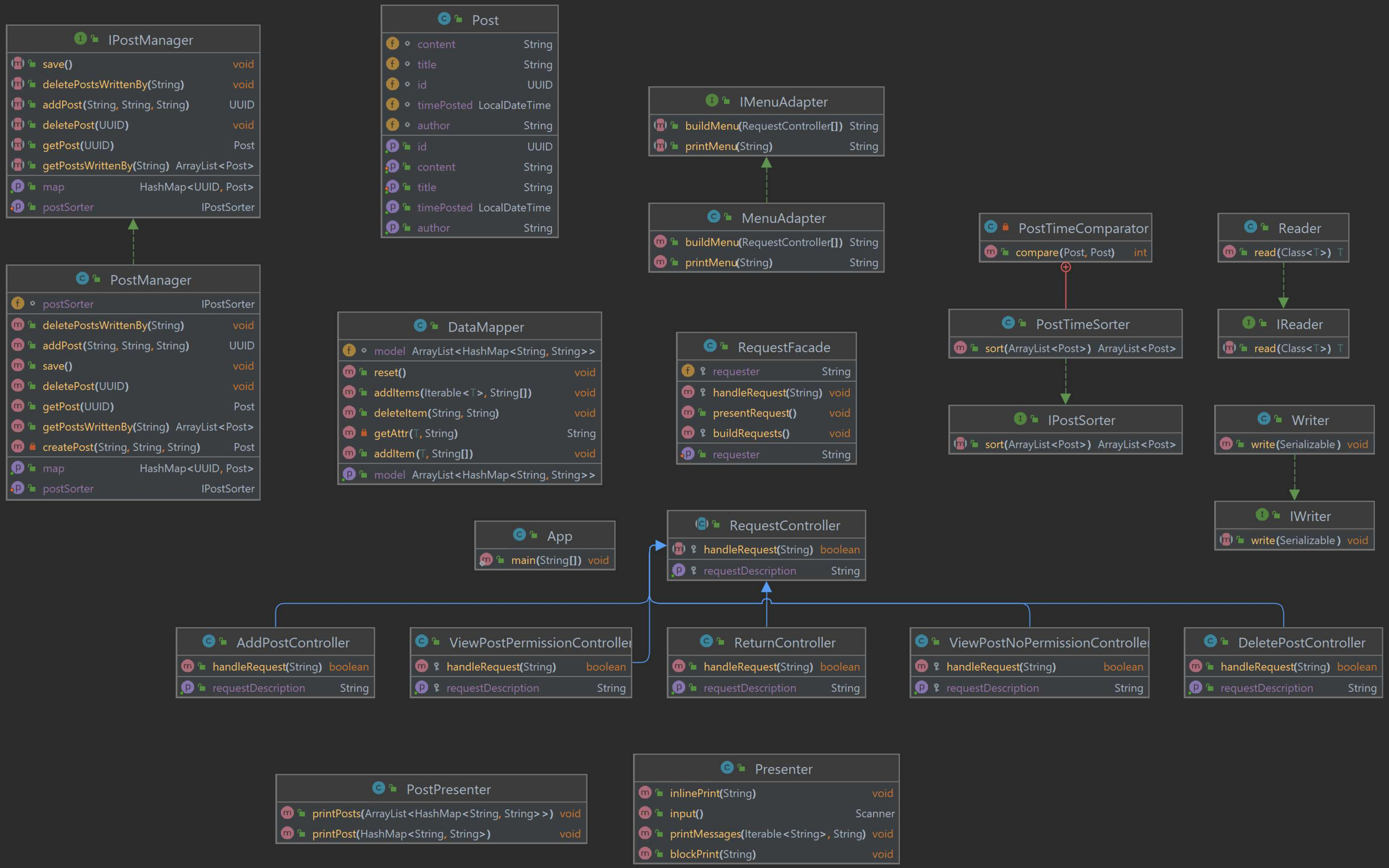
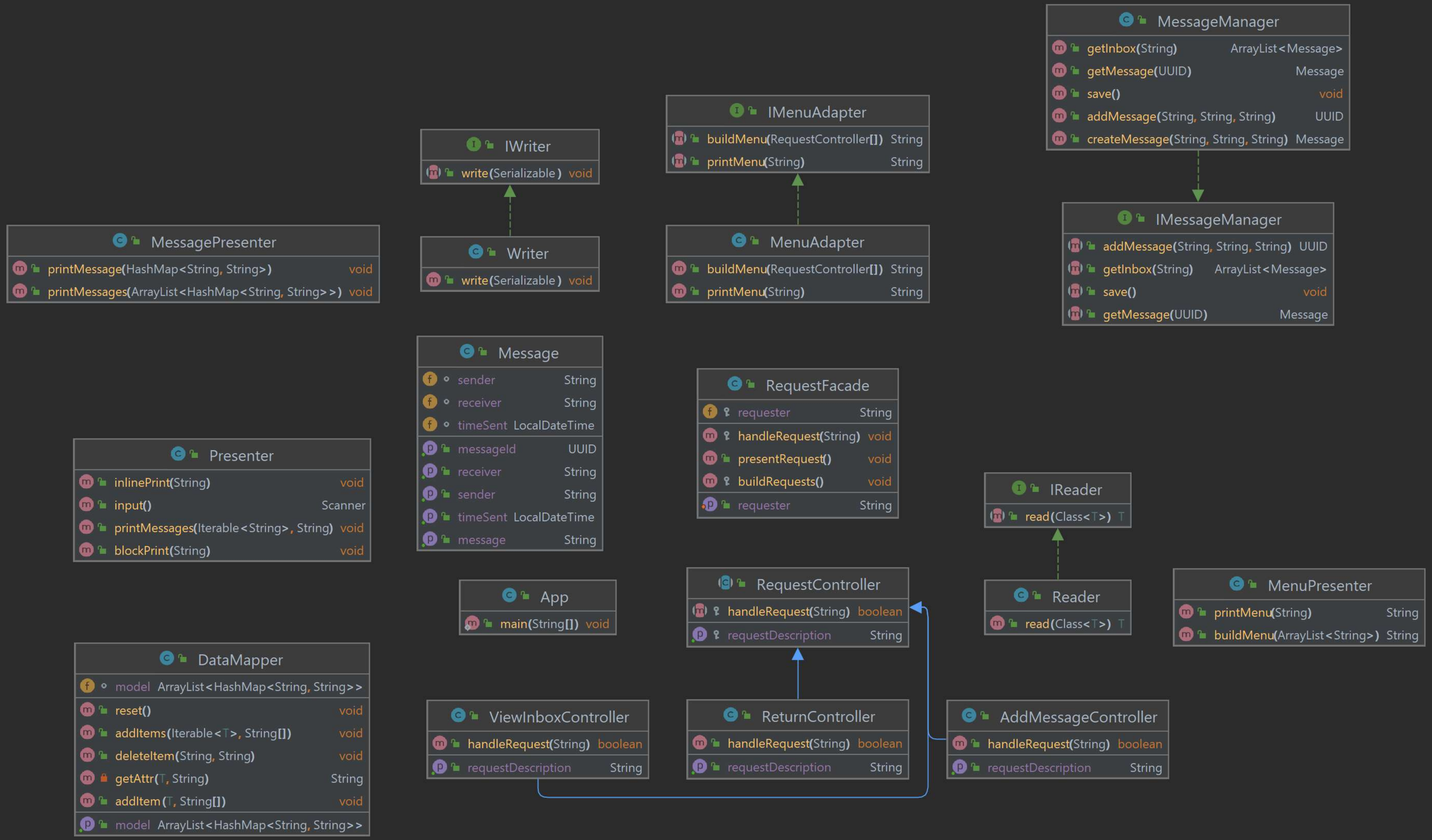
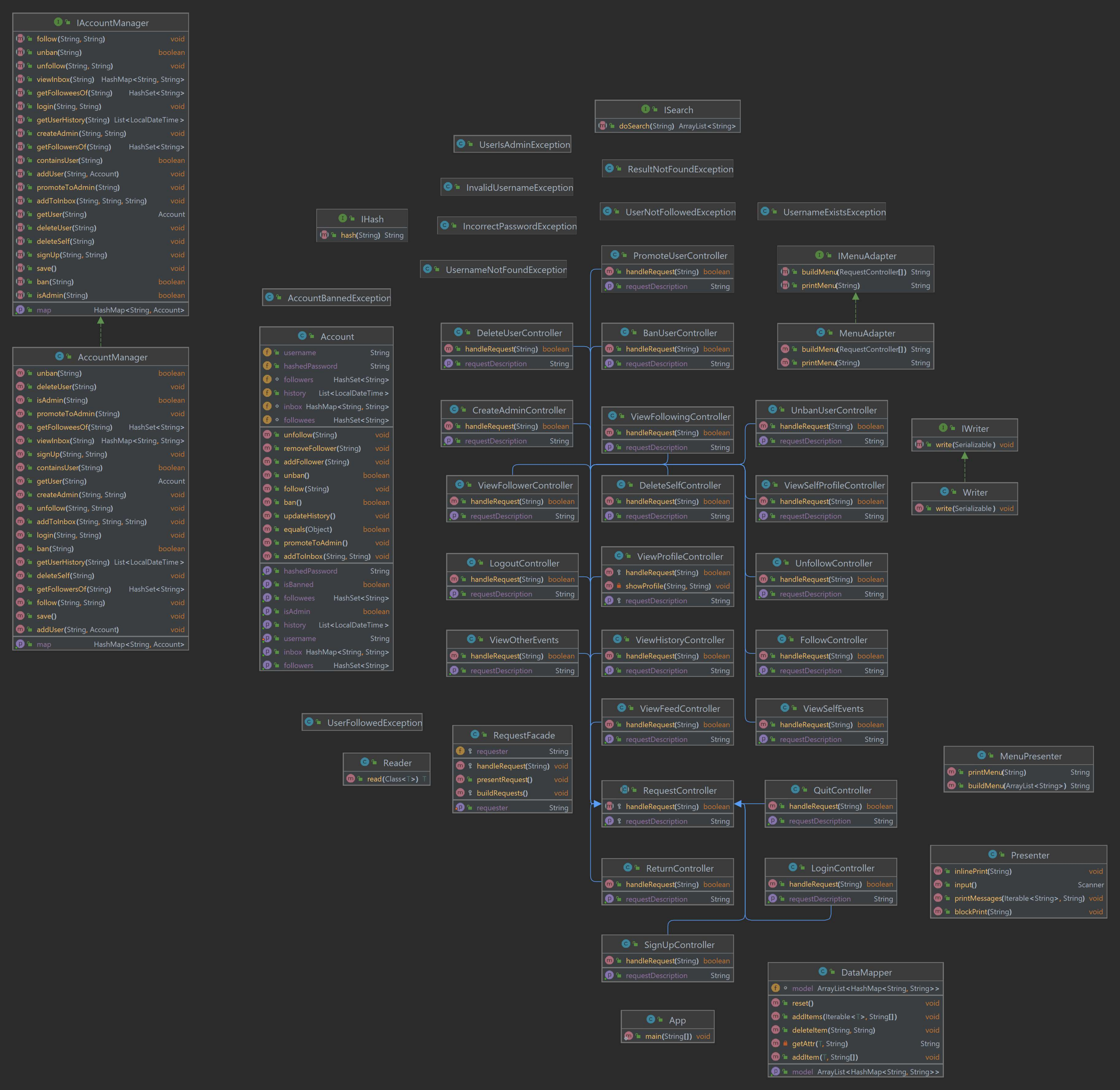
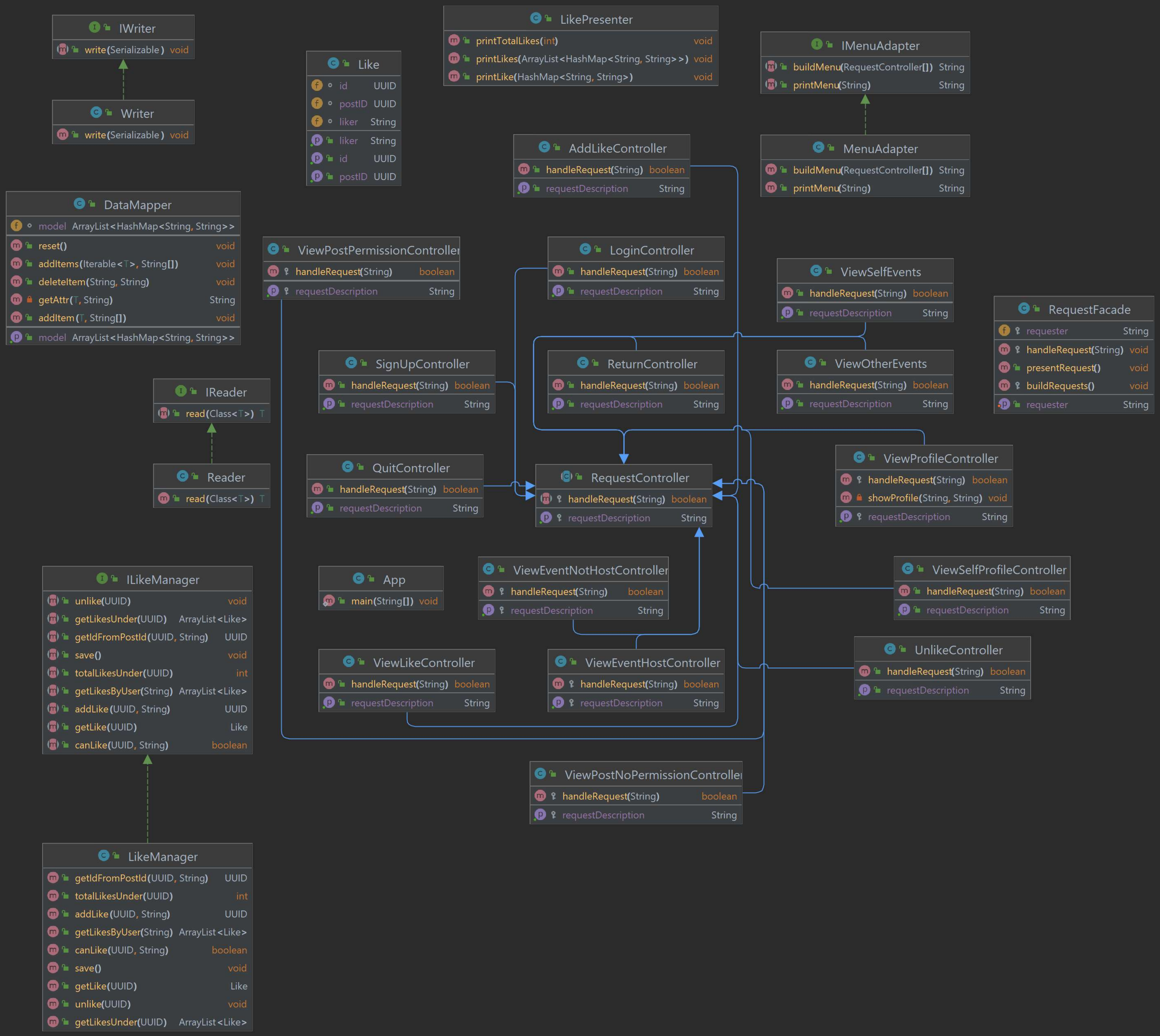
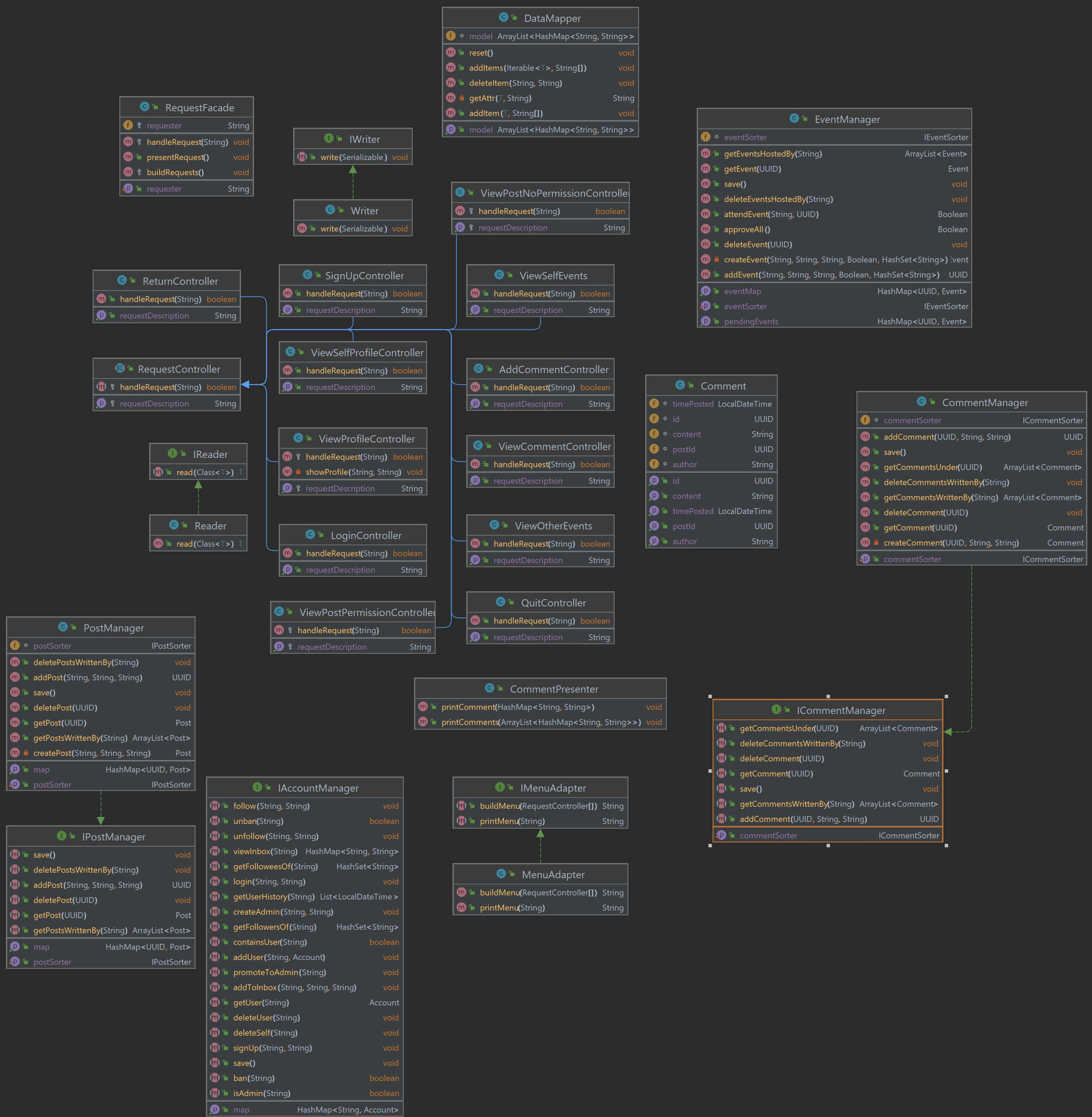
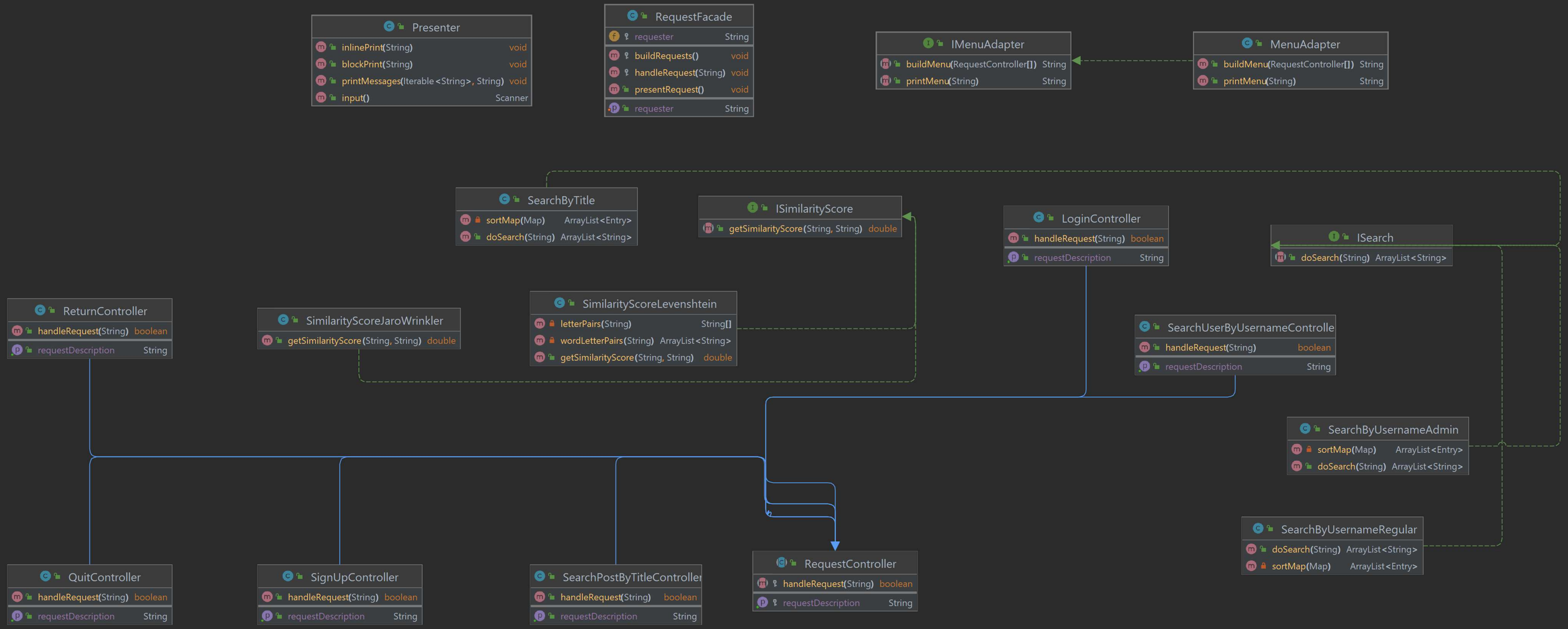

IPostManager
save() void
deletePostsWrittenBy(String) void
addPost(String, String, String) UUID
deletePost(UUID) void
getPost(UUID) Post
getPostsWrittenBy(String) ArrayList<Post>
map HashMap<UUID, Post>
postSorter IPostSorter
Post
content String
title String
id UUID
timePosted LocalDateTime
author String
id UUID
content String
title String
timePosted LocalDateTime
author String
IMenuAdapter
buildMenu(RequestController[]) String
printMenu(String) String
MenuAdapter
buildMenu(RequestController[]) String
printMenu(String) String
PostTimeComparator
compare(Post, Post) int
Reader
read(Class<T>) T
PostTimeSorter
sort(ArrayList<Post>) ArrayList<Post>
IReader
read(Class<T>) T
IPostSorter
sort(ArrayList<Post>) ArrayList<Post>
Writer
write(Serializable) void
IWriter
write(Serializable) void
PostManager
postSorter IPostSorter
deletePostsWrittenBy(String) void
addPost(String, String, String) UUID
save() void
deletePost(UUID) void
getPost(UUID) Post
getPostsWrittenBy(String) ArrayList<Post>
createPost(String, String, String) Post
map HashMap<UUID, Post>
postSorter IPostSorter
DataMapper
model ArrayList<HashMap<String, String>>
reset() void
addItems(Iterable<T>, String[]) void
deleteItem(String, String) void
getAttr(T, String) String
addItem(T, String[]) void
model ArrayList<HashMap<String, String>>
RequestFacade
requester String
handleRequest(String) void
presentRequest() void
buildRequests() void
requester String
App
main(String[]) void
RequestController
handleRequest(String) boolean
requestDescription String
AddPostController
handleRequest(String) boolean
requestDescription String
ViewPostPermissionController
handleRequest(String) boolean
requestDescription String
ReturnController
handleRequest(String) boolean
requestDescription String
ViewPostNoPermissionController
handleRequest(String) boolean
requestDescription String
DeletePostController
handleRequest(String) boolean
requestDescription String
PostPresenter
printPosts(ArrayList<HashMap<String, String>>) void
printPost(HashMap<String, String>) void
Presenter
inlinePrint(String) void
input() Scanner
printMessages(Iterable<String>, String) void
blockPrint(String) void

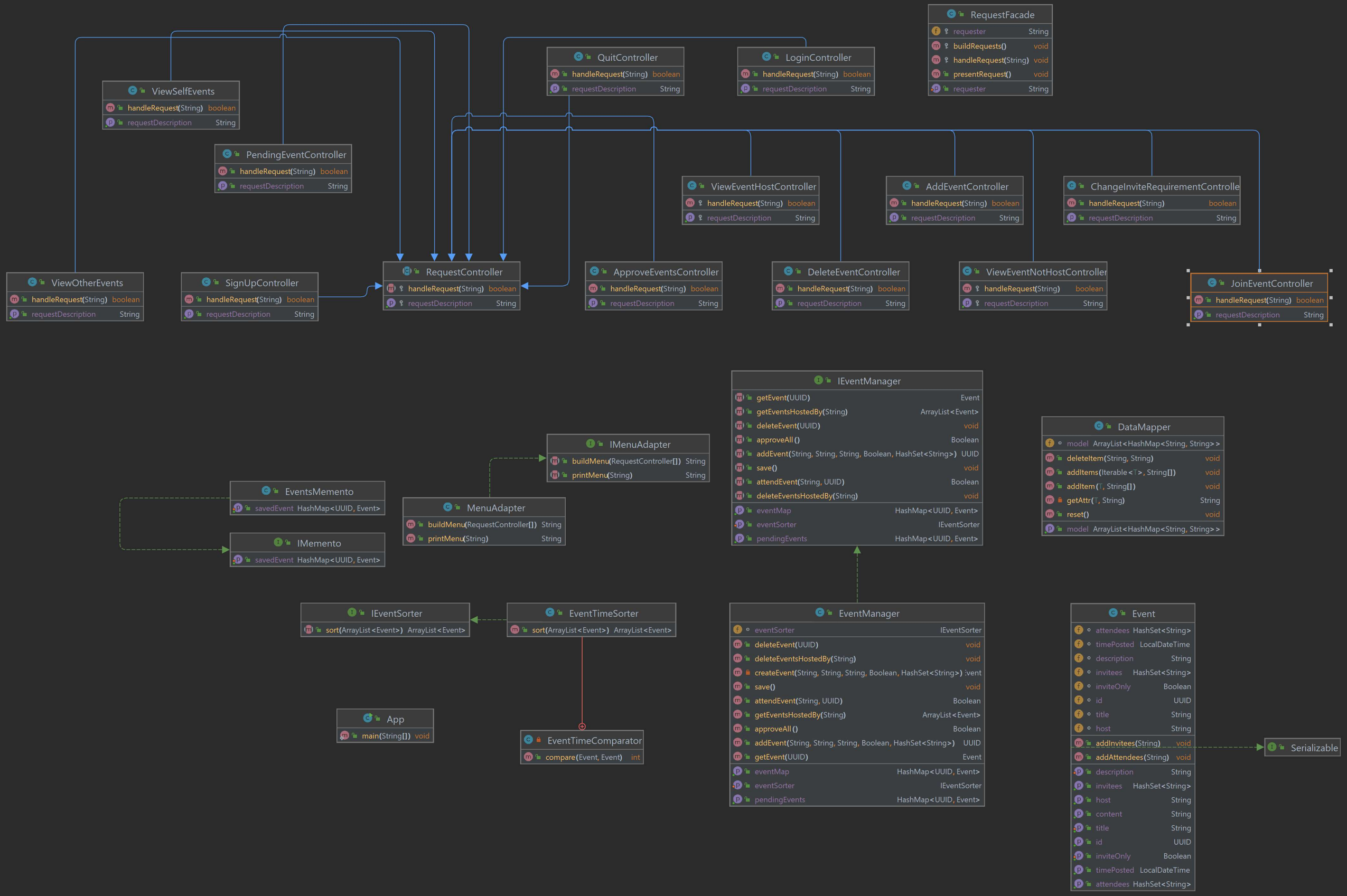

MessageManager
getInbox(String) ArrayList<Message>
getMessage(UUID) Message
save() void
addMessage(String, String, String) UUID
createMessage(String, String, String) Message
IMessageManager
addMessage(String, String, String) UUID
getInbox(String) ArrayList<Message>
save() void
getMessage(UUID) Message
IMenuAdapter
buildMenu(RequestController[]) String
printMenu(String) String
MenuAdapter
buildMenu(RequestController[]) String
printMenu(String) String
IWriter
write(Serializable) void
Writer
write(Serializable) void
MessagePresenter
printMessage(HashMap<String, String>) void
printMessages(ArrayList<HashMap<String, String>>) void
Message
sender String
receiver String
timeSent LocalDateTime
messageId UUID
receiver String
sender String
timeSent LocalDateTime
message String
RequestFacade
requester String
handleRequest(String) void
presentRequest() void
buildRequests() void
requester String
Presenter
inlinePrint(String) void
input() Scanner
printMessages(Iterable<String>, String) void
blockPrint(String) void
IReader
read(Class<T>) T
Reader
read(Class<T>) T
App
main(String[]) void
RequestController
handleRequest(String) boolean
requestDescription String
MenuPresenter
printMenu(String) String
buildMenu(ArrayList<String>) String
DataMapper
model ArrayList<HashMap<String, String>>
reset() void
addItems(Iterable<T>, String[]) void
deleteItem(String, String) void
getAttr(T, String) String
addItem(T, String[]) void
model ArrayList<HashMap<String, String>>
ViewInboxController
handleRequest(String) boolean
requestDescription String
ReturnController
handleRequest(String) boolean
requestDescription String
AddMessageController
handleRequest(String) boolean
requestDescription String


IAccountManager
follow(String, String) void
unban(String) boolean
unfollow(String, String) void
viewInbox(String) HashMap<String, String>
getFolloweesOf(String) HashSet<String>
login(String, String) void
getUserHistory(String) List<LocalDateTime>
createAdmin(String, String) void
getFollowersOf(String) HashSet<String>
containsUser(String) boolean
addUser(String, Account) void
promoteToAdmin(String) void
addToInbox(String, String, String) void
getUser(String) Account
deleteUser(String) void
deleteSelf(String) void
signUp(String, String) void
save() void
ban(String) boolean
isAdmin(String) boolean
map HashMap<String, Account>
AccountManager
unban(String) boolean
deleteUser(String) void
isAdmin(String) boolean
promoteToAdmin(String) void
getFolloweesOf(String) HashSet<String>
viewInbox(String) HashMap<String, String>
signUp(String, String) void
containsUser(String) boolean
getUser(String) Account
createAdmin(String, String) void
unfollow(String, String) void
addToInbox(String, String, String) void
login(String, String) void
ban(String) boolean
getUserHistory(String) List<LocalDateTime>
deleteSelf(String) void
getFollowersOf(String) HashSet<String>
follow(String, String) void
save() void
addUser(String, Account) void
map HashMap<String, Account>
ISearch
doSearch(String) ArrayList<String>
UserIsAdminException
ResultNotFoundException
InvalidUsernameException
UserNotFollowedException
UsernameExistsException
IHash
hash(String) String
IncorrectPasswordException
UsernameNotFoundException
PromoteUserController
handleRequest(String) boolean
requestDescription String
IMenuAdapter
buildMenu(RequestController[]) String
printMenu(String) String
AccountBannedException
Account
username String
hashedPassword String
followers HashSet<String>
history List<LocalDateTime>
inbox HashMap<String, String>
followees HashSet<String>
unfollow(String) void
removeFollower(String) void
addFollower(String) void
unban() boolean
follow(String) void
ban() boolean
updateHistory() void
equals(Object) boolean
promoteToAdmin() void
addToInbox(String, String) void
hashedPassword String
isBanned boolean
followees HashSet<String>
isAdmin boolean
history List<LocalDateTime>
username String
inbox HashMap<String, String>
followers HashSet<String>
DeleteUserController
handleRequest(String) boolean
requestDescription String
BanUserController
handleRequest(String) boolean
requestDescription String
MenuAdapter
buildMenu(RequestController[]) String
printMenu(String) String
CreateAdminController
handleRequest(String) boolean
requestDescription String
ViewFollowingController
handleRequest(String) boolean
requestDescription String
UnbanUserController
handleRequest(String) boolean
requestDescription String
IWriter
write(Serializable) void
ViewFollowerController
handleRequest(String) boolean
requestDescription String
DeleteSelfController
handleRequest(String) boolean
requestDescription String
ViewSelfProfileController
handleRequest(String) boolean
requestDescription String
Writer
write(Serializable) void
LogoutController
handleRequest(String) boolean
requestDescription String
ViewProfileController
handleRequest(String) boolean
showProfile(String, String) void
requestDescription String
UnfollowController
handleRequest(String) boolean
requestDescription String
ViewOtherEvents
handleRequest(String) boolean
requestDescription String
ViewHistoryController
handleRequest(String) boolean
requestDescription String
FollowController
handleRequest(String) boolean
requestDescription String
ViewFeedController
handleRequest(String) boolean
requestDescription String
ViewSelfEvents
handleRequest(String) boolean
requestDescription String
UserFollowedException
RequestFacade
requester String
handleRequest(String) void
presentRequest() void
buildRequests() void
requester String
Reader
read(Class<T>) T
MenuPresenter
printMenu(String) String
buildMenu(ArrayList<String>) String
RequestController
handleRequest(String) boolean
requestDescription String
QuitController
handleRequest(String) boolean
requestDescription String
Presenter
inlinePrint(String) void
input() Scanner
printMessages(Iterable<String>, String) void
blockPrint(String) void
ReturnController
handleRequest(String) boolean
requestDescription String
LoginController
handleRequest(String) boolean
requestDescription String
SignUpController
handleRequest(String) boolean
requestDescription String
DataMapper
model ArrayList<HashMap<String, String>>
reset() void
addItems(Iterable<T>, String[]) void
deleteItem(String, String) void
getAttr(T, String) String
addItem(T, String[]) void
model ArrayList<HashMap<String, String>>
App
main(String[]) void


IWriter
write(Serializable) void
Writer
write(Serializable) void
Like
id UUID
postID UUID
liker String
liker String
id UUID
postID UUID
LikePresenter
printTotalLikes(int) void
printLikes(ArrayList<HashMap<String, String>>) void
printLike(HashMap<String, String>) void
IMenuAdapter
buildMenu(RequestController[]) String
printMenu(String) String
MenuAdapter
buildMenu(RequestController[]) String
printMenu(String) String
AddLikeController
handleRequest(String) boolean
requestDescription String
DataMapper
model ArrayList<HashMap<String, String>>
reset() void
addItems(Iterable<T>, String[]) void
deleteItem(String, String) void
getAttr(T, String) String
addItem(T, String[]) void
model ArrayList<HashMap<String, String>>
ViewPostPermissionController
handleRequest(String) boolean
requestDescription String
LoginController
handleRequest(String) boolean
requestDescription String
ViewSelfEvents
handleRequest(String) boolean
requestDescription String
RequestFacade
requester String
handleRequest(String) void
presentRequest() void
buildRequests() void
requester String
SignUpController
handleRequest(String) boolean
requestDescription String
ReturnController
handleRequest(String) boolean
requestDescription String
ViewOtherEvents
handleRequest(String) boolean
requestDescription String
IReader
read(Class<T>) T
Reader
read(Class<T>) T
QuitController
handleRequest(String) boolean
requestDescription String
RequestController
handleRequest(String) boolean
requestDescription String
ViewProfileController
handleRequest(String) boolean
showProfile(String, String) void
requestDescription String
App
main(String[]) void
ViewEventNotHostController
handleRequest(String) boolean
requestDescription String
ViewSelfProfileController
handleRequest(String) boolean
requestDescription String
ILikeManager
unlike(UUID) void
getLikesUnder(UUID) ArrayList<Like>
getIdFromPostId(UUID, String) UUID
save() void
totalLikesUnder(UUID) int
getLikesByUser(String) ArrayList<Like>
addLike(UUID, String) UUID
getLike(UUID) Like
canLike(UUID, String) boolean
ViewLikeController
handleRequest(String) boolean
requestDescription String
ViewEventHostController
handleRequest(String) boolean
requestDescription String
UnlikeController
handleRequest(String) boolean
requestDescription String
ViewPostNoPermissionController
handleRequest(String) boolean
requestDescription String
LikeManager
getIdFromPostId(UUID, String) UUID
totalLikesUnder(UUID) int
addLike(UUID, String) UUID
getLikesByUser(String) ArrayList<Like>
canLike(UUID, String) boolean
save() void
getLike(UUID) Like
unlike(UUID) void
getLikesUnder(UUID) ArrayList<Like>


DataMapper
model ArrayList<HashMap<String, String>>
reset() void
addItems(Iterable<T>, String[]) void
deleteItem(String, String) void
getAttr(T, String) String
addItem(T, String[]) void
model ArrayList<HashMap<String, String>>
RequestFacade
requester String
handleRequest(String) void
presentRequest() void
buildRequests() void
requester String
IWriter
write(Serializable) void
Writer
write(Serializable) void
ViewPostNoPermissionControlle
handleRequest(String) boolean
requestDescription String
EventManager
eventSorter IEventSorter
getEventsHostedBy(String) ArrayList<Event>
getEvent(UUID) Event
save() void
deleteEventsHostedBy(String) void
attendEvent(String, UUID) Boolean
approveAll() Boolean
deleteEvent(UUID) void
createEvent(String, String, String, Boolean, HashSet<String>) Event
addEvent(String, String, String, Boolean, HashSet<String>) UUID
eventMap HashMap<UUID, Event>
eventSorter IEventSorter
pendingEvents HashMap<UUID, Event>
ReturnController
handleRequest(String) boolean
requestDescription String
SignUpController
handleRequest(String) boolean
requestDescription String
ViewSelfEvents
handleRequest(String) boolean
requestDescription String
ViewSelfProfileController
handleRequest(String) boolean
requestDescription String
RequestController
handleRequest(String) boolean
requestDescription String
AddCommentController
handleRequest(String) boolean
requestDescription String
Comment
timePosted LocalDateTime
id UUID
content String
postId UUID
author String
id UUID
content String
timePosted LocalDateTime
postId UUID
author String
CommentManager
commentSorter ICommentSorter
addComment(UUID, String, String) UUID
save() void
getCommentsUnder(UUID) ArrayList<Comment>
deleteCommentsWrittenBy(String) void
getCommentsWrittenBy(String) ArrayList<Comment>
deleteComment(UUID) void
getComment(UUID) Comment
createComment(UUID, String, String) Comment
commentSorter ICommentSorter
ViewProfileController
handleRequest(String) boolean
showProfile(String, String) void
requestDescription String
IReader
read(Class<T>) T
Reader
read(Class<T>) T
ViewCommentController
handleRequest(String) boolean
requestDescription String
LoginController
handleRequest(String) boolean
requestDescription String
ViewOtherEvents
handleRequest(String) boolean
requestDescription String
ViewPostPermissionControlle
handleRequest(String) boolean
requestDescription String
QuitController
handleRequest(String) boolean
requestDescription String
PostManager
postSorter IPostSorter
deletePostsWrittenBy(String) void
addPost(String, String, String) UUID
save() void
deletePost(UUID) void
getPost(UUID) Post
getPostsWrittenBy(String) ArrayList<Post>
createPost(String, String, String) Post
map HashMap<UUID, Post>
postSorter IPostSorter
CommentPresenter
printComment(HashMap<String, String>) void
printComments(ArrayList<HashMap<String, String>>) void
ICommentManager
getCommentsUnder(UUID) ArrayList<Comment>
deleteCommentsWrittenBy(String) void
deleteComment(UUID) void
getComment(UUID) Comment
save() void
getCommentsWrittenBy(String) ArrayList<Comment>
addComment(UUID, String, String) UUID
commentSorter ICommentSorter
IPostManager
save() void
deletePostsWrittenBy(String) void
addPost(String, String, String) UUID
deletePost(UUID) void
getPost(UUID) Post
getPostsWrittenBy(String) ArrayList<Post>
map HashMap<UUID, Post>
postSorter IPostSorter
IAccountManager
follow(String, String) void
unban(String) boolean
unfollow(String, String) void
viewInbox(String) HashMap<String, String>
getFolloweesOf(String) HashSet<String>
login(String, String) void
getUserHistory(String) List<LocalDateTime>
createAdmin(String, String) void
getFollowersOf(String) HashSet<String>
containsUser(String) boolean
addUser(String, Account) void
promoteToAdmin(String) void
addToInbox(String, String, String) void
getUser(String) Account
deleteUser(String) void
deleteSelf(String) void
signUp(String, String) void
save() void
ban(String) boolean
isAdmin(String) boolean
map HashMap<String, Account>
IMenuAdapter
buildMenu(RequestController[]) String
printMenu(String) String
MenuAdapter
buildMenu(RequestController[]) String
printMenu(String) String


Presenter
inlinePrint(String) void
blockPrint(String) void
printMessages(Iterable<String>, String) void
input() Scanner
RequestFacade
requester String
buildRequests() void
handleRequest(String) void
presentRequest() void
requester String
IMenuAdapter
buildMenu(RequestController[]) String
printMenu(String) String
MenuAdapter
buildMenu(RequestController[]) String
printMenu(String) String
SearchByTitle
sortMap(Map) ArrayList<Entry>
doSearch(String) ArrayList<String>
ISimilarityScore
getSimilarityScore(String, String) double
LoginController
handleRequest(String) boolean
requestDescription String
ISearch
doSearch(String) ArrayList<String>
ReturnController
handleRequest(String) boolean
requestDescription String
SimilarityScoreJaroWrinkler
getSimilarityScore(String, String) double
SimilarityScoreLevenshtein
letterPairs(String) String[]
wordLetterPairs(String) ArrayList<String>
getSimilarityScore(String, String) double
SearchUserByUsernameControlle
handleRequest(String) boolean
requestDescription String
SearchByUsernameAdmin
sortMap(Map) ArrayList<Entry>
doSearch(String) ArrayList<String>
SearchByUsernameRegular
doSearch(String) ArrayList<String>
sortMap(Map) ArrayList<Entry>
QuitController
handleRequest(String) boolean
requestDescription String
SignUpController
handleRequest(String) boolean
requestDescription String
SearchPostByTitleController
handleRequest(String) boolean
requestDescription String
RequestController
handleRequest(String) boolean
requestDescription String


RequestFacade
requester String
buildRequests() void
handleRequest(String) void
presentRequest() void
requester String
QuitController
handleRequest(String) boolean
requestDescription String
LoginController
handleRequest(String) boolean
requestDescription String
ViewSelfEvents
handleRequest(String) boolean
requestDescription String
PendingEventController
handleRequest(String) boolean
requestDescription String
ViewEventHostController
handleRequest(String) boolean
requestDescription String
AddEventController
handleRequest(String) boolean
requestDescription String
ChangeInviteRequirementControlle
handleRequest(String) boolean
requestDescription String
ViewOtherEvents
handleRequest(String) boolean
requestDescription String
SignUpController
handleRequest(String) boolean
requestDescription String
RequestController
handleRequest(String) boolean
requestDescription String
ApproveEventsController
handleRequest(String) boolean
requestDescription String
DeleteEventController
handleRequest(String) boolean
requestDescription String
ViewEventNotHostController
handleRequest(String) boolean
requestDescription String
JoinEventController
handleRequest(String) boolean
requestDescription String
IEventManager
getEvent(UUID) Event
getEventsHostedBy(String) ArrayList<Event>
deleteEvent(UUID) void
approveAll () Boolean
addEvent(String, String, String, Boolean, HashSet<String>) UUID
save() void
attendEvent(String, UUID) Boolean
deleteEventsHostedBy(String) void
eventMap HashMap<UUID, Event>
eventSorter IEventSorter
pendingEvents HashMap<UUID, Event>
DataMapper
model ArrayList<HashMap<String, String>>
deleteItem(String, String) void
addItems(Iterable<T>, String[]) void
addItem(T, String[]) void
getAttr(T, String) String
reset() void
model ArrayList<HashMap<String, String>>
IMenuAdapter
buildMenu(RequestController[]) String
printMenu(String) String
EventsMemento
savedEvent HashMap<UUID, Event>
MenuAdapter
buildMenu(RequestController[]) String
printMenu(String) String
IMemento
savedEvent HashMap<UUID, Event>
IEventSorter
sort(ArrayList<Event>) ArrayList<Event>
EventTimeSorter
sort(ArrayList<Event>) ArrayList<Event>
EventManager
eventSorter IEventSorter
deleteEvent(UUID) void
deleteEventsHostedBy(String) void
createEvent(String, String, String, Boolean, HashSet<String>) :vent
save() void
attendEvent(String, UUID) Boolean
getEventsHostedBy(String) ArrayList<Event>
approveAll () Boolean
addEvent(String, String, String, Boolean, HashSet<String>) UUID
getEvent(UUID) Event
eventMap HashMap<UUID, Event>
eventSorter IEventSorter
pendingEvents HashMap<UUID, Event>
Event
attendees HashSet<String>
timePosted LocalDateTime
description String
invitees HashSet<String>
inviteOnly Boolean
id UUID
title String
host String
addInvitees(String) void
addAttendees(String) void
description String
invitees HashSet<String>
host String
content String
title String
id UUID
inviteOnly Boolean
timePosted LocalDateTime
attendees HashSet<String>
Serializable
App
main(String[]) void
EventTimeComparator
compare(Event, Event) int