Gui << MetroForm>> <<use>>\_ + Window() Program

<<UserControl>>

TreeTab

- OrientationComboBoxSelectedIndexChanged(obbject, EventArgs): void

root : TreeNode<CircleNode>

+ InitializeLabel() : void

ArrangeTree() : void

VerticalOrientation(): void

HorizontalOrientation(): void

+ InitializeOrientationComboBox(): void

- PicTreePaint(object, PaintEventArgs) : void

- PicTreeReSize(object, EventArgs) : void

- AddNode(TreeNode<CircleNode>, string, Node) : void

+ InitializeTreeTab(Manager) : void

## - ImportButtonClick(object, EventArgs) : void - manager - OnMouseHoverImportButton(object, EventArgs): void # Main() : void OnMouseLeaveImportButton(object, EventArgs) : void - chartTab tableTab - infoTab treeTab <<UserControl>> <<UserControl>> <<UserControl>> TableTab ChartTab InfoTab + INFO PATH: string = "../../Data/info.html" - TableTab() + ChartTab() + InitializeTableTab(Manager) : void + InitializeChartTab(Manager) : void + InfoTab() InitializeAttributeComboBox(): void GenerateTypeChart() : void - GenerateTable() : void - GenerateOdorChart(): void GenerateInfo(): void - AttributeComboBoxSelectedIndexChanged(object, EventArgs): void - GenerateRingNumberChart(): void OnMouseHoverFilterbutton(object, EventArgs) : void - GenerateBruisesChart(): void OnMouseLeaveFilterButton(object, EventArgs) : void - GenerateCapColorChart(): void FilterButton\_Click(object, EventArgs) : void

Model

```
Manager
      + TRAINING PERCENTAGE : double = 0.8
      - decisionTree : DecisionTree
      << Property>> + DecisionTree : DecisionTree
       - Manager(string)
      + GenerateDecisionTree(): void
      + DecisionTreeSuccessPercentage() : double
      + GenerateEmptyTable(): DataTable
      + GenerateDataTable() : DataTable
      + GenerateFilteredDataTable(string, string) : DataTable
      + GenerateTypeChart(): DataTable
      + GenerateOdorChart() : DataTable
      + GenerateRingNumberChart() : DataTable
      + GenerateBruisesChart(): DataTable
      + GenerateCapColorChart() : DataTable
      + Load(string): void
                                          - dataSet
                                   ...*shroom
+ CAP SHAPE : char[] = {'b', 'c', 'x', 'f', 'k', 's'}
+CAP_SURFACE: char[] = {'f', 'g', 'y', 's'}
+CAP COLOR: char[] = {'n', 'b', 'c', 'g', 'r', 'p', 'u', 'e', 'w', 'y' }
+BRUISES[]: char[] = {'t', 'f'}
+ODOR[]: char[] = {'a', 'l', 'c', 'y', 'f', 'm', 'n', 'p', 's'}
+GILL ATTACHMENT: char[] = {'a', 'd', 'f', 'n'}
+GILL SPACING: char[] = {'c', 'w', 'd'}
+GILL SIZE: char[] = \{'b', 'n'\}
+GILL COLOR: char[] = {'k', 'n', 'b', 'h', 'g', 'r', 'o', 'p', 'u', 'e', 'w', 'y'}
+STALK SHAPE: char[] = {'e', 't'}
+STALK ROOT: char[] = {'b', 'c', 'u', 'e', 'z', 'r', '?'}
+STALK SURFACE ABOVE RING: char[] = {'f', 'y', 'k', 's'}
+STALK SURFACE BELOW RING: char[] = {'f', 'y', 'k', 's'}
+STALK COLOR ABOVE RING: char[] = {'n', 'b', 'c', 'g', 'o', 'p', 'e', 'w', 'y'}
+STALK COLOR BELOW RING: char[] = {'n', 'b', 'c', 'g', 'o', 'p', 'e', 'w', 'y'}
+VEIL_TYPE: char[] = {'p', 'u'}
+VEIL COLOR: char[] = {'n', 'o', 'w', 'y'}
+RING NUMBER: char[] = {'n', 'o', 't'}
+RING TYPE: char[] = {'c', 'e', 'f', 'l', 'n', 'p', 's', 'z'}
+SPORE PRINT COLOR = {'k', 'n', 'b', 'h', 'r', 'o', 'u', 'w', 'y'}
+POPULATION: char[] = {'a', 'c', 'n', 's', 'v', 'y'}
+HABITAT: char[] = {'g', 'l', 'm', 'p', 'u', 'w', 'd'}
-type: MushroomType
-capShape:char
-capSurface:char
-capColor:char
-bruises:char
-odor:char
-gillAttachment:char
-gillSpacing:char
-gillSize: char
-gillColor: char
-stalkShape: char
-stalkRoot: char
-stalkSurfaceAboveRing: char
-stalkSurfaceBelowRing: char
-stalkColorAboveRing: char
-stalkColorBelowRing: char
-veilType: char
-veilColor: char
-ringNumber: char
-ringType: char
-sporePrintColor: char
-population: char
-habitat: char
<<Pre><<Pre>roperty>> +Type: MushroomType
<< Property>> + CapShape: char
<< Property>> + CapSurface: char
<<Pre><<Pre>roperty>> +CapColor: char
<< Property>> +Bruises: char
<<Pre><<Pre>coperty>> +Odor: char
<<Pre><<Pre>roperty>> +GillAttachment:char
<< Property>> +GillSpacing:char
<<Pre><<Pre>roperty>> +GillSize:char
<<Pre><<Pre>roperty>> +GillColor: char
<< Property>> +StalkShape: char
<<Pre><<Pre>roperty>> +StalkRoot: char
<< Property>> +StalkSurfaceAboveRing: char
<<Pre><<Pre>roperty>> +StalkSurfaceBelowRing: cnal
<< Property>> +StalkColorAboveRing: char
<< Property>> +StalkColorBelowRing: char
<< Property>> +VeilType: char
<<Pre><<Pre>roperty>> +VeilColor: char
<< Property>> +RingNumber: char
<< Property>> +RingType: char
<< Property>> +SporePrintColor: char
<< Property>> +Population: char
<< Property>> +Habitat: char
```