#### FileNode

nom: Stringtype: FileTypeparent: FileNode# actual file: File

- children: Map<String, FileNode>

+ FileNode(nom: String, parent: FileNode, type: FileType, actual file: File)

+ findChild(name: String): FileNode

+ getNome(): String

+ setName(nom: String): void

+ getChildren(): Collection<FileNode>

+ getParen(): FileNode

+ setParent(p: FileNode): void

+ addChild(name: String, childNode: FileNode): void

+ isDirectory(): boolean + buildTree(file: File): void

+ findFileExtension(file: File): String {exceptions =

UnexpectedFileTypeException}

+ buildTreeWithImages(): ArrayList<PhotoNode> {exceptions= UnexpectedFileTypeException, IOException}

+ openInFinder(): String

+ findImagesWithTag(tagWeAreLookingFor: String):

ArrayList<PhotoNode>



# <<enumeration>> FileType

FILE DIRECTORY PHOTO

## **PhotoNodeTest**

+ PhotoNodeTest()

+ testInitializeTags(): void

+ testCreateNewTags1(): void

+ testCreateNewTags2(): void

+ testCreateNewTags3(): void

+ testCreateNewTags4(): void

+ testCreateNewTags5(): void

# Tag

tags: ArrayList<String>

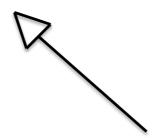
+ Tag()

+ newTag(newTag: String): void

+ deleteTag(toDelete: String): void

## UnexpectedFileTypeException

+ UnexpectedFileTypeException(message: String)



### **PhotoNode**

tags: ArrayList<String>

+ allTags: ArrayList<Strings>

+ PhotoNode(name: String, parent: FileNode, file: File)

+ <u>isValidTag(newTag : String): boolean</u> {exceptions =

IOException}

+ addTagToPhoto(newTag: String): void

+ deleteTagFromPhoto(toDelete: String): void

+ stripOfTags(name: String): String

+ confirmRename(): String

+ confirmRename(renameTo: String): void

+ untilNextWhitespace(s: String, i: int): String

+ setTags(): void

+ getTags(): ArrayList<Strings>

+ initializeTags(): void {exceptions = IOException}

+ createNewTag(tagName: String): boolean {exception

= IOException}

+ <u>deleteTag(tagName: String): boolean</u> {exception =

IOException}

+ photoWithTag(al: ArrayList<PhotoNode>,

tagName:String): ArrayList<String>

+ mostCommonTag(al: ArrayList<PhotoNode>): String

+ mostTags(al: ArrayList<PhotoNode>): PhotoNode

+ toString(): String



#### PhotoRenamerShell PhotoRenamerShell

- file: File
- + log: ArrayList<LogEntry>
- + PhotoRenamerShell(file: File)
- + getFile(): File
- + readySet():ArrayList<PhotoNode> {exceptions =
  - UnexpectedFileTypeException, IOException}
- + readySet(f: File): ArrayList<PhotoNode> {exceptions
  - = UnexpectedFileTypeException, IOException)
- + run(): void {excpetions = IOException,
  - UnexpectedFileTypeException, URISyntaxException}
- + buildLog(photos: ArrayList<PhotoNode>): void
- + deserializeLog(): void
- serializeLog(): void
- + displayLog(currName: String): void
- getLog(currName :String): LogEntry



# LogEntry

- serialVersionUID: long = 1
- currName: String
- entries: ArrayList<String[]>
- + LogEntry (currName: String)
- + newName(newName: String): void
- + getRevertedName(index: int): String
- + toString(): String
- # getCurrName(): String
- setCurrName(newName: String): void



<<interface>>

Serializable

#### PhotoRenamer

- photos: ArrayList<PhotoNode>
- instance PhotoRenamer();
- currentPhoto: PhotoNode
- currentIndex: int directory: File
- masterlist: JList<String> currentTags: JList<String>
- photoTags: DefaultListModel<String>
- jMasterList: DefaultListModel<String>
- dWindow: JFrame iFrame: JFrame
- panelEast: JPanel
- panelCenter: JPanel
- panelWest: JPanel
- imageInformation: JPanel
- noTagName: JLabel
- yesTagName: JLabel
- actualImage: JPanel
- # imageLabel: JLabel
- # imIcon: ImageIcon
- currentTagsScrollPane: JscrollPane
- PhotoRenamer()
- PhotoRenamer(file: File) {exceptions =
  - UnexpectedFileTypeException, IOException}
- + getInstanceOf(): PhotoRenamer
- + buildDWindow(): JFrame
- + buildImageTagger(): void {exceptions =
  - UnexpectedFileTypeException, IOException}
- + buildPanelEast(): void
- + updateIcon(): void
- + buildImagePanel(): void
- buildImageInfoPanel(): void
- + buildPhotoTagsPanel(): void
- + buildTagFunctionsPanel(): JPanel
- + PhotoNode getCurrentPhoto(): PhotoNode
- + setCurrentPhoto(currentPhoto: PhotoNode): void
- + getDirectory(): File
- + setDirectory(directory: File): void
- + nextPhoto(): void
- + previousPhoto(): void
- # setCurrentTags(currentTags: JList<String): void
- getPhotoTags():DefaultListModel<String>
- # setPhotoTags(photoTags: DefaultListModel<String>): void
- getjMasterList():DefaultListModel<String>
- # setjMasterList(jMasterList: DefaultListModel<String>): void
- getdWindow(): JFrame
- # setdWindow(dWindow: JFrame): void
- # getiFrame(): JFrame
- # setiFrame(iFrame: JFrame): void
- getPanelEast():JPanel
- # setPanelEast(panelEast: JPanel): void
- # getPanelCenter(): JPanel
- # setPanelCenter(panelCenter: JPanel): void
- getPanelWest():JPanel
- setPanelWest(panelWest: JPanel): void