gui::Main

+ system: GoGoPhotoSystem

+ main(args String[])
+ start(primaryStage Stage)
+ stop()

gui::MainSceneController

- path: String
- logBook: Button
- viewImage: Button

+ logBookButtonClicked()
+ viewImageButtonClicked()
+ getPath(): String
+ setPath(path: String)
+ initialize(location: URL, resources: ResourceBundle)

gui::AlterBox

+ display(title: String, message: String)

«interface»
initialize

Application

gui::ViewHistoryController

- goBack: Button
- usedNameColumn: TableView
- table: TableView

+ goBackButtonClicked()
+ initialize(location: URL, resources: ResourceBundle)
+ backToOldButtonCliked()
- getChosenImage(): Image

gui::LocalOnlyController

- chosenImage:Image
- chooseButton:Button
- goBackButton:Button
- table:TableView<ExistingImage>
- nameColumn:TableColumn
- pathColumn:TableColumn

+ initialize(location: URL, resources: ResourceBundle)
+ chooseButtonClicked()
+ goBackButtonClicked()
~ getChosenImage(): Image

gui:: ListAllController

- chosenImage:Image
- chooseButton:Button
- goBackButton:Button
- table:TableView<ExistingImage>
- nameColumn:TableColumn
- pathColumn:TableColumn

+ initialize(location: URL, resources: ResourceBundle)
+ chooseButtonClicked()
+ goBackButtonClicked()
~ getChosenImage():Image

gui::ViewImageController

- listAll: Button
- localOnly: Button

+ localOnlyButtonClicked()
+ initialize(location: URL, resources: ResourceBundle)
+ listAllButtonClicked()

gui::AddTagController

- moveImage: Button
- viewHistory: Button
- goBack: Button
- text: Text
- imageView: ImageView
- newTag: TextField
- table1: TableView
- table2: TableView
- tagColumn1: TableColumn
- tagcolumn2: TableColumn

+ goBackButtonClicked()
+ initialize(location: URL, resources: ResourceBundle)
+ viewHistoryButtonClicked()
+ moveTheImageButtonClicked()
+ addButtonClicked()
+ deleteButtonClicked()
- getChosenImage(): Image

gui::LogSceneController

- goBack: Button
- table: TableView<ExistingImage>
- nameColumn: TableColumn
- pathColumn: TableColumn

+ initialize(location: URL, resources: ResourceBundle)
- goBackButtonClicked()

Model::UsedName

- name: String

~ UsedName(name:Strin
+ getName(): String

Model::DirectoryManager

~ DirectoryManager()
+ findLocalImages(path: String): ArrayList<ExistingImage>
+ findAllImages(path: String): ArrayList<ExistingImage>
- arrayConcatenation(arr1: ArrayList<ExistingImage>, arr2: ArrayList<ExistingImage>): ArrayList<ExistingImage>
- isImage(file: File): boolean

Model::Tag

- name: String

~ Tag(name:String)
+ getName(): String
+ getOriginalName():String

Model::LoggerManager

+ ReadLogMessages(logger: Logger, path: String): ArrayList<LogMessage>

Model::Image

- name: String
- tags: ArrayList<String>
- path: String
- renamingHistory: ArrayList<String>
- imageId: int
- type: String

+ Image(name:String, type:String, path:String)
~ getType(): String
~ getName(): String
+ getTagList(): ArrayList<Tag>
+ resetTags(tagInputs: ArrayList<String>)
+ getUsedNameList(): ArrayList<UsedName>
+ setName(newName: String)
+ getRenamingHistory(): ArrayList<String>
+ addNewNameToHistory(newName: String)
+ getPath(): String
+ setPath(newPath: String)
~ getTags(): ArrayList<String>
+ notifyUp(key: String)
+ notifyDown(key: String)
+ addTags(String)
+ deleteTags(tag: String)
+ getFullName():String
+ toString(): String
+ getInitialName(): String
+ getAllTags(): String

Model::ImageManager

- images: ArrayList<Image>

~ ImageManager()
+ addImage(picture: Image)
+ foundImage(path: String): Image
+ deleteImage(picture: Image): boolean
+ renameFile(image: Image, newName: String)
+ moveImageToDirectory(image: Image, new_path: String)
+ backToOldName(image: Image, name: String)
+ deleteTagFromName(image: Image, name: String): boolean
+ addExistingTagToName(image: Image, tagNmae: String): boolean
+ addTagToName(image: Image, tagNmae:String): boolean
+ saveToImageManagerFile(filePath: String)
+ foundTag(mage: Image, newTag: String): boolean

Model::ExistingImage

- name: String
- path: String

~ ExistingImage(name: String, path: String)
+ getImageName(): String
+ getImageType(): String
+ getName(): String
+ getPath(): String

**«interface»
Observer**

+ updateUp(String)
+ updateDown(String)

Model::LogMessage

- time: String
- name: String

~ LogMessage(time:String, name:String)
+ toString():String
+ getTime(): String
+ getContent(): String

Model::MyFormatter

- df: DateFormatter

+ format(record: LogRecord): String
+ getHead(h: Handler): String
+ getTail(h: Handler): String

Model::GoGoPhotoSystem

- tagManager: TagManager
- imageManager: ImageManager
+ logger: Logger
- directoryManager: DirectoryManager

+ GoGoPhotoSystem(tagFilePath: String, imageFilePath: String)
- readFromTagManagerFile(path: String)
- readFromImageManagerFile(path: String)
+ saveToTagManagerFile(filePath: String)
+ saveToImageManagerFile(filePath: String)
+ getDirectoryManager(): DirectoryManager
+ getTagManager(): TagManager
+ getImageManager(): ImageManager

Model::TagManager

~ tags: HashMap<String, Integer>

~ TagManager()
~ updateUp(key: String)
~ updateDown(key: String)
~ saveToTagManagerFile(filePath: String)
+ getExistingTags(): ArrayList<Tag>

Formatter

**«interface»
Observable**

+ notifyUp(key: String)
+ notifyDown(key: String)

**«interface»
Serializable**