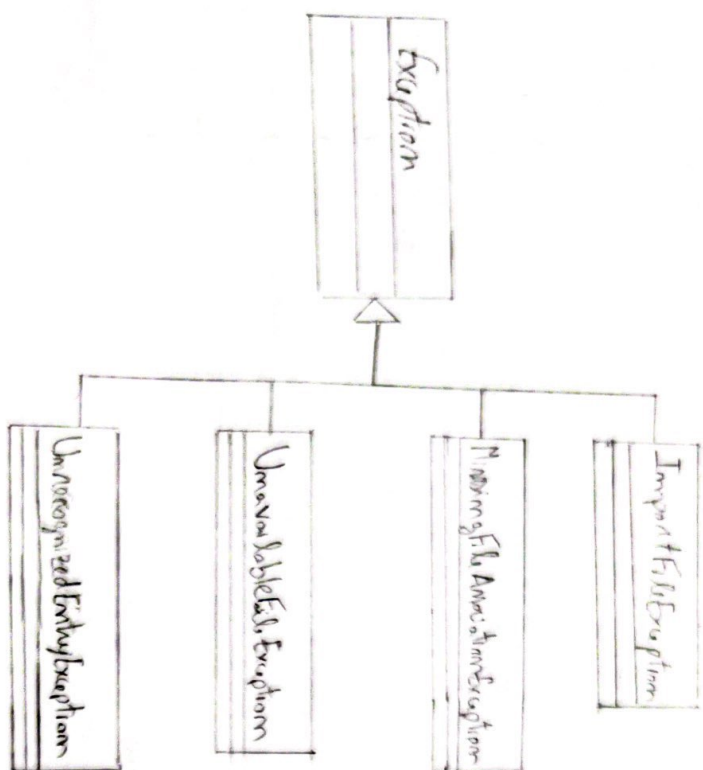


Scanned with CamScanner

Declaro por minha honra que este diagrama foi realizado apenas pelos elementos que compõem o grupo do projeto. Henrique Silva 103606





Devido por motivos humanos que este documento foi realizado apenas pelos elementos que constituem o grupo do projeto. Henrique Silva 403606

NetworkManager
-network: Network
+load(): void
+save(): void
+importFile(): void
+exportFile(): void
+openMenuClient(): void
+openMenuKup(): void
+saveFile(): void
+openFile(): void

Network
-NUM-Communications: int
-showPlans: bool
-clientClientID
+showGlobalBalance(): double
+enableClientNotification(clientKey: string): void
+disableClientNotification(clientKey: string): void
+registerClient(clientKey: string, name: string, mpy int): void
+showAllClients(): void
+showClient(clientKey: string): void
+showClientPaymentsAndDebits(clientKey: string): void
+showTerminalsWithPositiveBalance(): void
+showAllCommunications(): void
+showCommunicationsToClient(clientKey: string): void
+showClientsWithoutDebits(): void
+showClientsWithDebits(): void
+showUnusedTerminals(): void
+showAllTerminals(): void
+registerTerminal(clientKey: string): void
+openTerminalConsole(): void
+showCommunicationsFromClient(clientKey: string): void
+showGlobalContacts(clientKey: string): void
+showGlobalContacts(clientKey: string): void



Client	
- clientKey: String	
- name: String	
- email: String	
- terminated: boolean	
- notification: boolean	
- debt: double	
- debtType: ClientType	
- notifications: Notification	
+ changeClientType(ClientType: ClientType): void	
+ addTerminated(Terminated: Terminated): void	
+ addPayment(Payment: Payment): void	
+ addDebt(Debt: Debt): void	
+ removeTerminated(Terminated: Terminated): void	
+ getPayment(): void	
+ addNotification(Notification: Notification): void	
+ removeNotification(): void	

Communication	
ClientType	
NORMAL	
GOLD	
PLATINUM	

Communication	
- communicationKey: int	
- startTerminated: Terminated	
- endTerminated: Terminated	
- communicationCost: double	
- note: String	
+ startCommunication(Note: CommunicationNote): void	
+ endCommunication(Note: CommunicationNote): void	

CommunicationState	
ONGOING	
ENDED	

TextCommunication	
- message: String	

AudioCommunication	
- duration: int	

VideoCommunication	
- duration: int	

TextPlan	
- communicationKey: int	
- duration: int	
- cost: double	
- state: CommunicationState	
- note: String	
+ startTextCommunication(TextCommunication: TextCommunication, clientType: ClientType): void	
+ endTextCommunication(TextCommunication: TextCommunication, clientType: ClientType): void	
+ getAudioCosts(AudioCommunication: AudioCommunication, clientType: ClientType): double	
+ getVideoCosts(VideoCommunication: VideoCommunication, clientType: ClientType): double	

BasicPlan	
- communicationKey: int	
- duration: int	
- cost: double	
- state: CommunicationState	
- note: String	
+ startTextCommunication(TextCommunication: TextCommunication, clientType: ClientType): void	
+ endTextCommunication(TextCommunication: TextCommunication, clientType: ClientType): void	
+ getAudioCosts(AudioCommunication: AudioCommunication, clientType: ClientType): double	
+ getVideoCosts(VideoCommunication: VideoCommunication, clientType: ClientType): double	



Deleto por minha honra que este diagrama foi elaborado apenas pelos alunos que compoem o grupo do projeto.

Hernique Silva 103106

Notification
- IsCallContact: boolean
- deliveryMeans: NotificationDeliveryMeans
- notificationType: NotificationType
- terminal: Terminal
+ callClient(): string

«enumination»
NotificationDeliveryMeans
APP
MMS
SMS

«enumination»
NotificationType
OFF-TO-SILENT
OFF-TO-TOLE
SILENT-TO-TOLE
BUSY-TO-TOLE

«abstract»
Terminal
- terminalKey: string
- client: Client
- findTerminal(): Terminal[]
- state: TerminalState
- payment: double
- debts: double
- communication: Communication[]
+ addEnd (terminal: Terminal): void
+ sendInteractiveCommunication(): void
+ performPayment (value: double): void
+ removeEnd(): void
+ notifyCommunication (textCommunication: TextCommunication): void
+ showOngoingCommunication(): Communication
+ showTerminalBalance(): double
+ removeTerminal (id: TerminalState): void
+ activateInteractiveCommunication(): void
+ turnOffTerminal (id: TerminalState): void
+ turnOnTerminal (id: TerminalState): void
+ payDebt (payment: double): void

Basic Terminal
+ activateInteractiveCommunication (AudioCommunication): void
+ deactivateInteractiveCommunication (AudioCommunication): void
+ payDebt (payment: double): void

«abstract»
TerminalState
+ terminal: Terminal

Occupied State
+ disableStartCommunications(): boolean
+ canReceiveCommunication(): boolean

Off State
+ disableAllCommunications(): boolean

Quite State
+ canImviseAllCommunications(): boolean
+ canNotReceiveTextMessages(): boolean

Wait State
+ canReceiveAllCommunications(): boolean
+ canImviseAllCommunications(): boolean