

Jonathan Konopka

Timothy Zhang

Photos Assignment UML Diagram

Classes

Photos
+users: List<User> -serialVersionUID: long +stg: Stage +user: User +album: Album +photo: Photo
+Start(primaryStage:Stage): void +Stop(): void +Main(args:String[]): void +writeApp(users:List<user>): void +readApp(): List<user> +createStock(): void

User
-serialVersionUID: long -albums: List<Album> -name: String -tags: List<String>
+deleteAlbum(String name): void +setName(String name): void +getName(): String +addAlbum(Album album): void +getAlbums(): List<Album> +addTag(String tag): void +getTags(): List<String> +toString(): String

Album
-serialVersionUID: long -photos: List<Photo> -name: String
+setName(String name): void +getName(): String +addPhoto(Photo photo): void +getPhoto(): List<Photo> +containsPhoto(Photo photo): Boolean +removePhoto(Photo photo): void +earliestDate(): String +latestDate(): String +numberOfPhotos(): String +toString(): String

Photo
-path: String -serialVersionUID: long -caption: String -date: long -tags: Map<String, String>
+getPath(): String +getDate(): long +getCaption(): String +setCaption(String cap): void +addTag(String tag, String value): void +getTagValue(String tag): String

AlbumViewController
+album: Album +display: ObservableList<Photo>
+initialize(): void +back(ActionEvent e): void +add(ActionEvent e): void +delete(ActionEvent e): void +open(ActionEvent e): void

SearchController
+album: Album +user: User +albums: List<Album> +checkIfSearched: int +display: ObservableList<Photo>
+initialize(): void +tagSearch(ActionEvent e): void +create(ActionEvent e): void +dateSearch(ActionEvent e): void +back(ActionEvent e): void

AdminController
+users: List<User> +display: ObservableList<Photo>
+initialize(): void +logout(ActionEvent e): void +add(ActionEvent e): void +delete(ActionEvent e): void

TagSearchController
+user: User +tags: List<String> +albums: List<Album> +photos: List<Photo> +tempAlbum: Album +searched: List<Photo> +searchedAlbum: Album
+initialize(): void +back(ActionEvent e): void +search(ActionEvent e): void

PhotoCell
-loadFXML(): void #updateItem(Photo photo, boolean empty): void

PhotoCellFactory
+call(ListView<Photo> arg0): ListCell<Photo>

LoginController
+initialize(): void +login(ActionEvent e): void +checkNames(String name): User

MetadataController
+initialize(): void +saveCaption(ActionEvent e): void +back(ActionEvent e): void +saveTag(ActionEvent e): void

MovePhotoController
+initialize(): void +cancel(ActionEvent e): void +move(ActionEvent e): void +copy(ActionEvent e): void

DateSearchController
+user: User +albums: List<Album> +photos: List<Photo> +tempAlbum: Album +searched: List<Photo> +searchedAlbum: Album
+initialize(): void +back(ActionEvent e): void +search(ActionEvent e): void -inDateRange(Date input, Date starting, Date ending): boolean

TagsAddController
+tags: List<String> +display: ObservableList<Photo> +user: User
+initialize(): void +back(ActionEvent e): void +delete(ActionEvent e): void

ImageAddController
+album: Album
+initialize(): void +select(ActionEvent e): void +add(ActionEvent e): void +goBack(ActionEvent e): void

AlbumListController
+user: user +albums: List<Album> +display: ObservableList<Photo>
+initialize(): void +logout(ActionEvent e): void +addTag(ActionEvent e): void +add(ActionEvent e): void +search(ActionEvent e): void +delete(ActionEvent e): void +open(ActionEvent e): void +rename(ActionEvent e): void

ImageViewController
+initialize(): void +back(ActionEvent e): void +lastPhoto(ActionEvent e): void +nextPhoto(ActionEvent e): void +move(ActionEvent e): void +editMeta(ActionEvent e): void

Relations

