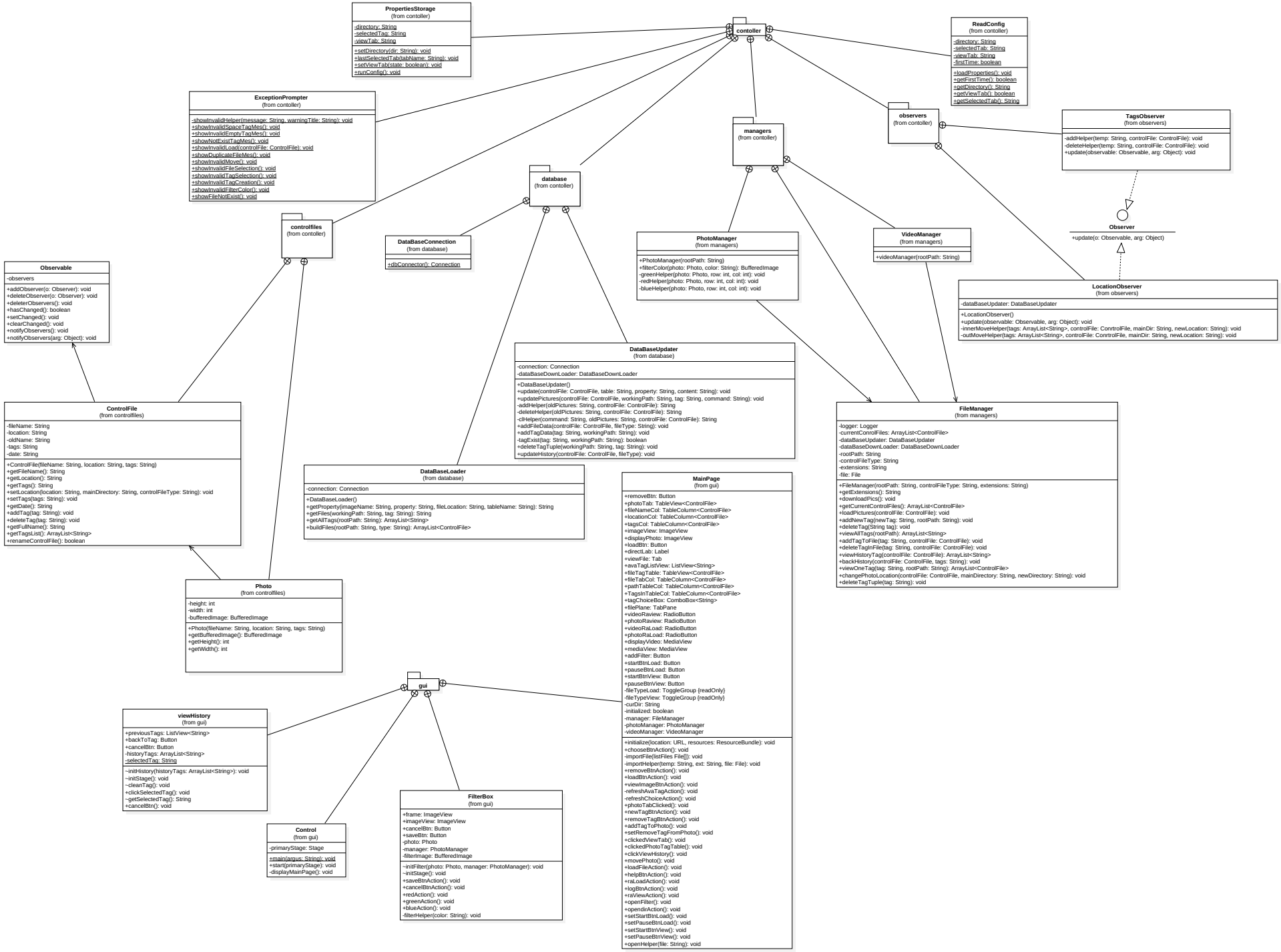
Model::Main


PropertiesStorage
(from contoller)
-directory: String
-selectedTag: String
-viewTab: String
+setDirectory(dir: String): void
+lastSelectedTab(tabName: String): void
+setViewTab(state: boolean): void
+runConfig(): void
contoller
ReadConfig
(from contoller)
-directory: String
-selectedTab: String
-viewTab: String
-firstTime: boolean
+loadProperties(): void
+getFirstTime(): boolean
+getDirectory(): String
+getViewTab(): boolean
+getSelectedTab(): String
ExceptionPrompter
(from contoller)
-showInvalidHelper(message: String, warningTitle: String): void
+showInvalidSpaceTagMes(): void
+showInvalidEmptyTagMes(): void
+showNotExistTagMes(): void
+showInvalidLoad(controlFile: ControlFile): void
+showDuplicateFileMes(): void
+showInvalidMove(): void
+showInvalidFileSelection(): void
+showInvalidTagSelection(): void
+showInvalidTagCreation(): void
+showInvalidFilterColor(): void
+showFileNotExist(): void
managers
(from contoller)
observers
(from contoller)
TagsObserver
(from observers)
-addHelper(temp: String, controlFile: ControlFile): void
-deleteHelper(temp: String, controlFile: ControlFile): void
+update(observable: Observable, arg: Object): void
database
(from contoller)
controlfiles
(from contoller)
DataBaseConnection
(from database)
+dbConnector(): Connection
PhotoManager
(from managers)
+PhotoManager(rootPath: String)
+filterColor(photo: Photo, color: String): BufferedImage
-greenHelper(photo: Photo, row: int, col: int): void
-redHelper(photo: Photo, row: int, col: int): void
-blueHelper(photo: Photo, row: int, col: int): void
VideoManager
(from managers)
+videoManager(rootPath: String)
Observer
+update(o: Observable, arg: Object)
Observable
-observers
+addObserver(o: Observer): void
+deleteObserver(o: Observer): void
+deleteObservers(): void
+hasChanged(): boolean
+setChanged(): void
+clearChanged(): void
+notifyObservers(): void
+notifyObservers(arg: Object): void
LocationObserver
(from observers)
-dataBaseUpdater: DataBaseUpdater
+LocationObserver()
+update(observable: Observable, arg: Object): void
-innerMoveHelper(tags: ArrayList<String>, controlFile: ConrtrolFile, mainDir: String, newLocation: String): void
-outMoveHelper(tags: ArrayList<String>, controlFile: ConrtrolFile, mainDir: String, newLocation: String): void
DataBaseUpdater
(from database)
-connection: Connection
-dataBaseDownLoader: DataBaseDownLoader
+DataBaseUpdater()
+update(controlFile: ControlFile, table: String, property: String, content: String): void
+updatePictures(controlFile: ControlFile, workingPath: String, tag: String, command: String): void
-addHelper(oldPictures: String, controlFile: ControlFile): String
-deleteHelper(oldPictures: String, controlFile: ControlFile): String
-clHelper(command: String, oldPictures: String, controlFile: ControlFile): String
+addFileData(controlFile: ControlFile, fileType: String): void
+addTagData(tag: String, workingPath: String): void
-tagExist(tag: String, workingPath: String): boolean
+deleteTagTuple(workingPath: String, tag: String): void
+updateHistory(controlFile: ControlFile, fileType): void
ControlFile
(from controlfiles)
-fileName: String
-location: String
-oldName: String
-tags: String
-date: String
+ControlFile(fileName: String, location: String, tags: String)
+getFileName(): String
+getLocation(): String
+getTags(): String
+setLocation(location: String, mainDirectory: String, controlFileType: String): void
+setTags(tags: String): void
+getDate(): String
+addTag(tag: String): void
+deleteTag(tag: String): void
+getFullName(): String
+getTagsList(): ArrayList<String>
+renameControlFile(): boolean
DataBaseLoader
(from database)
-connection: Connection
+DataBaseLoader()
+getProperty(imageName: String, property: String, fileLocation: String, tableName: String): String
+getFiles(workingPath: String, tag: String): String
+getAllTags(rootPath: String): ArrayList<String>
+buildFiles(rootPath: String, type: String): ArrayList<ControlFile>
FileManager
(from managers)
-logger: Logger
-currentConrolFiles: ArrayList<ControlFile>
-dataBaseUpdater: DataBaseUpdater
-dataBaseDownLoader: DataBaseDownLoader
-rootPath: String
-controlFileType: String
-extensions: String
-file: File
+FileManager(rootPath: String, controlFileType: String, extensions: String)
+getExtensions(): String
+downloadPics(): void
+getCurrentControlFiles(): ArrayList<ControlFile>
+loadPictures(controlFile: ControlFile): void
+addNewTag(newTag: String, rootPath: String): void
+deleteTag(String tag): void
+viewAllTags(rootPath): ArrayList<String>
+addTagToFile(tag: String, controlFile: ControlFile): void
+deleteTagInFile(tag: String, controlFile: ControlFile): void
+viewHistoryTag(controlFile: ControlFile): ArrayList<String>
+backHistory(controlFile: ControlFile, tags: String): void
+viewOneTag(tag: String, rootPath: String): ArrayList<ControlFile>
+changePhotoLocation(controlFile: ControlFile, mainDirectory: String, newDirectory: String): void
+deleteTagTuple(tag: String): void
Photo
(from controlfiles)
-height: int
-width: int
-bufferedImage: BufferedImage
+Photo(fileName: String, location: String, tags: String)
+getBufferedImage(): BufferedImage
+getHeight(): int
+getWidth(): int
MainPage
(from gui)
+removeBtn: Button
+photoTab: TableView<ControlFile>
+fileNameCol: TableColumn<ControlFile>
+locationCol: TableColumn<ControlFile>
+tagsCol: TableColumn<ControlFile>
+imageView: ImageView
+displayPhoto: ImageView
+loadBtn: Button
+directLab: Label
+viewFile: Tab
+avaTagListView: ListView<String>
+fileTagTable: TableView<ControlFile>
+fileTabCol: TableColumn<ControlFile>
+pathTableCol: TableColumn<ControlFile>
+TagsInTableCol: TableColumn<ControlFile>
+tagChoiceBox: ComboBox<String>
+filePlane: TabPane
+videoRaview: RadioButton
+photoRaview: RadioButton
+videoRaLoad: RadioButton
+photoRaLoad: RadioButton
+displayVideo: MediaView
+mediaView: MediaView
+addFilter: Button
+startBtnLoad: Button
+pauseBtnLoad: Button
+startBtnView: Button
+pauseBtnView: Button
-fileTypeLoad: ToggleGroup {readOnly}
-fileTypeView: ToggleGroup {readOnly}
-curDir: String
-initialized: boolean
-manager: FileManager
-photoManager: PhotoManager
-videoManager: VideoManager
+initialize(location: URL, resources: ResourceBundle): void
+chooseBtnAction(): void
-importFile(listFiles File[]): void
-importHelper(temp: String, ext: String, file: File): void
+removeBtnAction(): void
+loadBtnAction(): void
+viewImageBtnAction(): void
-refreshAvaTagAction(): void
-refreshChoiceAction(): void
+photoTabClicked(): void
+newTagBtnAction(): void
+removeTagBtnAction(): void
+addTagToPhoto(): void
+setRemoveTagFromPhoto(): void
+clickedViewTab(): void
+clickedPhotoTagTable(): void
+clickViewHistory(): void
+movePhoto(): void
+loadFileAction(): void
+helpBtnAction(): void
+raLoadAction(): void
+logBtnAction(): void
+raViewAction(): void
+openFilter(): void
+opendirAction(): void
+setStartBtnLoad(): void
+setPauseBtnLoad(): void
+setStartBtnView(): void
+setPauseBtnView(): void
+openHelper(file: String): void
gui
viewHistory
(from gui)
+previousTags: ListView<String>
+backToTag: Button
+cancelBtn: Button
-historyTags: ArrayList<String>
-selectedTag: String
~initHistory(historyTags: ArrayList<String>): void
~initStage(): void
~cleanTag(): void
+clickSelectedTag(): void
~getSelectedTag(): String
+cancelBtn(): void
Control
(from gui)
-primaryStage: Stage
+main(argus: String): void
+start(primaryStage): void
-displayMainPage(): void
FilterBox
(from gui)
+frame: ImageView
+imageView: ImageView
+cancelBtn: Button
+saveBtn: Button
-photo: Photo
-manager: PhotoManager
-filterImage: BufferedImage
~initFilter(photo: Photo, manager: PhotoManager): void
~initStage(): void
+saveBtnAction(): void
+cancelBtnAction(): void
+redAction(): void
+greenAction(): void
+blueAction(): void
-filterHelper(color: String): void