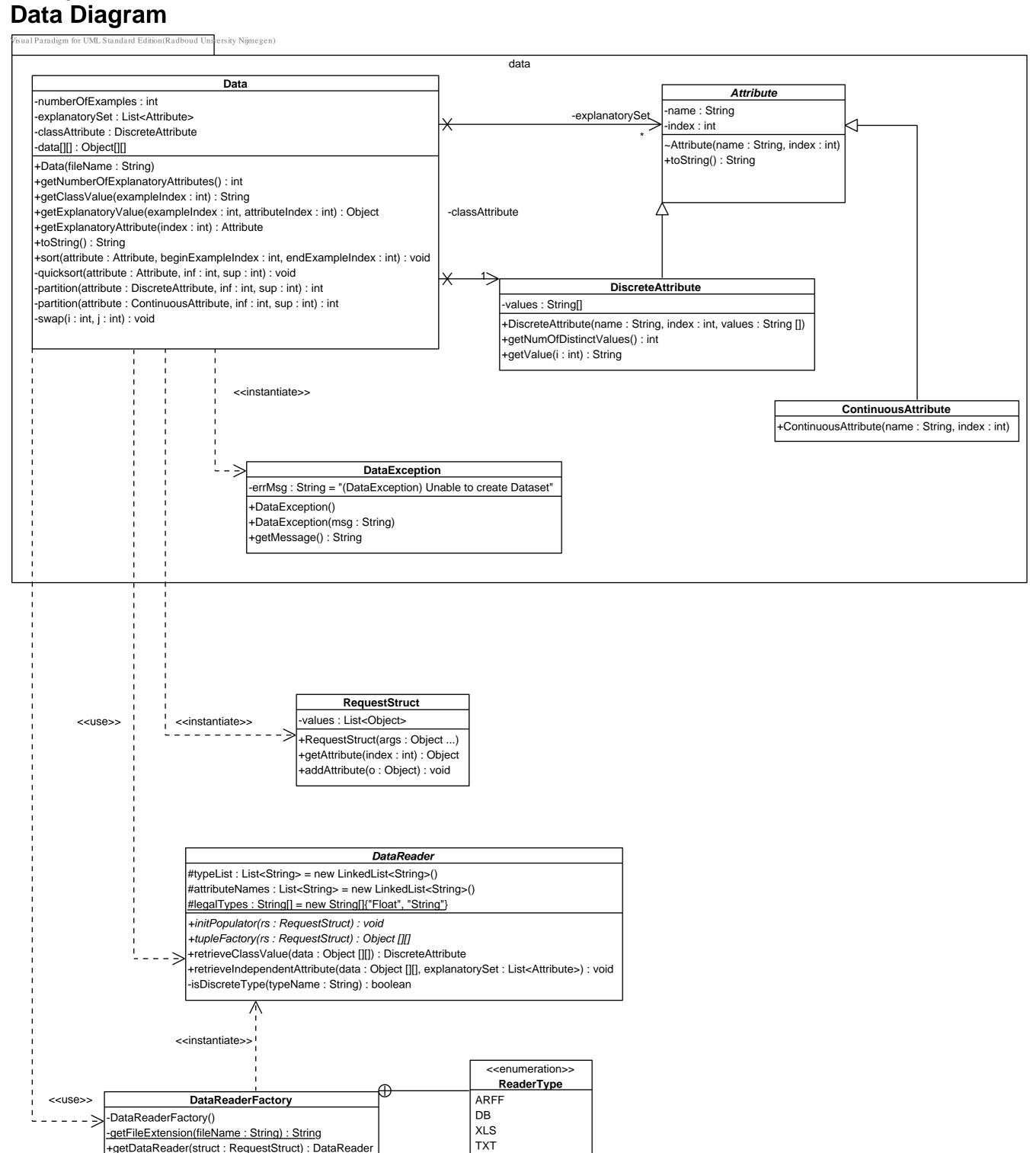
Caso di studio: "Alberi di decisione" Metodi avanzati di programmazione (a.a. 2012/2013)		
Alessandro Suglia, Gianvito Taneburgo, Daniele Loliva		

Class Diagram

+getDataReader(struct : RequestStruct) : DataReader



Class Diagram Main Diagram Visual Paradigm for UML Standard Edition(Radboud University Nijmegen) **DbAccess** #conn : Connection boundary IncorrectLoginException #currentUser : String MultiServer LoginForm -errMsg : String = "(Login Exception): Wrong combination of username and password." #currentPass : String -user : JTextField -clientArea : JList<String> ~USER_ID : String = "decisiontreeid" +IncorrectLoginException() listModel : DefaultListModel<String> -password : JPasswordField ~PASSWORD : String = "dectreepassword" +IncorrectLoginException(errMsg : String) <<use>> -logArea : JTextArea -DBMSBox : JComboBox<String> ->|+getMessage() : String ~SERVER : String = "localhost" -serverSocket : ServerSocket = null -DBMSModel : DefaultComboBoxModel<String> ~DATABASE : String = "classificationdata" -serverPort : int -correctUser : boolean = false #getDBMSDriverClass(): String LeafNode -closingSocket : Socket = null -cp : JFrame = new JFrame("Admin access") #existDatabase() : boolean -predictedClassValue : String +MultiServer(port : int) +main(args : String []) : void #getMainConnectionString(): String ~LeafNode(trainingSet : Data, beginExampleIndex : int, endExampleIndex : int) -initAndShowGUI(): void -LoginForm(port : int) #getRootConnectionString(): String -determineMostFrequentClass(trainingSet : Data) : String -closeSocket(): void <<instantiate>> -isCorrectUser(): boolean #createDatabase(s : Statement) : void -addClient(s : Socket) : void -tryLogin(port : int) : void +getNumberOfChildren(): int ~initDatabase(): void -getSocketInfo(s : Socket) : String +toString() : String +initConnection(user : String, pass : String) : void removeClient(s : Socket) : void +getConnection(): Connection +run() : void +closeConnection(): void <<use>>> <<use>>> DataException -errMsg : String = "(DataException) Unable to create Dataset" +DataException() +DataException(msg : String) ServeOneClient <<use>>> +getMessage(): String InvalidMailAddressException <<use>>> socket : Socket <<instantiate>> +InvalidMailAddressException() -in: ObjectInputStream +InvalidMailAddressException(msg : String) -out : ObjectOutputStream Data -tree : DecisionTree -numberOfExamples : int NoTreeExists +ServeOneClient(s : Socket) <<use>>> -explanatorySet : List<Attribute> -errMsg : String = "(Exception) Decision tree doesn't exists." +run(): void -classAttribute : DiscreteAttribute +NoTreeExists() -predictClass(tree : DecisionTree) : String -data[][] : Object[][] +NoTreeExists(msg : String) +Data(fileName : String) +getMessage() : String <<use>>> +getNumberOfExplanatoryAttributes(): int +getClassValue(exampleIndex : int) : String MailAddressValidator +getExplanatoryValue(exampleIndex : int, attributeIndex : int) : Object <<instantiate>> -MailAddressValidator() +getExplanatoryAttribute(index : int) : Attribute +validate(mailAddress : String) : void 111 11 <<use>>> +toString() : String +sort(attribute : Attribute, beginExampleIndex : int, endExampleIndex : int) : void 111 11 -quicksort(attribute : Attribute, inf : int, sup : int) : void RequestStruct -partition(attribute : DiscreteAttribute, inf : int, sup : int) : int -partition(attribute : ContinuousAttribute, inf : int, sup : int) : int : | -values : List<Object> <<Interface>> **DataWriterFactory** -swap(i:int, j:int):void +RequestStruct(args : Object ...) **IDataWriter** +createDataWriter(extension : String) : IDataWriter +getAttribute(index : int) : Object +write(rs : RequestStruct) : void <<instantiate>> +addAttribute(o : Object) : void -tree <<use>>> DecisionTree MailSender -root : Node -MailSender() -childTree[] : DecisionTree[] -getRecipientsAddresses(struct : MailDataTO) : InternetAddress [] +subTree(child : int) : DecisionTree -createAttachment(message : MimeMessage, struct : MailDataTO, attachmentName : String) : void <<use>>> +saveTree(fileName : String) : void -childTree -getServerMailName(struct : MailDataTO) : String +loadTree(fileName : String) : DecisionTree +sendEmail(struct : MailDataTO, attachmentName : String) : void -DecisionTree() +DecisionTree(trainingSet : Data) 111111 -learnTree(trainingSet : Data, begin : int, end : int, numberOfExamplesPerLeaf : int) : void <<use>>> SplitNode -isLeaf(trainingSet: Data, begin: int, end: int, numberOfExamplesPerLeaf: int): boolean 1 1 1 11 1 1 1 11 _ /_ >#infoGain : float -determineBestSplitNode(trainingSet : Data, begin : int, end : int) : SplitNode 11115 +toString(): String -attribute : Attribute 1111 $1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1$ #mapSplit : List<SplitInfo> 1111 1 1 1 1 +compareTo(s : SplitNode) : int #setSplitInfo(trainingSet: Data, beginExampleIndex: int, endExampleIndex: int, attribute: Attribute): void 1111 ~SplitNode(trainingSet : Data, beginExampleIndex : int, endExampleIndex : int, attribute : Attribute) 1111 Task ~getInformationGain(): float 1 1 1 1 +getNumberOfChildren(): int +doInBackground(): Boolean 1 1 1 1 ~getSplitInfo(child : int) : SplitInfo +done(): void 1111 +formulateQuery() : String - 1 1 1 1 +toString(): String <<instantiate>> -task 1 1 1 1 1111 1111 1111 **FileReceiver** 1111 FileSender 1111 -FileReceiver() 1111 -fileName : String +downloadFile(s: Socket, downDir: String, extension: String): String 1111 -s : Socket 1111 1111 -progressMonitor : JProgressBar <<instatiate>> - -logArea : JTextArea -currException : Exception **PdfCreator** -task : Task -MAX_Y : int = 780 <<use>> +FileSender(s : Socket, fileName : String, logArea : JTextArea) <u>-MAX_X</u> : int = 600 **SupportedExtensions** -xPos: int +propertyChange(evt : PropertyChangeEvent) : void -supportedExt : ArrayList<String> = new ArrayList<String>() <<instantiate>> -yPos: int +SupportedExtensions() -textDim : int = 11 +getSupportedExtension(): List<String> -titleDim: int = 15 -paragraphDim : int = 13 -textFont : PDFont = PDType1Font.TIMES_ROMAN <<enumeration>> <<use>>> -titleFont : PDFont = PDType1Font.TIMES_BOLD Utility ServerCommand -paragraphFont : PDFont = PDType1Font.TIMES_BOLD_ITALIC -Utility() \i/ <<instantiate>> BUILD_TREE_FROM_DB = 1 -currFontDim: int = com.mapgroup.classificator.utility.PdfCreator.titleDim +checkDirectory(currDir : String) : void SERIALIZE_TREE = 2 MailDataTO -contentStream : PDPageContentStream +getAvailableName(fileName : String) : String LOAD_SERIALIZED_TREE = 3 -addressees : List<String> -document : PDDocument +cleanFolder(folderName : String) : void START_PREDICTION = 4 -from : String -PdfCreator() +checkMainFolder(): void CLOSE_CONNECTION = 5 -password : String -hasNextLine(): boolean +encryptIP(ip : String) : String SUPPORTED_EXTENSIONS = 6 -mailText : String -newLines(numLines: int): void -mapChar(temp : char) : char UPLOAD_FILE = 7 -object : String -addParagraphTitle(title : String) : void -nonExistsFile(fileName : String) : boolean AVAILABLE_DAT_FILES = 8 -attachFlag : boolean +createReport(tracker: PredSessionTO, pdfFileName: String): void SEND EMAIL = 9 -track : PredSessionTO -writeText(c : Color, msg : String) : void VALIDATE_EMAIL = 10 +isAttached(): boolean CONVERT_DATASET = 11 +setAttach(flag : boolean) : void -id : int

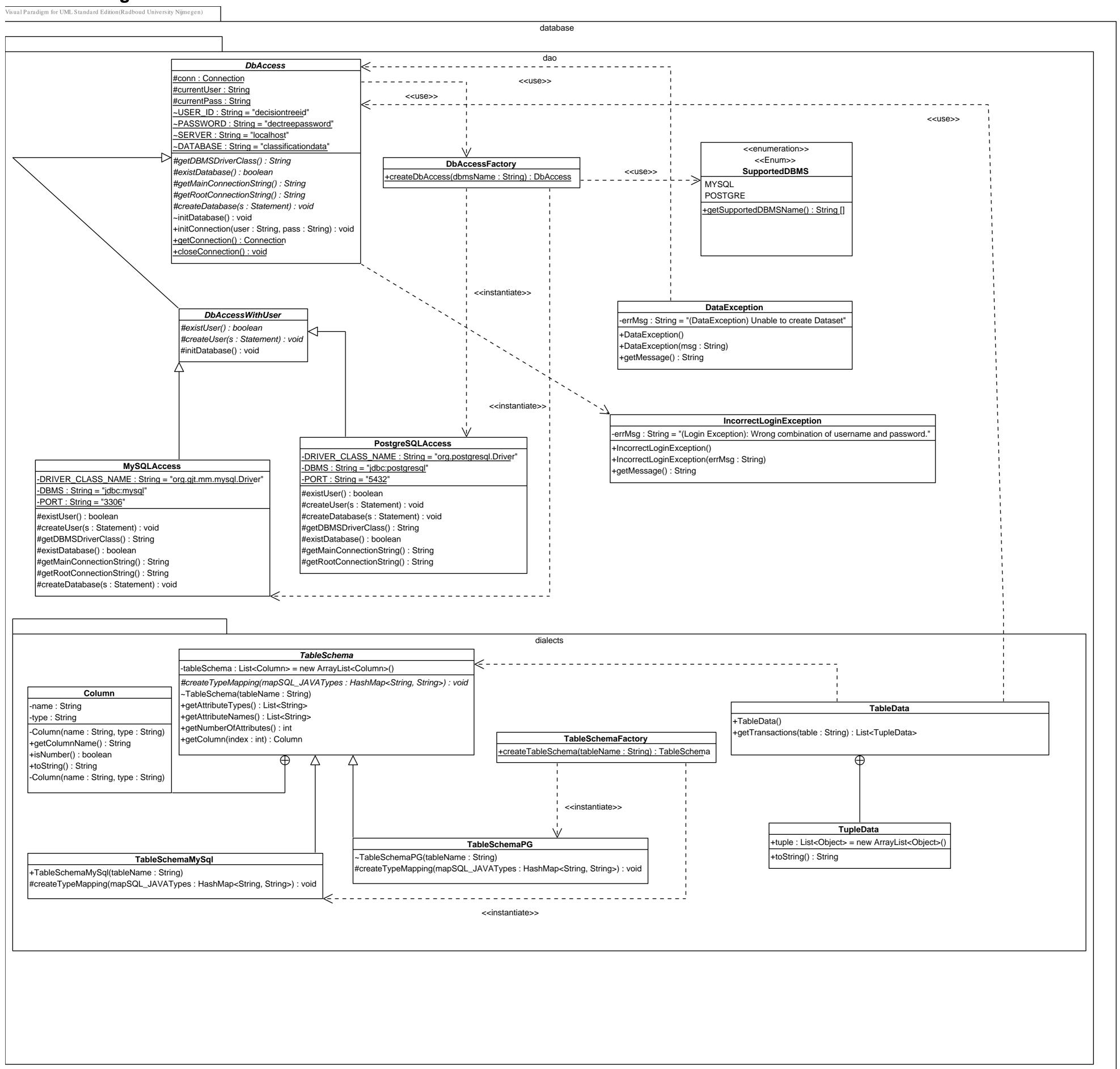
-ServerCommand(id : int)

+getSessionTO(): PredSessionTO

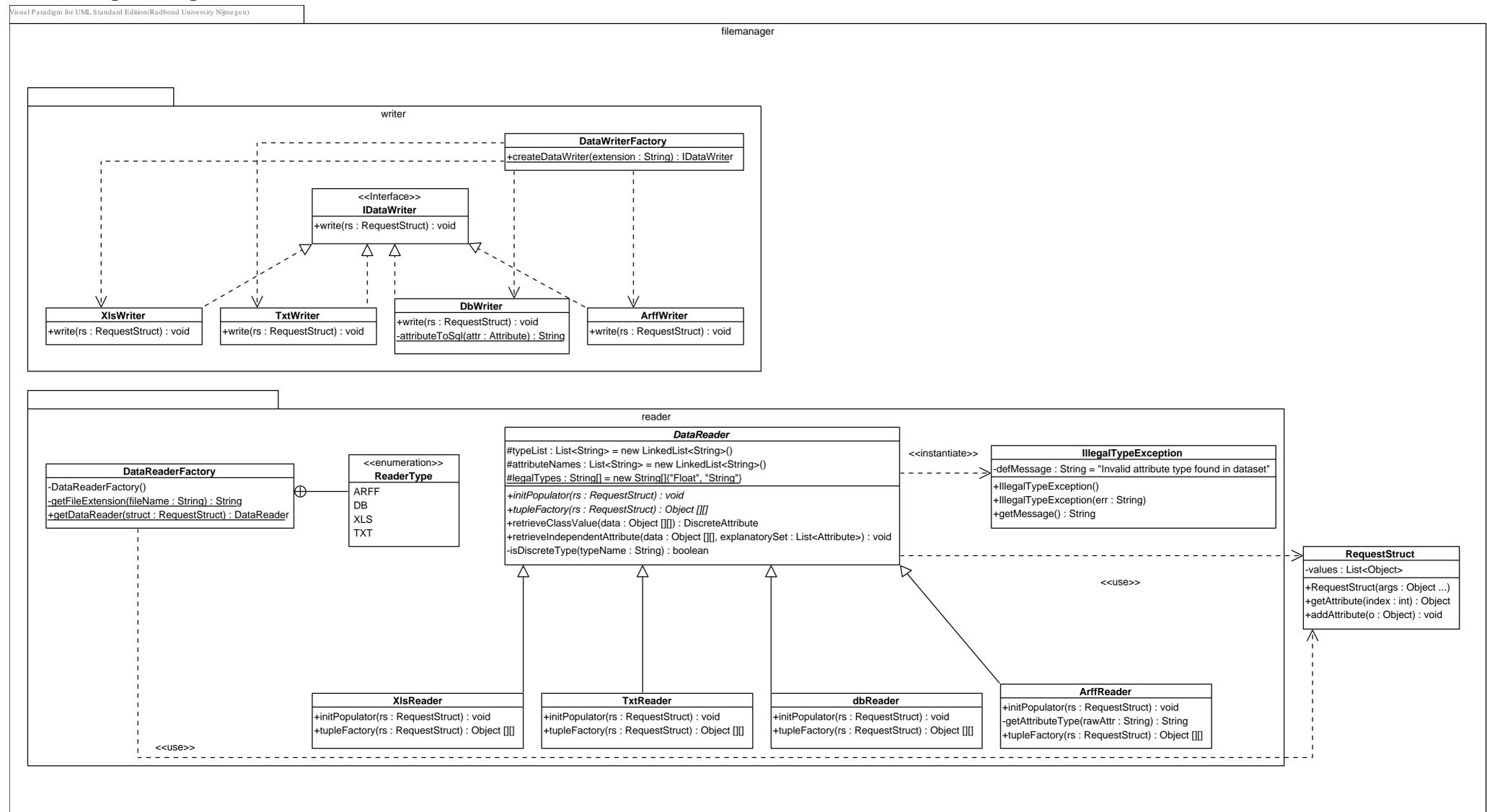
+setSessionTO(track : PredSessionTO) : void

Class Diagram

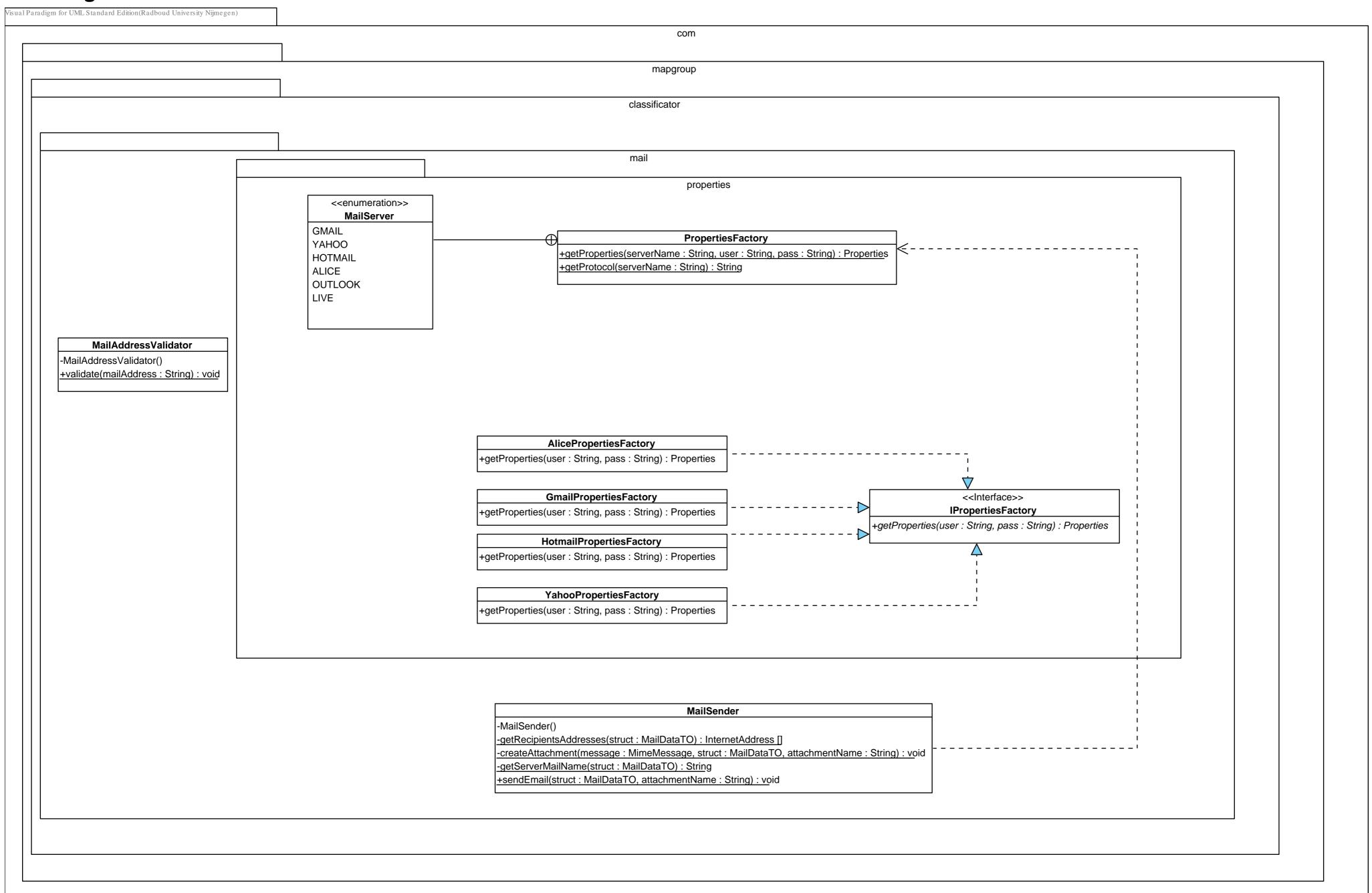
Database Diagram



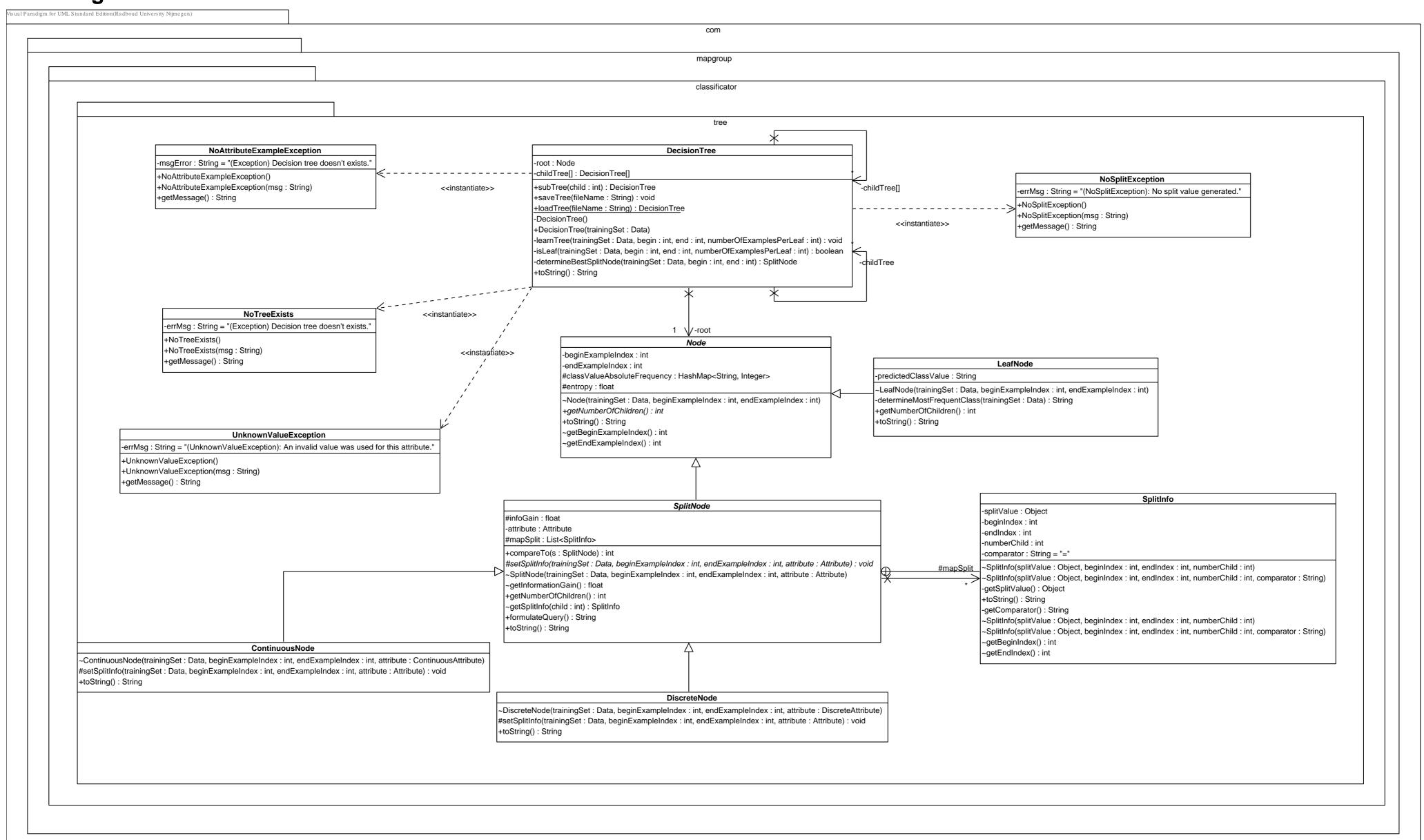
FileManager Diagram



Class Diagram Mail diagram



Class Diagram Tree diagram



Class Diagram Package diagram

