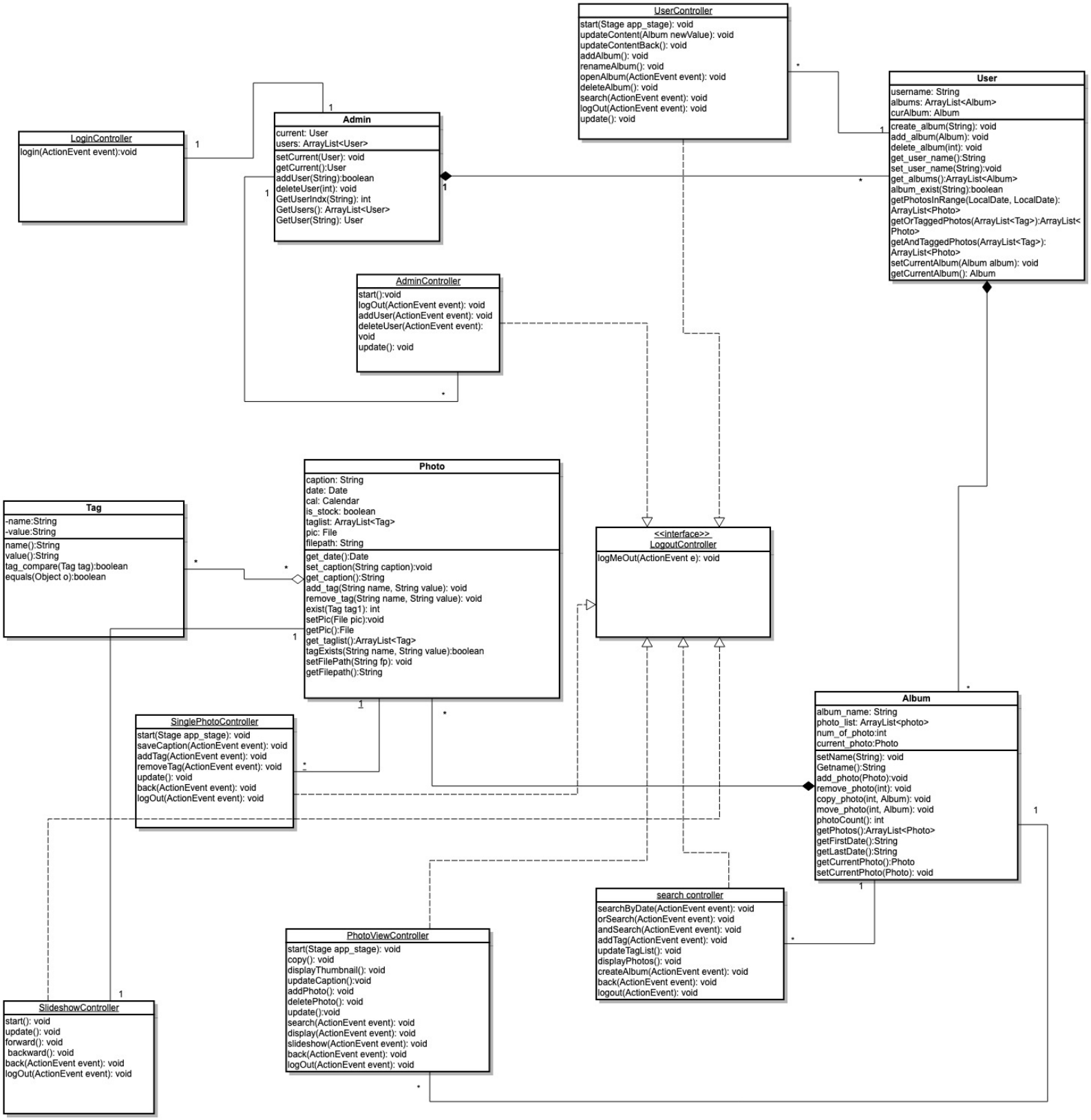
UserController
start(Stage app_stage): void
updateContent(Album newValue): void
updateContentBack(): void
addAlbum(): void
renameAlbum(): void
openAlbum(ActionEvent event): void
deleteAlbum(): void
search(ActionEvent event): void
logOut(ActionEvent event): void
update(): void
User
username: String
albums: ArrayList<Album>
curAlbum: Album
create_album(String): void
add_album(Album): void
delete_album(int): void
get_user_name():String
set_user_name(String):void
get_albums():ArrayList<Album>
album_exist(String):boolean
getPhotosInRange(LocalDate, LocalDate): ArrayList<Photo>
getOrTaggedPhotos(ArrayList<Tag>):ArrayList<Photo>
getAndTaggedPhotos(ArrayList<Tag>): ArrayList<Photo>
setCurrentAlbum(Album album): void
getCurrentAlbum(): Album
LoginController
login(ActionEvent event):void
Admin
current: User
users: ArrayList<User>
setCurrent(User): void
getCurrent():User
addUser(String):boolean
deleteUser(int): void
GetUserIndx(String): int
GetUsers(): ArrayList<User>
GetUser(String): User
AdminController
start():void
logOut(ActionEvent event): void
addUser(ActionEvent event): void
deleteUser(ActionEvent event): void
update(): void
Photo
caption: String
date: Date
cal: Calendar
is_stock: boolean
taglist: ArrayList<Tag>
pic: File
filepath: String
get_date():Date
set_caption(String caption):void
get_caption():String
add_tag(String name, String value): void
remove_tag(String name, String value): void
exist(Tag tag1): int
setPic(File pic):void
getPic():File
get_taglist():ArrayList<Tag>
tagExists(String name, String value):boolean
setFilePath(String fp): void
getFilepath():String
Tag
-name:String
-value:String
name():String
value():String
tag_compare(Tag tag):boolean
equals(Object o):boolean
<<interface>>
LogoutController
logMeOut(ActionEvent e): void
Album
album_name: String
photo_list: ArrayList<photo>
num_of_photo:int
current_photo:Photo
setName(String): void
Getname():String
add_photo(Photo):void
remove_photo(int): void
copy_photo(int, Album): void
move_photo(int, Album): void
photoCount(): int
getPhotos():ArrayList<Photo>
getFirstDate():String
getLastDate():String
getCurrentPhoto():Photo
setCurrentPhoto(Photo): void
SinglePhotoController
start(Stage app_stage): void
saveCaption(ActionEvent event): void
addTag(ActionEvent event): void
removeTag(ActionEvent event): void
update(): void
back(ActionEvent event): void
logOut(ActionEvent event): void
search controller
searchByDate(ActionEvent event): void
orSearch(ActionEvent event): void
andSearch(ActionEvent event): void
addTag(ActionEvent event): void
updateTagList(): void
displayPhotos(): void
createAlbum(ActionEvent event): void
back(ActionEvent event): void
logout(ActionEvent): void
PhotoViewController
start(Stage app_stage): void
copy(): void
displayThumbnail(): void
updateCaption():void
addPhoto(): void
deletePhoto(): void
update():void
search(ActionEvent event): void
display(ActionEvent event): void
slideshow(ActionEvent event): void
back(ActionEvent event): void
logOut(ActionEvent event): void
SlideshowController
start(): void
update(): void
forward(): void
backward(): void
back(ActionEvent event): void
logOut(ActionEvent event): void