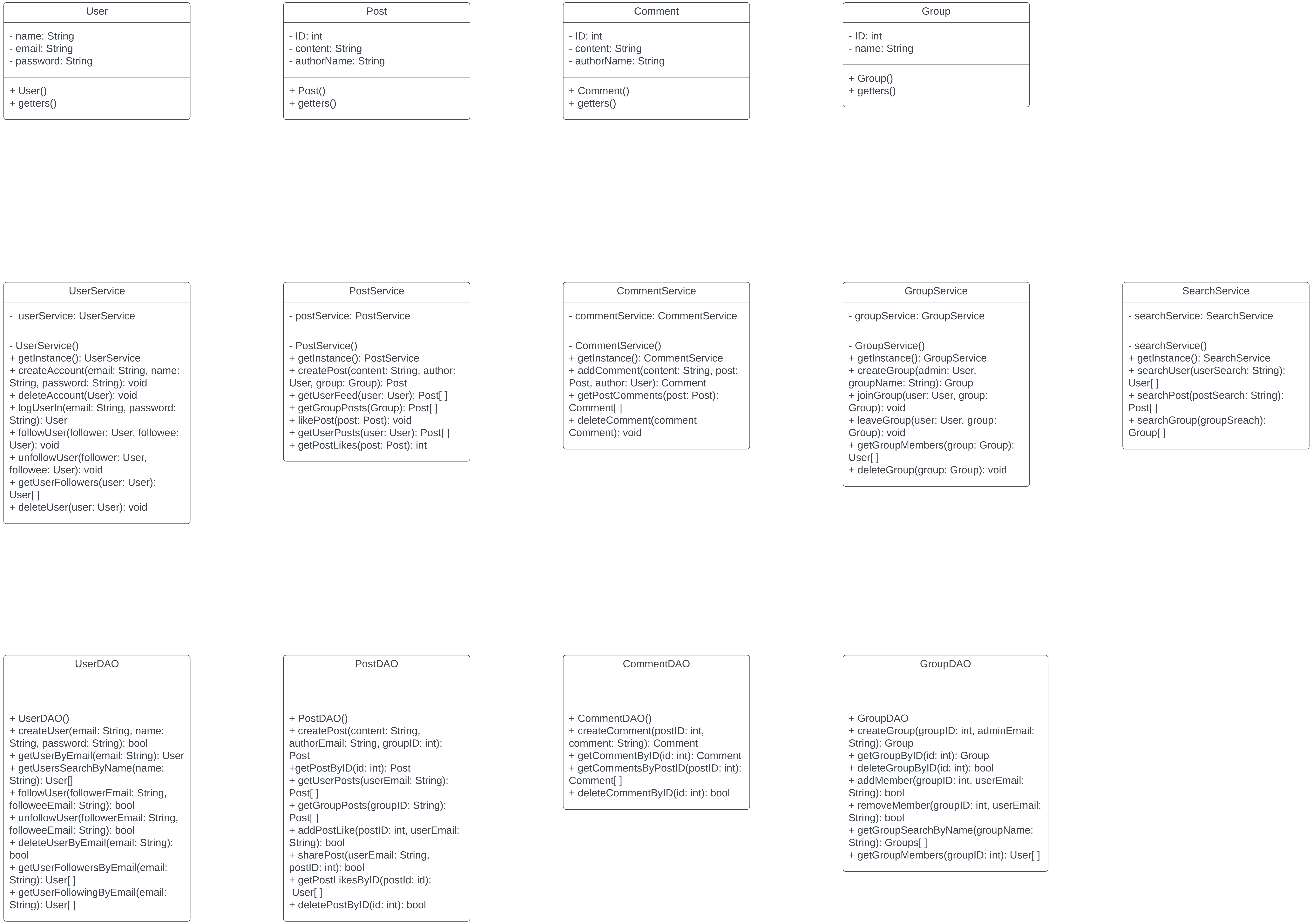

User

- name: String
- email: String
- password: String

+ User()
+ getters()

Post

- ID: int
- content: String
- authorName: String

+ Post()
+ getters()

Comment

- ID: int
- content: String
- authorName: String

+ Comment()
+ getters()

Group

- ID: int
- name: String

+ Group()
+ getters()

UserService

- userService: UserService

- UserService()
+ getInstance(): UserService
+ createAccount(email: String, name: String, password: String): void
+ deleteAccount(User): void
+ logUserIn(email: String, password: String): User
+ followUser(follower: User, followee: User): void
+ unfollowUser(follower: User, followee: User): void
+ getUserFollowers(user: User): User[ ]
+ deleteUser(user: User): void

PostService

- postService: PostService

- PostService()
+ getInstance(): PostService
+ createPost(content: String, author: User, group: Group): Post
+ getUserFeed(user: User): Post[ ]
+ getGroupPosts(Group): Post[ ]
+ likePost(post: Post): void
+ getUserPosts(user: User): Post[ ]
+ getPostLikes(post: Post): int

CommentService

- commentService: CommentService

- CommentService()
+ getInstance(): CommentService
+ addComment(content: String, post: Post, author: User): Comment
+ getPostComments(post: Post): Comment[ ]
+ deleteComment(comment Comment): void

GroupService

- groupService: GroupService

- GroupService()
+ getInstance(): GroupService
+ createGroup(admin: User, groupName: String): Group
+ joinGroup(user: User, group: Group): void
+ leaveGroup(user: User, group: Group): void
+ getGroupMembers(group: Group): User[ ]
+ deleteGroup(group: Group): void

SearchService

- searchService: SearchService

- searchService()
+ getInstance(): SearchService
+ searchUser(userSearch: String): User[ ]
+ searchPost(postSearch: String): Post[ ]
+ searchGroup(groupSreach): Group[ ]

UserDAO

+ UserDAO()
+ createUser(email: String, name: String, password: String): bool
+ getUserByEmail(email: String): User
+ getUsersSearchByName(name: String): User[]
+ followUser(followerEmail: String, followeeEmail: String): bool
+ unfollowUser(followerEmail: String, followeeEmail: String): bool
+ deleteUserByEmail(email: String): bool
+ getUserFollowersByEmail(email: String): User[ ]
+ getUserFollowingByEmail(email: String): User[ ]

PostDAO

+ PostDAO()
+ createPost(content: String, authorEmail: String, groupID: int): Post
+getPostByID(id: int): Post
+ getUserPosts(userEmail: String): Post[ ]
+ getGroupPosts(groupID: String): Post[ ]
+ addPostLike(postID: int, userEmail: String): bool
+ sharePost(userEmail: String, postID: int): bool
+ getPostLikesByID(postId: id): User[ ]
+ deletePostByID(id: int): bool

CommentDAO

+ CommentDAO()
+ createComment(postID: int, comment: String): Comment
+ getCommentByID(id: int): Comment
+ getCommentsByPostID(postID: int): Comment[ ]
+ deleteCommentByID(id: int): bool

GroupDAO

+ GroupDAO
+ createGroup(groupID: int, adminEmail: String): Group
+ getGroupByID(id: int): Group
+ deleteGroupByID(id: int): bool
+ addMember(groupID: int, userEmail: String): bool
+ removeMember(groupID: int, userEmail: String): bool
+ getGroupSearchByName(groupName: String): Groups[ ]
+ getGroupMembers(groupID: int): User[ ]