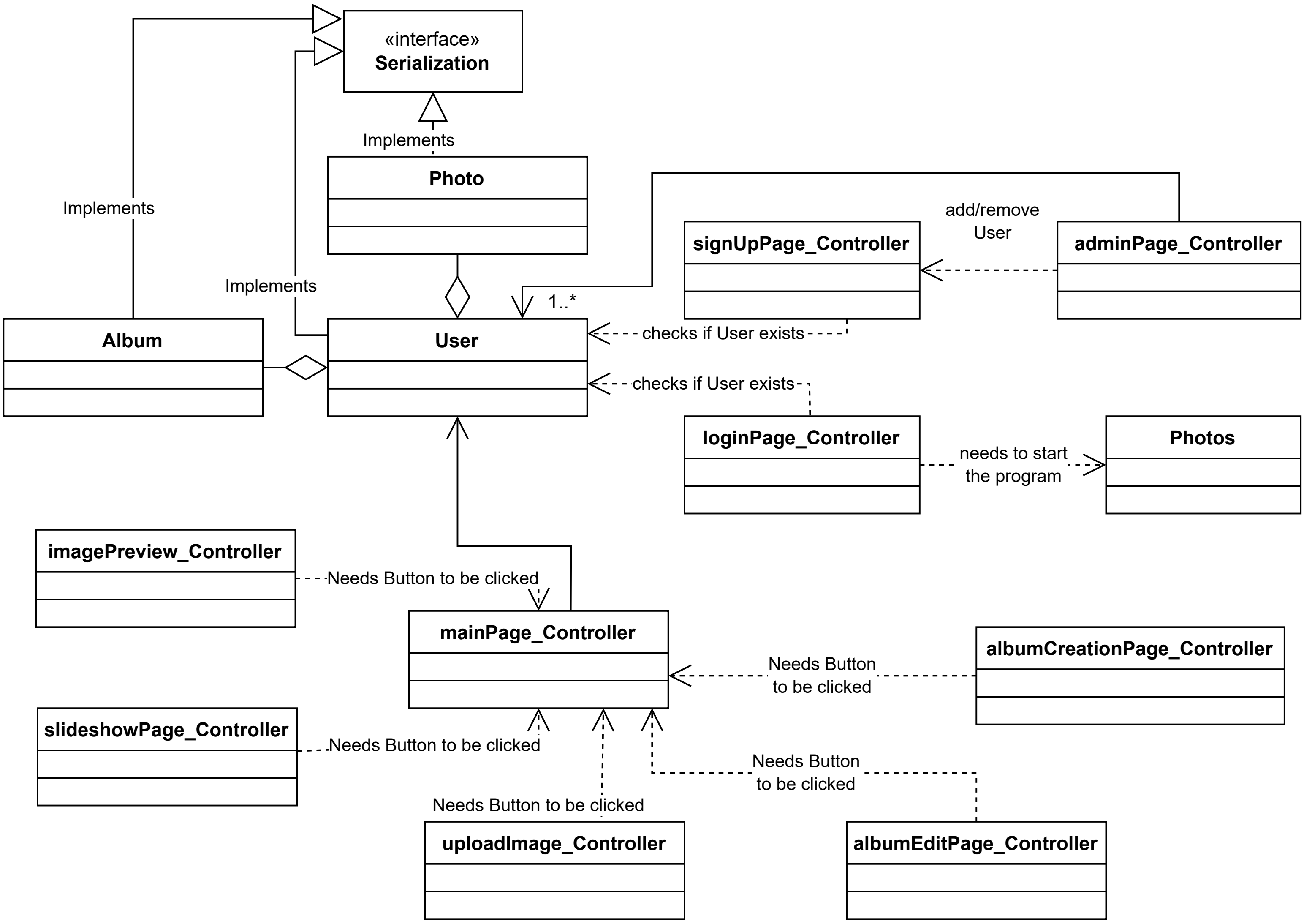
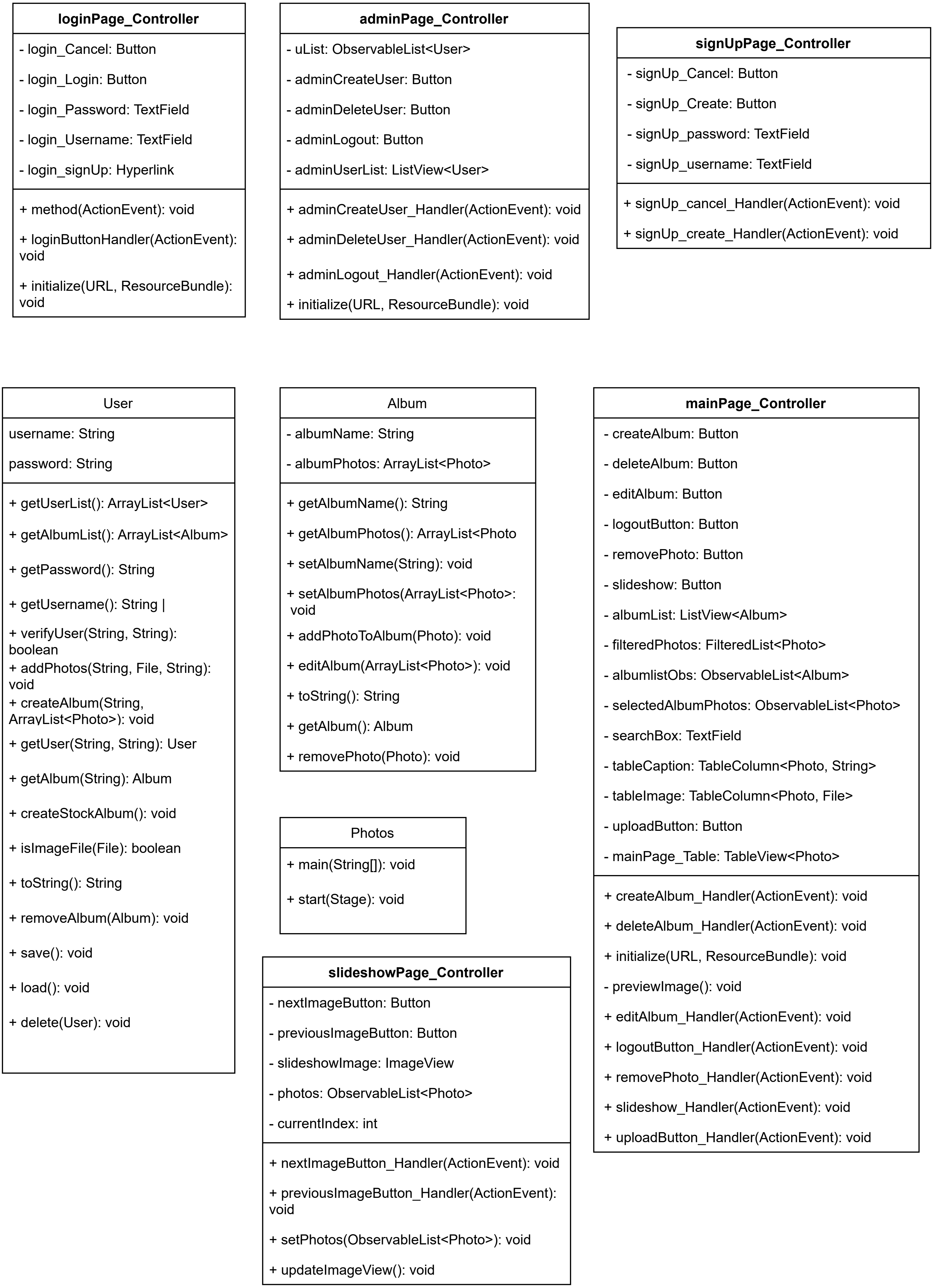
Photos

UML Diagram of the interaction
between all the controllers with
the Objects (User, Photo, Album)





loginPage_Controller

- login_Cancel: Button
- login_Login: Button
- login_Password: TextField
- login_Username: TextField
- login_signUp: Hyperlink

+ method(ActionEvent): void
+ loginButtonHandler(ActionEvent): void
+ initialize(URL, ResourceBundle): void

adminPage_Controller

- uList: ObservableList<User>
- adminCreateUser: Button
- adminDeleteUser: Button
- adminLogout: Button
- adminUserList: ListView<User>

+ adminCreateUser_Handler(ActionEvent): void
+ adminDeleteUser_Handler(ActionEvent): void
+ adminLogout_Handler(ActionEvent): void
+ initialize(URL, ResourceBundle): void

signUpPage_Controller

- signUp_Cancel: Button
- signUp_Create: Button
- signUp_password: TextField
- signUp_username: TextField

+ signUp_cancel_Handler(ActionEvent): void
+ signUp_create_Handler(ActionEvent): void

User

username: String

password: String

+ getUserList(): ArrayList<User>
+ getAlbumList(): ArrayList<Album>
+ getPassword(): String
+ getUsername(): String |
+ verifyUser(String, String): boolean
+ addPhotos(String, File, String): void
+ createAlbum(String, ArrayList<Photo>): void
+ getUser(String, String): User
+ getAlbum(String): Album
+ createStockAlbum(): void
+ isImageFile(File): boolean
+ toString(): String
+ removeAlbum(Album): void
+ save(): void
+ load(): void
+ delete(User): void

Album

- albumName: String
- albumPhotos: ArrayList<Photo>

+ getAlbumName(): String
+ getAlbumPhotos(): ArrayList<Photo
+ setAlbumName(String): void
+ setAlbumPhotos(ArrayList<Photo>: void
+ addPhotoToAlbum(Photo): void
+ editAlbum(ArrayList<Photo>): void
+ toString(): String
+ getAlbum(): Album
+ removePhoto(Photo): void

Photos

+ main(String[]): void
+ start(Stage): void

slideshowPage_Controller

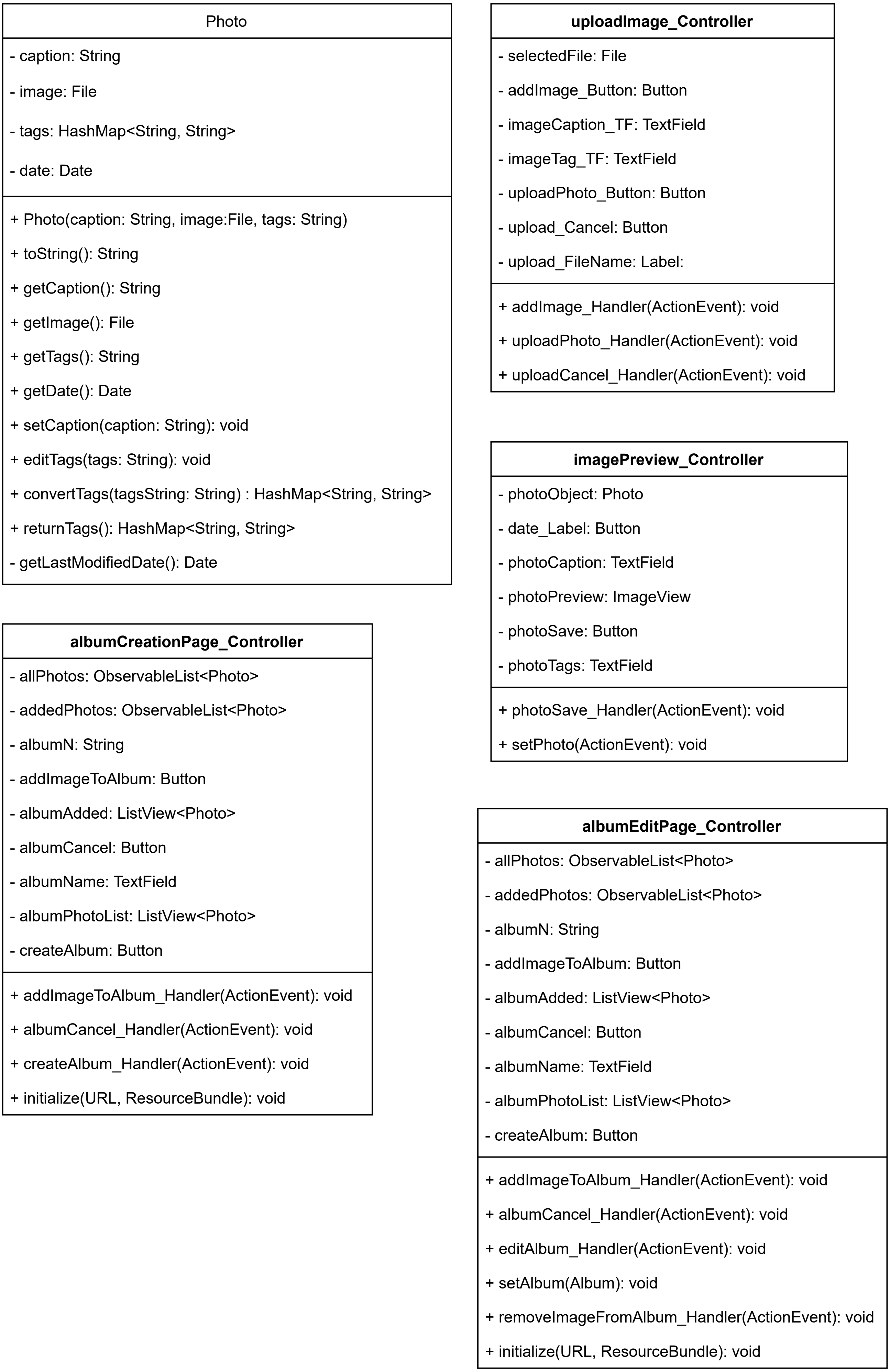
- nextImageButton: Button
- previousImageButton: Button
- slideshowImage: ImageView
- photos: ObservableList<Photo>
- currentIndex: int

+ nextImageButton_Handler(ActionEvent): void
+ previousImageButton_Handler(ActionEvent): void
+ setPhotos(ObservableList<Photo>): void
+ updateImageView(): void

mainPage_Controller

- createAlbum: Button
- deleteAlbum: Button
- editAlbum: Button
- logoutButton: Button
- removePhoto: Button
- slideshow: Button
- albumList: ListView<Album>
- filteredPhotos: FilteredList<Photo>
- albumlistObs: ObservableList<Album>
- selectedAlbumPhotos: ObservableList<Photo>
- searchBox: TextField
- tableCaption: TableColumn<Photo, String>
- tableImage: TableColumn<Photo, File>
- uploadButton: Button
- mainPage_Table: TableView<Photo>

+ createAlbum_Handler(ActionEvent): void
+ deleteAlbum_Handler(ActionEvent): void
+ initialize(URL, ResourceBundle): void
- previewImage(): void
+ editAlbum_Handler(ActionEvent): void
+ logoutButton_Handler(ActionEvent): void
+ removePhoto_Handler(ActionEvent): void
+ slideshow_Handler(ActionEvent): void
+ uploadButton_Handler(ActionEvent): void



Photo

- caption: String
- image: File
- tags: HashMap<String, String>
- date: Date

+ Photo(caption: String, image:File, tags: String)
+ toString(): String
+ getCaption(): String
+ getImage(): File
+ getTags(): String
+ getDate(): Date
+ setCaption(caption: String): void
+ editTags(tags: String): void
+ convertTags(tagsString: String) : HashMap<String, String>
+ returnTags(): HashMap<String, String>
- getLastModifiedDate(): Date

uploadImage_Controller

- selectedFile: File
- addImage_Button: Button
- imageCaption_TF: TextField
- imageTag_TF: TextField
- uploadPhoto_Button: Button
- upload_Cancel: Button
- upload_FileName: Label:

+ addImage_Handler(ActionEvent): void
+ uploadPhoto_Handler(ActionEvent): void
+ uploadCancel_Handler(ActionEvent): void

imagePreview_Controller

- photoObject: Photo
- date_Label: Button
- photoCaption: TextField
- photoPreview: ImageView
- photoSave: Button
- photoTags: TextField

+ photoSave_Handler(ActionEvent): void
+ setPhoto(ActionEvent): void

albumCreationPage_Controller

- allPhotos: ObservableList<Photo>
- addedPhotos: ObservableList<Photo>
- albumN: String
- addImageToAlbum: Button
- albumAdded: ListView<Photo>
- albumCancel: Button
- albumName: TextField
- albumPhotoList: ListView<Photo>
- createAlbum: Button

+ addImageToAlbum_Handler(ActionEvent): void
+ albumCancel_Handler(ActionEvent): void
+ createAlbum_Handler(ActionEvent): void
+ initialize(URL, ResourceBundle): void

albumEditPage_Controller

- allPhotos: ObservableList<Photo>
- addedPhotos: ObservableList<Photo>
- albumN: String
- addImageToAlbum: Button
- albumAdded: ListView<Photo>
- albumCancel: Button
- albumName: TextField
- albumPhotoList: ListView<Photo>
- createAlbum: Button

+ addImageToAlbum_Handler(ActionEvent): void
+ albumCancel_Handler(ActionEvent): void
+ editAlbum_Handler(ActionEvent): void
+ setAlbum(Album): void
+ removeImageFromAlbum_Handler(ActionEvent): void
+ initialize(URL, ResourceBundle): void