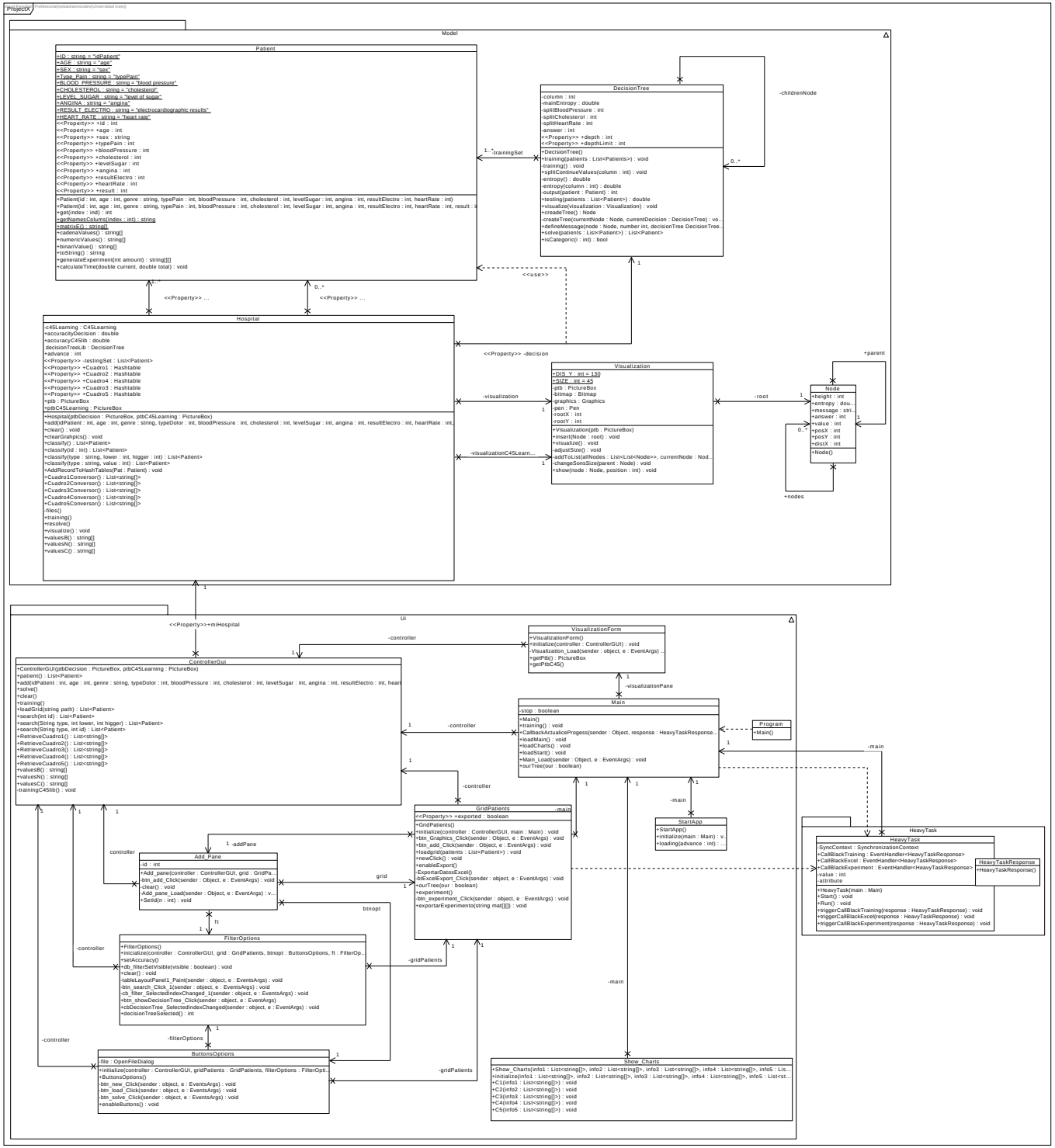
ProjectX
Model
Patient
+ID : string = "idPatient"
+AGE : string = "age"
+SEX : string = "sex"
+Type_Pain : string = "typePain"
+BLOOD_PRESSURE : string = "blood pressure"
+CHOLESTEROL : string = "cholesterol"
+LEVEL_SUGAR : string = "level of sugar"
+ANGINA : string = "angina"
+RESULT_ELECTRO : string = "electrocardiographic results"
+HEART_RATE : string = "heart rate"
<<Property>> +id : int
<<Property>> +age : int
<<Property>> +sex : string
<<Property>> +typePain : int
<<Property>> +bloodPressure : int
<<Property>> +cholesterol : int
<<Property>> +levelSugar : int
<<Property>> +angina : int
<<Property>> +resultElectro : int
<<Property>> +heartRate : int
<<Property>> +result : int
+Patient(id : int, age : int, genre : string, typePain : int, bloodPressure : int, cholesterol : int, levelSugar : int, angina : int, resultElectro : int, heartRate : int)
+Patient(id : int, age : int, genre : string, typePain : int, bloodPressure : int, cholesterol : int, levelSugar : int, angina : int, resultElectro : int, heartRate : int, result : i
+get(index : int) : int
+getNamesColumn(index : int) : string
+matrixEn() : string[]
+cadenaValues() : string[]
+numericValues() : string[]
+binariValue() : string[]
+toString() : string
+generateExperiment(int amount) : string[][]
+calculateTime(double current, double total) : void
DecisionTree
-column : int
-mainEntropy : double
-splitBloodPressure : int
-splitCholesterol : int
-splitHeartRate : int
-answer : int
<<Property>> +depth : int
<<Property>> +depthLimit : int
+DecisionTree()
+training(patients : List<Patients>) : void
-training() : void
+splitContinueValues(column : int) : void
-entropy() : double
-entropy(column : int) : double
-output(patient : Patient) : int
+testing(patients : List<Patient>) : double
+visualize(visualization : Visualization) : void
+creadeTree() : Node
-creatTree(currentNode : Node, currentDecision : DecisionTree) : vo...
+defineMessage(node : Node, number int, decisionTree DecisionTree...
+solve(patients : List<Patient>) : List<Patient>
+isCategoric(i : int) : bool
-childrenNode
0..*
1..*
-trainingSet
<<use>>
1
<<Property>> ...
<<Property>> ...
Hospital
-c4SLearning : C4SLearning
+accuracityDecision : double
+accuracyC4Slib : double
decisionTreeLib : DecisionTree
+advance : int
<<Property>> -testingSet : List<Patient>
<<Property>> +Cuadro1 : Hashtable
<<Property>> +Cuadro2 : Hashtable
<<Property>> +Cuadro4 : Hashtable
<<Property>> +Cuadro3 : Hashtable
<<Property>> +Cuadro5 : Hashtable
+ptb : PictureBox
+ptbC4SLearning : PictureBox
+Hospital(ptbDecision : PictureBox, ptbC4SLearning : PictureBox)
+add(idPatient : int, age : int, genre : string, typeDolor : int, bloodPressure : int, cholesterol : int, levelSugar : int, angina : int, resultElectro : int, heartRate : int,
+clear() : void
+clearGraphics() : void
+classify() : List<Patient>
+classify(id : int) : List<Patient>
+classify(type : string, lower : int, higger : int) : List<Patient>
+classify(type : string, value : int) : List<Patient>
+AddRecordToHashTables(Pat : Patient) : void
+Cuadro1Conversor() : List<string[]>
+Cuadro2Conversor() : List<string[]>
+Cuadro3Conversor() : List<string[]>
+Cuadro4Conversor() : List<string[]>
+Cuadro5Conversor() : List<string[]>
-files()
+training()
+resolve()
+visualize() : void
+valuesB() : string[]
+valuesN() : string[]
+valuesC() : string[]
<<Property>> -decision
-visualization
-visualizationC4SLearn...
Visualization
+DIS_Y : int = 130
+SIZE : int = 45
-ptb : PictureBox
-bitmap : Bitmap
-graphics : Graphics
-pen : Pen
-rootX : int
-rootY : int
+Visualization(ptb : PictureBox)
+insert(Node : root) : void
+visualize() : void
-adjustSize() : void
-addToList(allNodes : List<List<Node>>, currentNode : Nod...
-changeSonsSize(parent : Node) : void
+show(node : Node, position : int) : void
-root
+parent
Node
+height : int
+entropy : dou...
+message : stri...
+answer : int
+value : int
+posX : int
+posY : int
+distX : int
+Node()
+nodes
0..*
Ui
<<Property>>+miHospital
-controller
VisualizationForm
+VisualizationForm()
+initialize(controller : ControllerGUI) : void
-Visualization_Load(sender : object, e : EventArgs) ...
+getPtb() : PictureBox
+getPtbC45()
-visualizationPane
ControllerGui
+ControllerGUI(ptbDecision : PictureBox, ptbC4SLearning : PictureBox)
+patient() : List<Patient>
+add(idPatient : int, age : int, genre : string, typeDolor : int, bloodPressure : int, cholesterol : int, levelSugar : int, angina : int, resultElectro : int, heart
+solve()
+clear()
+training()
+loadGrid(string path) : List<Patient>
+search(int id) : List<Patient>
+search(String type, int lower, int higger) : List<Patient>
+search(String type, int id) : List<Patient>
+RetrieveCuadro1() : List<string[]>
+RetrieveCuadro2() : List<string[]>
+RetrieveCuadro3() : List<string[]>
+RetrieveCuadro4() : List<string[]>
+RetrieveCuadro5() : List<string[]>
+valuesB() : string[]
+valuesN() : string[]
+valuesC() : string[]
-trainingC4Slib() : void
Main
-stop : boolean
+Main()
+training() : void
+CallbackActualizeProgess(sender : Object, response : HeavyTaskResponse...
+loadMain() : void
+loadCharts() : void
+loadStart() : void
+Main_Load(sender : Object, e : EventArgs) : void
+ourTree(our : boolean)
Program
+Main()
-main
-controller
-controller
GridPatients
<<Property>> +exported : boolean
+GridPatients()
+initialize(controller : ControllerGUI, main : Main) : void
+btn_Graphics_Click(sender : Object, e : EventArgs) : void
+btn_add_Click(sender : Object, e : EventArgs) : void
+loadgrid(patients : List<Patient>) : void
+newClick() : void
+enableExport()
-ExportarDatosExcel()
-btExcelExport_Click(sender : object, e : EventArgs) : void
+ourTree(our : boolean)
+experiment()
-btn_experiment_Click(sender : object, e : EventArgs) : void
+exportarExperimento(string mat[][]) : void
-main
StartApp
+StartApp()
+initialize(main : Main) : v...
+loading(advance : int) : ...
-main
HeavyTask
HeavyTask
-SyncContext : SynchronizationContext
+CallBlackTraining : EventHandler<HeavyTaskResponse>
+CallBlackExcel : EventHandler<HeavyTaskResponse>
+CallBlackExperiment : EventHandler<HeavyTaskResponse>
-value : int
-attribute
+HeavyTask(main : Main)
+Start() : void
+Run() : void
+triggerCallBlackTraining(response : HeavyTaskResponse) : void
+triggerCallBlackExcel(response : HeavyTaskResponse) : void
+triggerCallBlackExperiment(response : HeavyTaskResponse) : void
HeavyTaskResponse
+HeavyTaskResponse()
controller
Add_Pane
-id : int
+Add_pane(controller : ControllerGUI, grid : GridPa...
-btn_add_Click(sender : Object, e : EventArgs) : void
-clear() : void
-Add_pane_Load(sender : Object, e : EventArgs) : v...
+SetId(n : int) : void
-addPane
grid
btnopt
ft
FilterOptions
+FilterOptions()
+iniciialize(controller : ControllerGUI, grid : GridPatients, btnopt : ButtonsOptions, ft : FilterOp...
+setAccuracy()
+db_filterSetVisible(visible : boolean) : void
+clear() : void
-tableLayoutPanel1_Paint(sender : object, e : EventsArgs) : void
-btn_search_Click_1(sender : object, e : EventsArgs) : void
-cb_filter_SelectedIndexChanged_1(sender : object, e : EventsArgs) : void
+btn_showDecisionTree_Click(sender : object, e : EventArgs)
+cbDecisionTree_SelectedIndexChanged(sender : object, e : EventArgs) : void
+decisionTreeSelected() : int
-controller
-gridPatients
-filterOptions
-controller
ButtonsOptions
-file : OpenFileDialog
+initialize(controller : ControllerGUI, gridPatients : GridPatients, filterOptions : FilterOpti...
+ButtonsOptions()
-btn_new_Click(sender : object, e : EventsArgs) : void
-btn_load_Click(sender : object, e : EventsArgs) : void
-btn_solve_Click(sender : object, e : EventsArgs) : void
+enableButtons() : void
-gridPatients
-main
Show_Charts
+Show_Charts(info1 : List<string[]>, info2 : List<string[]>, info3 : List<string[]>, info4 : List<string[]>, info5 : Lis...
+initialize(info1 : List<string[]>, info2 : List<string[]>, info3 : List<string[]>, info4 : List<string[]>, info5 : List<st...
+C1(info1 : List<string[]>) : void
+C2(info2 : List<string[]>) : void
+C3(info3 : List<string[]>) : void
+C4(info4 : List<string[]>) : void
+C5(info5 : List<string[]>) : void