeDate:String - sessionsPerWeek:String - additionalInfo:String - hasFreeLesson:boolean
+ BidResponse(String bidderId, String bidderName, String rate, String rateType, String duration, String timeDate, String sessionsPerWeek, String additionalInfo, boolean hasFreeLesson) + toJson():String + toString():String + getBidderId():String + getRate:String + getDuration:String + getTimeDate:String + getSessionsPerWeek:String + getAddInfo:String + hasFreeLesson:String

Subject
- id: String - name: String - description: String - idToName: Map<String, String>
+ getId():String + getAllSubjectNames():String[] + getSubjectId(String name):String - mapIdToName():void

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

ContractDurationFrame
- three: JRadioButtonMenuItem - six: JRadioButtonMenuItem - twelve: JRadioButtonMenuItem - twentyFour: JRadioButtonMenuItem - longer: JRadioButtonMenuItem - longerDuration: JTextField - duration: ButtonGroup - selectedDuration: int
+ show(): void + getDuration(): int + reload(): void + getSelectedDuration(): void + isNumeric(String s): boolean

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

Main Controller
- display: Display - user: User - initiatedBids: List<Bid> - allBids: List<Bid> - monitoredBids: List<Bid> - allContracts: List<Contract> - studentExpiredContracts: List<Contract> - activeBid: Bid - subscriberBid: Bid - activeContract: Contract - subscriberContract: Contract - activeMessage: Message - homeView: HomeView - studentAllBids: StudentAllBids - studentAllContracts: StudentAllContracts - createRequest: CreateRequest - 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 - fetchAllContractAsFirstParty(): void - fetchAllContractAsSecondParty(): void - reFetchAllContractAsFirstParty(): void - reFetchAllContractAsFirstParty(): void - fetchStudentExpiredContract(): void - subscribeBidCreation(): void - subscribeBidNewResponse(): void - subscribeMessage(): void - subscribeContractCreation(): void - showStudentMessagePanel(): void - showTutorMessagePanel(): void

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

<<interface>> MessageView
getLogArea(List<String> messageLog): JTextArea getChatBox(): JTextField

<<interface>> MouseClickedListener
+ 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

<<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

<<interface>> Observer
update(EventType e)

NotificationFrame
- message: String
+ show(): void

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

Utils
- mapper:ObjectMapper + format:SimpleDateFormat + SUCCESS_CONTRACT_CREATION:NotificationFrame + SUCCESS_BID_CREATION:NotificationFrame + INSUFFICIENT_COMPETENCY:NotificationFrame + SUCCESS_MATCH_REQUEST:NotificationFrame + 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

<<abstract>> View
- display: Display
placeComponents(): void # setSwitchPanelListener(RemovablePanel panel, Component component, View view): void # addMouseListener(JPanel main, Component component, RemovablePanel panel)

StudentAllBids
- bids: List<Bid> - bidList: JList<Bid>
+ getSelectedBid(): Bid + setListListener(MouseClickListener listener)

TutorMonitor
- timer: Timer - bids: List<Bid> - responseList: JList<BidResponse> - responses: List<BidResponse>
onAttached():void

ContractReuse
- contracts: List<Contract> - contractList: JList<Contract>
+ setReuseContractListener(MouseClickListener listener): void + getSelectedContract: Contract

TutorView
placeComponent # setSwitchPanel(): void

ContractCessationInfo
- secondPartyId:String - duration:int - competency:String - hourPerLesson:String - sessionsPerWeek:String - rate:String
+ ContractCessationInfo(String secondPartyId, int duration, String competency, String hourPerLesson, String sessionsPerWeek, String rate) + ContractCessationInfo(JsonNode node) + toJson():String + createContractAddInfo():ContractAddInfo + getSecondPartyId():String

StudentView
- user:User + createMatchRequest:JButton + viewAllBids:JButton + viewContracts:JButton + reuseContracts:JButton + main: RemovablePanel
+ StudentView(Display display, User user) - placeComponents():void

ContractReuse
- contracts:List<Contract> - contractList: JList<Contract>
+ ContractReuse(List<Contract> contracts) + placeComponents():void + update(EventType e):void + setReuseContractListener(MouseClickListener listener):void + getSelectedContract():Contract

