

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	

Bid User MatchRequest - id:String - id: String competency: String type:BidType - givenName: String - hourPerLesson: String initiator:User - familyName: String - sessionsPerWeek: String dateCreated:Date - rate: String - UserName: String subject:Subject - isStudent: boolean closeddown:boolean addInfo:BidAddInfo - isTutor: boolean getCompetency(): String + WEEK_IN_MILLIS:int - monitor: List<String> getHourPerLesson(): String + HALF_HOUR_IN_MILLIS:int getSessionsPerWeek(): String + MIN_IN_MILLIS:int getMonitor(Iterator<JsonNode> iter): List<String> getRate(): String + COMPETENCY_PADDING:int + addBidToMonitor(Bid bid): void type: Enum<BidType> monitor(Bid bid): boolean + Bid (String id, BidType type, User initiator, Date + logIn(String username, String password): User <<Enum>> dateCreated, Date dateClosedDown, Subject subject) EventType + Bid(JsonNode node) BID_CREATED + Bid(JsonNode node, User user) User BID_CLOSEDDOWN getId():String BID_NEWRESPONSE getType():BidType BID_FETCH_NEWRESPONSE_FROM_API - id: String getInitiatorId():String MESSAGE_PATCH - givenName: String getDateCreated():String CONTRACT_CREATED - familyName: String getSubject():Subject CONTRACT_SIGN - UserName: String getInitiatorName():String CONTRACT_ONE_PARTY_SIGN - isStudent: boolean toString():String CONTRACT_EXPIRE - isTutor: boolean - isExpired():boolean CONTRACT_TERMINATE - monitor: List<String> getAll():List<Bid> CONTRACT_PATCH + screenClosedBid(List<Bid> bids):List<Bid> CONTRACT_ONE_MONTH_BEFORE_EXPIRE + toJson():String CONTRACT_DELETED getMonitor(Iterator<JsonNode> iter): List<String> + postBid(String type, String initiatorId, String subjectId, CONTRACT_RENEWED addBidToMonitor(Bid bid): void BidAddInfo addInfo):void CONTRACT_CESSATIONINFO_UPDATED monitor(Bid bid): boolean + patchBid():void MONITOR_CHANGED + logIn(String username, String password): User updateBid():void USER SUBSCRIBE NEW BID closeDownBid():void getMessages():List<Message> getResponse():List<BidResponse> Message <<abstract>> - addResponse(BidResponse r):void Observable checkEligibility(User tutor):boolean id: String getRequestCompetency():String bidId: String getRequestHourPerLesson():String observers: Map<EventType, List<Observer>> poster: User getRequestRate():String content: String + getRequestSessionPerWeek():String addInfo: MessageAddInfo inform(EventType e): void getAll(): List<Message> subscribe(EventType e, Observer o): void BidResponse <<Enum>> addNewMessage(String content, String username): void + unsubscribe(EventType e, Observer o): void BidType - bidderId:String getPosterId(): String emptySubscription(EventType e): void - bidderName:String Open - postMessage(String bidId, String posterId, String content, - rate:String Close MessageAddInfo addInfo): void - duration:String - getMessageLog(): List<String> - timeDate:String Subject sessionsPerWeek:String additionalInfo:String - id: String hasFreeLesson:boolean MessageAddInfo name: String + BidResponse(String bidderId description: String String bidderName, idToName: Map<String, String> · messageLog: List<String> String rate, String rateType, String duration, + getMessageLo + getId(): String String timeDate, + addNewMessage(String content, String userName): void + getAllSubjectNames(): String[] String sessionsPerWeek, + getSubjectId(String name): String getMessageLog(Iterator<JsonNode> iter): List<String> String additionalInfo, - mapIdToName(): void boolean hasFreeLesson) + toJson():String + toString():String + getBidderId():String + getRate:String getDuration:String

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

+ getTimeDate:String

+ getAddInfo:String

+ hasFreeLesson:String

+ getSessionsPerWeek:String

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

 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

secondPartyId:String

ContractCessationInfo

StudentView + 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

Application + API KEY FILE NAME:String + myApiKey:String + rootUrl:String

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

+ main(String[] args):void

AuthenticationView

+ 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

- display:Display

+ display():void

homeButton:JButton

+ View(Display display)

createPanel():JPanel

- placeComponents():void

comp, View newView):void

+ mouseClicked(MouseEvent e):void

RemovablePanel newPanel):void + mouseClicked(MouseEvent e):void

MouseClickListener listener):void

+ activePanel:RemovablePanel

createPanel(LayoutManager layout:JPanel

- addMouseListener(JComponent component,

<<abstract>>

- setSwitchPanelListener(RemovablePanel panel, Component

setSwitchPanelListener(JPanel main, Component comp,

allContracts: List<Contract> - studentExpiredContracts: List<Contract> activeBid: Bid subscriberBid: Bid - activeContract: Contract - subscriberContract: Contract activeMessage: Message homeView: HomeView - studentAllBids: StudentAllBids

Main Controller

- user: User

- display: Display

- allBids: List<Bid>

initiatedBids: List<Bid>

monitoredBids: List<Bid>

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

List<Contract> contracts + show(): void

JButton[] buttons)

<<interface>>

update(EventType e) NotificationFrame message: String show(): void RemovablePanel

Observer

- RemovablePanel(LayoutManager

+ 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 FINVALID FIELDS:NotificationFrame REACHED CONTRACT LIMIT:NotificationFrame CONTRACT SIGNED:NotificationFrame

HomeView

+ HomeView(Display display, User user)

<<interface>>

MouseClickListener

+ mousePressed(MouseEvent e):void

+ mouseReleased(MouseEvent e):void

+ mouseEntered(MouseEvent e):void

NearExpiryContractFrame

<<interface>>

NavigationPane

showNavigationPane(Display display, JPanel parent,

+ mouseExited(MouseEvent e) :void

<<interface>>

MessageView

getLogArea(List<String> messageLog): JTextArea

user:User

+ panel:Jpanel

+ logout:JButton

+ studentButton:JButton

placeComponents():void

getChatBox(): JTextField

+ tutorButton:JButton

+ OTHER PARTY PENDING:NotificationFrame + INVALID USER:NotificationFrame + CANNOT_CREATE_BID:NotificationFrame + CANNOT MODIFY BID:NotificationFrame + formatDate(String datetime):Date

Model Display - frame: JFrame - FRAME_WIDTH: int - FRAME_HEIGHT: int + createPanel(JComponent panel): void + removePanel(JComponent panel): void + closeWindow(): void <<interface>> Observer update(EventType e) StudentAllBids User - bids: List<Bid> - bidList: JList<Bid> - id: String - givenName: String Main Controller UserService - familyName: String - UserName: String + getSelectedBid(): Bid - display: Display - isStudent: boolean - userList: List<User> + setListListener(MouseClickListener listener) - user: User - isTutor: boolean - initiatedBids: List<Bid> - isAdmin: boolean - allBids: List<Bid> monitor: List<String> - monitoredBids: List<Bid> + addUser(User): void TutorMonitor - allContracts: List<Contract> + getMonitor(Iterator<JsonNode> iter): List<String> - studentExpiredContracts: List<Contract> + addBidToMonitor(Bid bid): void - activeBid: Bid - timer: Timer + monitor(Bid bid): boolean - subscriberBid: Bid - bids: List<Bid> + logIn(String username, String password): User - activeContract: Contract responseList: JList<BidResponse> - subscriberContract: Contract - responses: List<BidResponse> - activeMessage: Message - homeView: HomeView - studentAllBids: StudentAllBids - studentAllContracts: StudentAllContracts onAttached():void - createRequest: CreateRequest - studentView: StudentView UserAccount - studentMessage: StudentMessage - contractReuse: ContractReuse ContractReuse - reviseContractTerm: ReviseContractTerm - competencies: List<Competency> - tutorAllBids: TutorAllBids - qualifications: List<Qualification> - tutorAllContracts: TutorAllContracts initiatedBid: List<Bid> contracts: List<Contract> - createBid: CreateBid - additionalInfo: Object - contractList: JList<Contract> - tutorView: TutorView - tutorResponse: TutorResponseView - tutorMonitor: TutorMonitorView + setReuseContractListener(MouseClickListener listener): void - tutorMessage: TutorMessageView + getSelectedContract: Contract - createSameTutorContract: CreateSameTutorContract # setSwitchPanelListener(RemovablePanel panel, - createDifferentTutorContract: CreateDifferentTutorContract + start(): void - initStudent(): void <<abstract>> - initTutor(): void Observable fetchInitiatedBids(): void reFetchAllBids(): void - observers: Map<EventType, List<Observer>> - fetchAllMonitoredBids(): void - fetchAllContractAsFirstParty(): void fetchAllContractAsSecondParty(): void - reFetchAllContractAsFirstParty(): void + inform(EventType e): void BidService CompetencyService - reFetchAllContractAsFirstParty:() void + subscribe(EventType e, Observer o): void fetchStudentExpiredContract:(): void + unsubscribe(EventType e, Observer o): void subscribeBidCreation(): void - bidList: List<Bid> - competencyList: List<Competency> + emptySubscription(EventType e): void - subscribeBidNewResponse(): void subscribeMessage(): void subscribeContractCreation(): void + addBid(): void - showStudentMessagePanel(): void + addCompetency(): void MatchRequest - showTutorMessagePanel(): void - competency: String - hourPerLesson: String - sessionsPerWeek: String - rate: String + getCompetency(): String + getHourPerLesson(): String - type:BidType - initiator:User + getSessionsPerWeek(): String - dateCreated:Date + getRate(): String subject:Subject - closeddown:boolean - addInfo:BidAddInfo + WEEK_IN_MILLIS:int Contract + 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(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>

+ checkEligibility(User tutor):boolean + getRequestCompetency():String + getRequestHourPerLesson():String

+ getRequestSessionPerWeek():String

+ getRequestRate():String

<<abstract>>

View

- display: Display

placeComponents(): void

Component component, View view): void

component, RemovablePanel panel)

addMouseListener(JPanel main, Component

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

<<interface>>

<<interface>> NavigationPane

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

NotificationFrame - message: String List<Contract> contracts

ContractDurationFrame - three: JRadioButtonMenuItem - six: JRadioButtonMenuItem - twelve: JRadioButtonMenuItem - twentyFour: JRadioButtonMenuItem - longer: JRadioButtonMenuItem - longerDuration: JTextField - duration: ButtonGroup

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

- selectedDuration: int

+ show(): void

javax.swing.JPanel

NearExpiryContractFrame + show(): void

Bid Service

+ addBid(): void + removeBid(): void + closeBid():void + postAllBid()

bidObserver: List<Observer> - fetchedBids: List<Bid>

RemovablePane RemovablePanel(LayoutManager + onRemoved(): void

+ onAttached(): void

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

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

- 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

ContractReuse

listener):void

+ getSelectedContract():Contract

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

+ viewContracts:JButton + reuseContracts:JButton + main: RemovablePanel + StudentView(Display display, User user)

+ createMatchRequest:JButton

+ viewAllBids:JButton

StudentView

- placeComponents():void

<<abstract>> View

- user:User

- 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