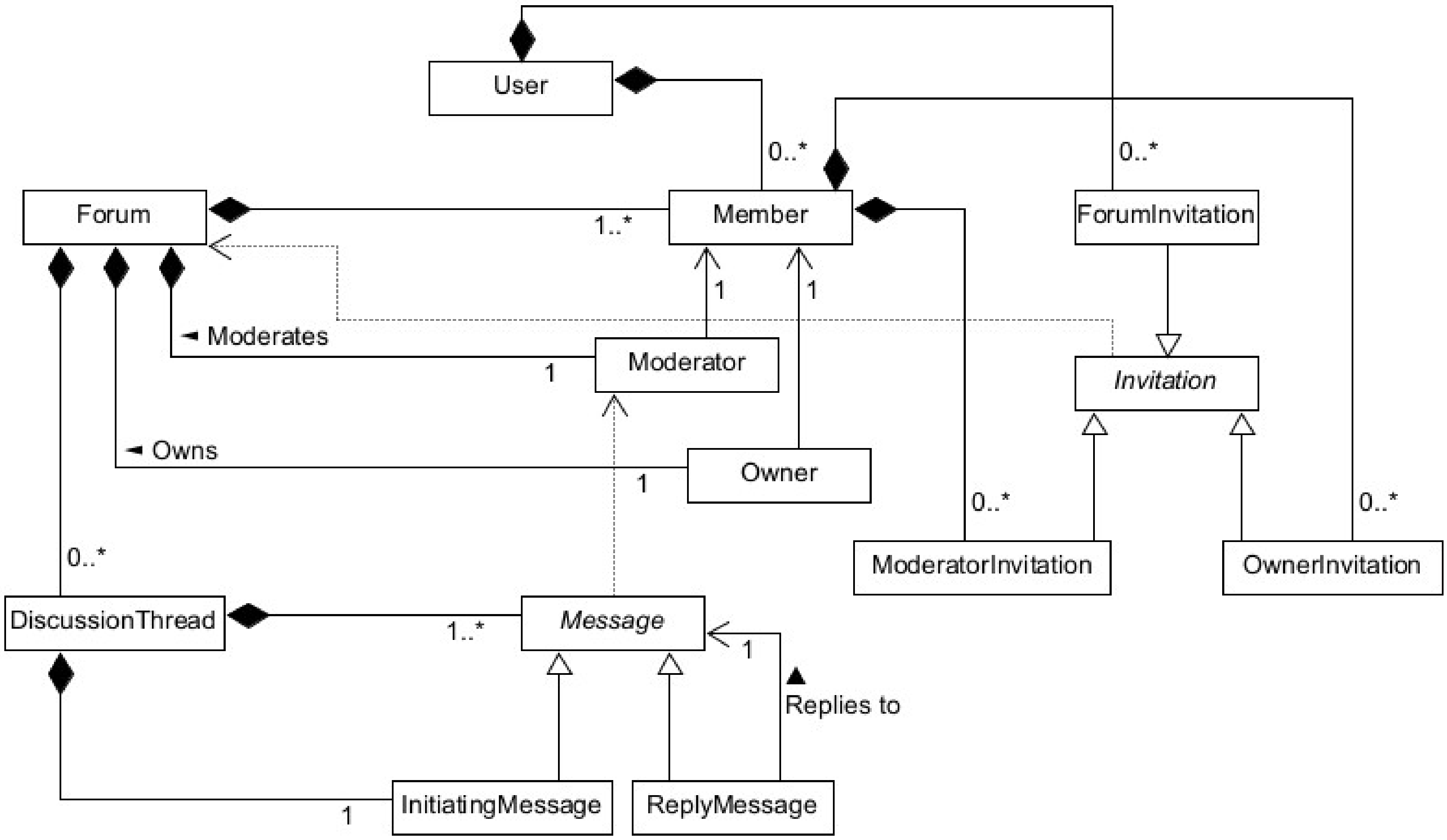
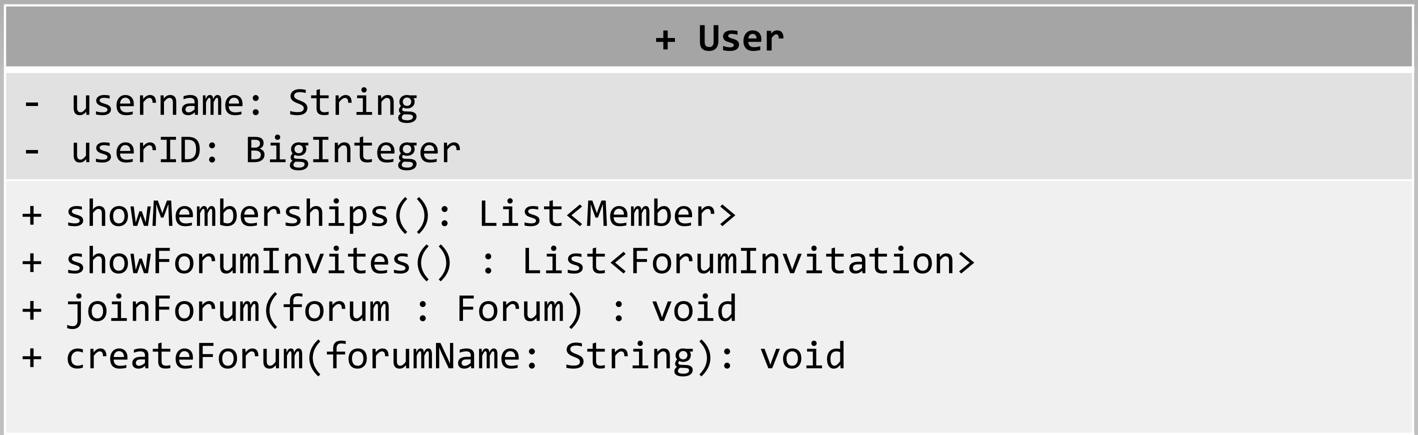
Assignment 4

UML for Online Discussion Forum

By Paul Kotys and Max Humes


User
0..*
0..*
Forum
1..*
Member
ForumInvitation
1
1
◄ Moderates
Moderator
1
Invitation
◄ Owns
Owner
1
0..*
0..*
0..*
ModeratorInvitation
OwnerInvitation
DiscussionThread
1..*
Message
1
▲
Replies to
1
InitiatingMessage
ReplyMessage

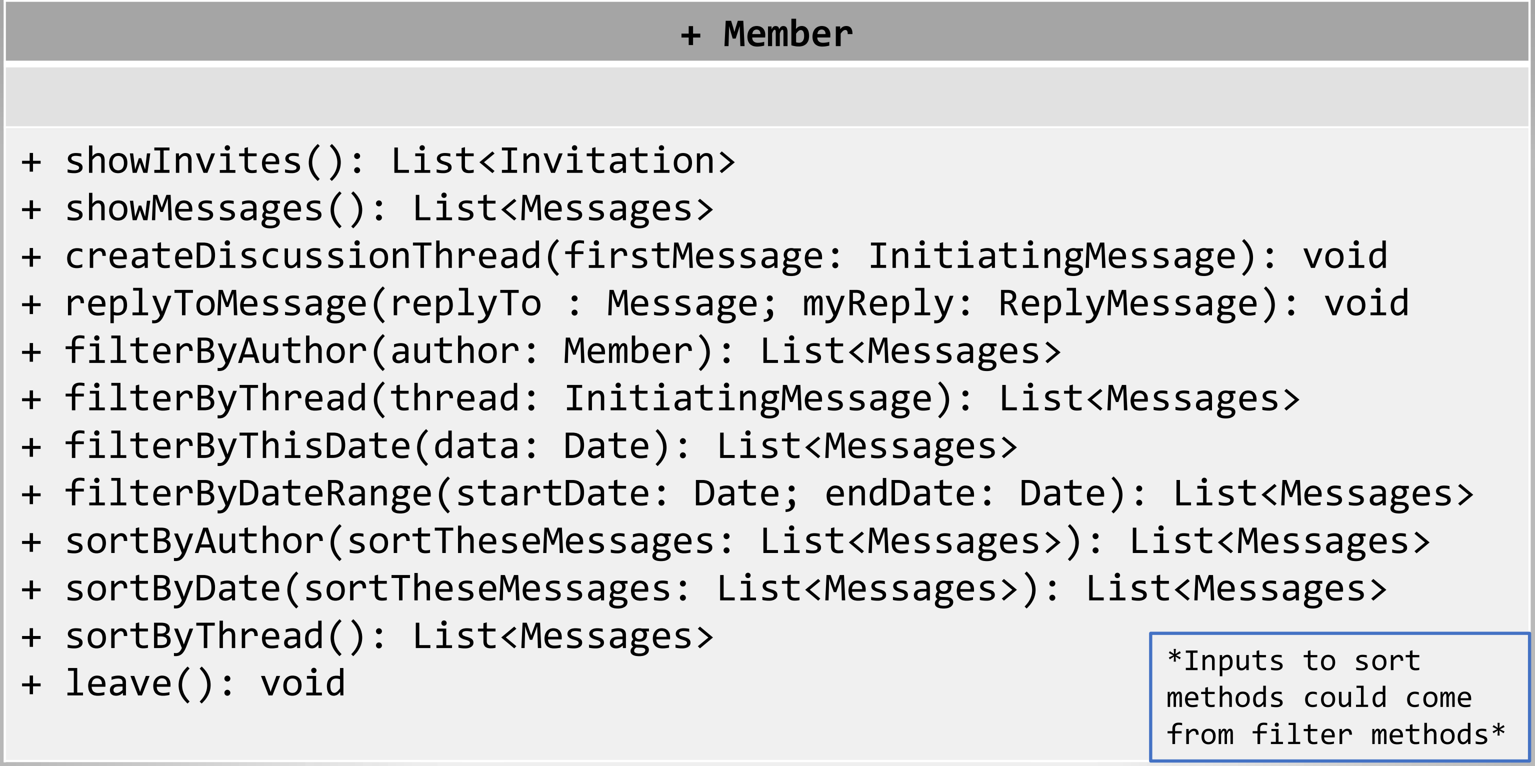


+ User

```
-  username: String
-  userID: BigInteger
```

```
+ showMemberships(): List<Member>
+ showForumInvites() : List<ForumInvitation>
+ joinForum(forum : Forum) : void
+ createForum(forumName: String): void
```

Class	**Data Structure**
User	List<Member> memberOf;
User	List<ForumInvitation> forumInvites;

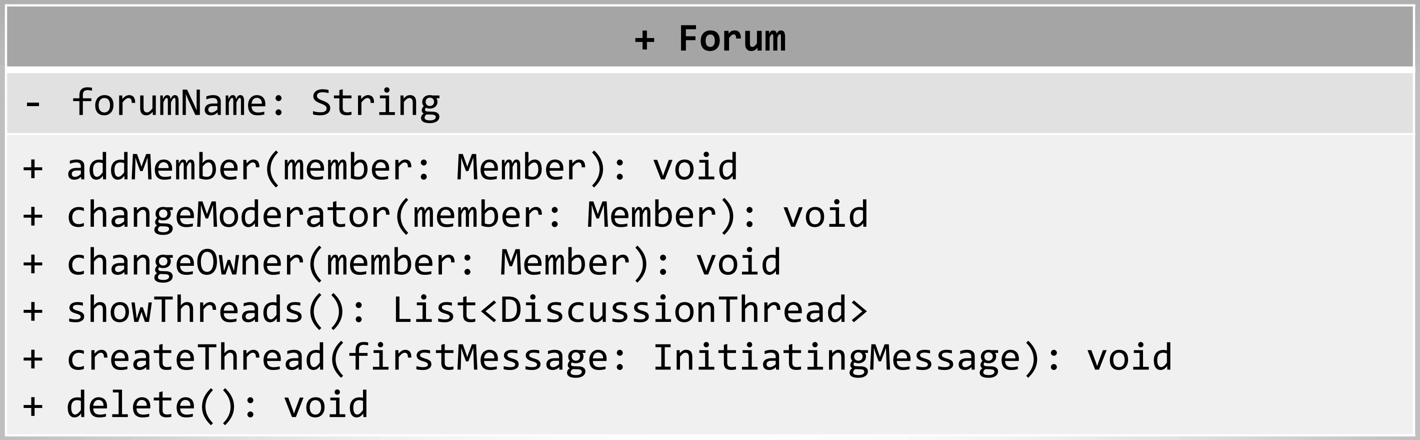


```
+ Member

+ showInvites(): List<Invitation>
+ showMessages(): List<Messages>
+ createDiscussionThread(firstMessage: InitiatingMessage): void
+ replyToMessage(replyTo : Message; myReply: ReplyMessage): void
+ filterByAuthor(author: Member): List<Messages>
+ filterByThread(thread: InitiatingMessage): List<Messages>
+ filterByThisDate(data: Date): List<Messages>
+ filterByDateRange(startDate: Date; endDate: Date): List<Messages>
+ sortByAuthor(sortTheseMessages: List<Messages>): List<Messages>
+ sortByDate(sortTheseMessages: List<Messages>): List<Messages>
+ sortByThread(): List<Messages>
+ leave(): void
```

Inputs to sort methods could come from filter methods

Class	Data Structure
Member	User user;
Member	Forum forum;
Member	List<Message> myMessages;
Member	List<Invitation> invites;

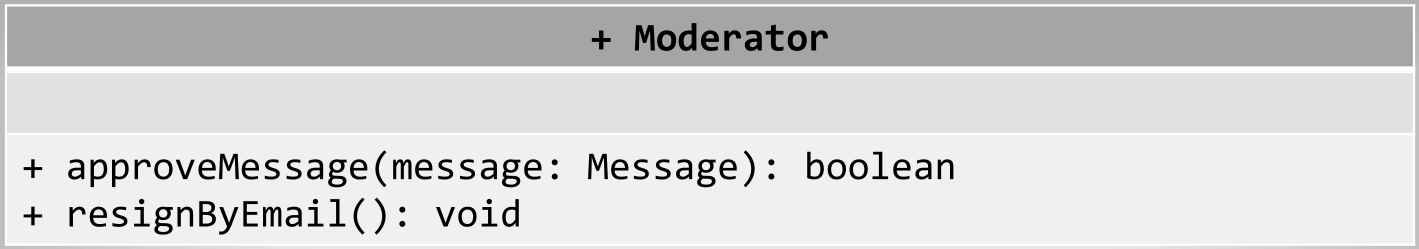


+ Forum

- forumName: String

```
+ addMember(member: Member): void
+ changeModerator(member: Member): void
+ changeOwner(member: Member): void
+ showThreads(): List<DiscussionThread>
+ createThread(firstMessage: InitiatingMessage): void
+ delete(): void
```

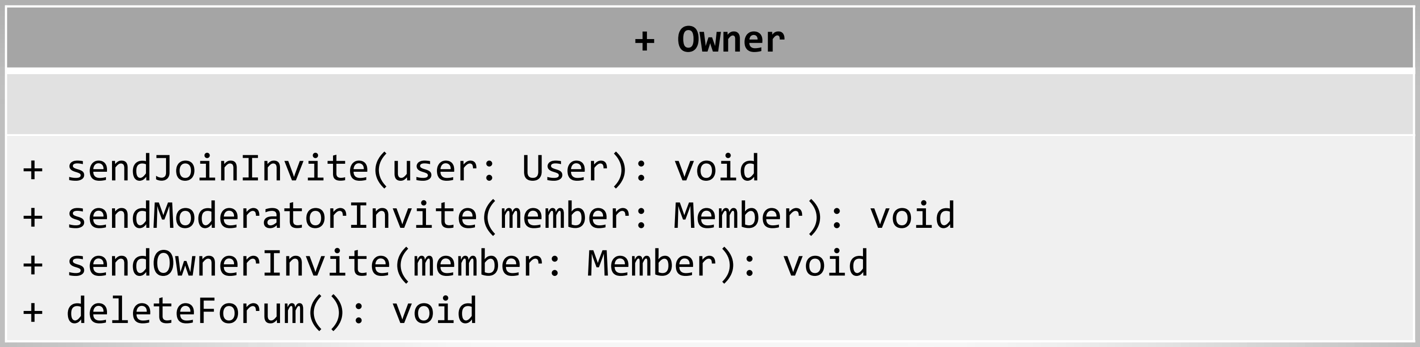
Class	Data Structure
Forum	Owner owner;
Forum	Moderator moderator;
Forum	List<Member> members;
Forum	List<DiscussionThread> threads;



+ Moderator

+ approveMessage(message: Message): boolean
+ resignByEmail(): void

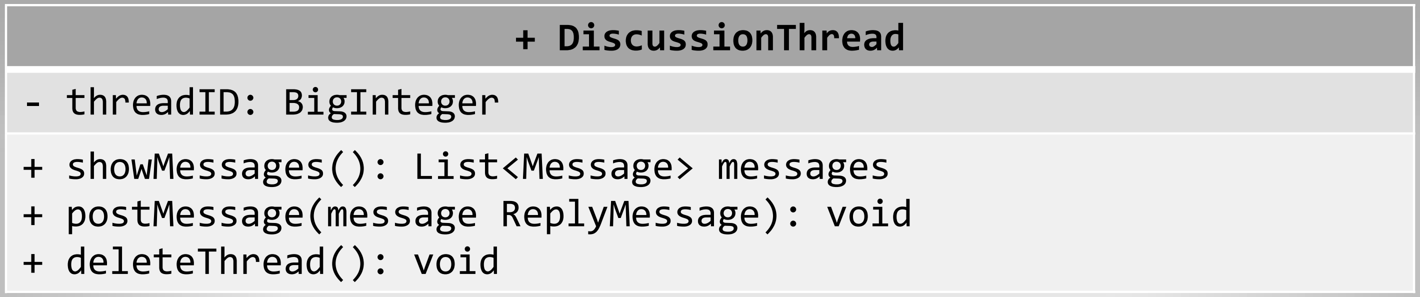
Class	Data Structure
Moderator	Member member;
Moderator	Forum forum;



+ Owner

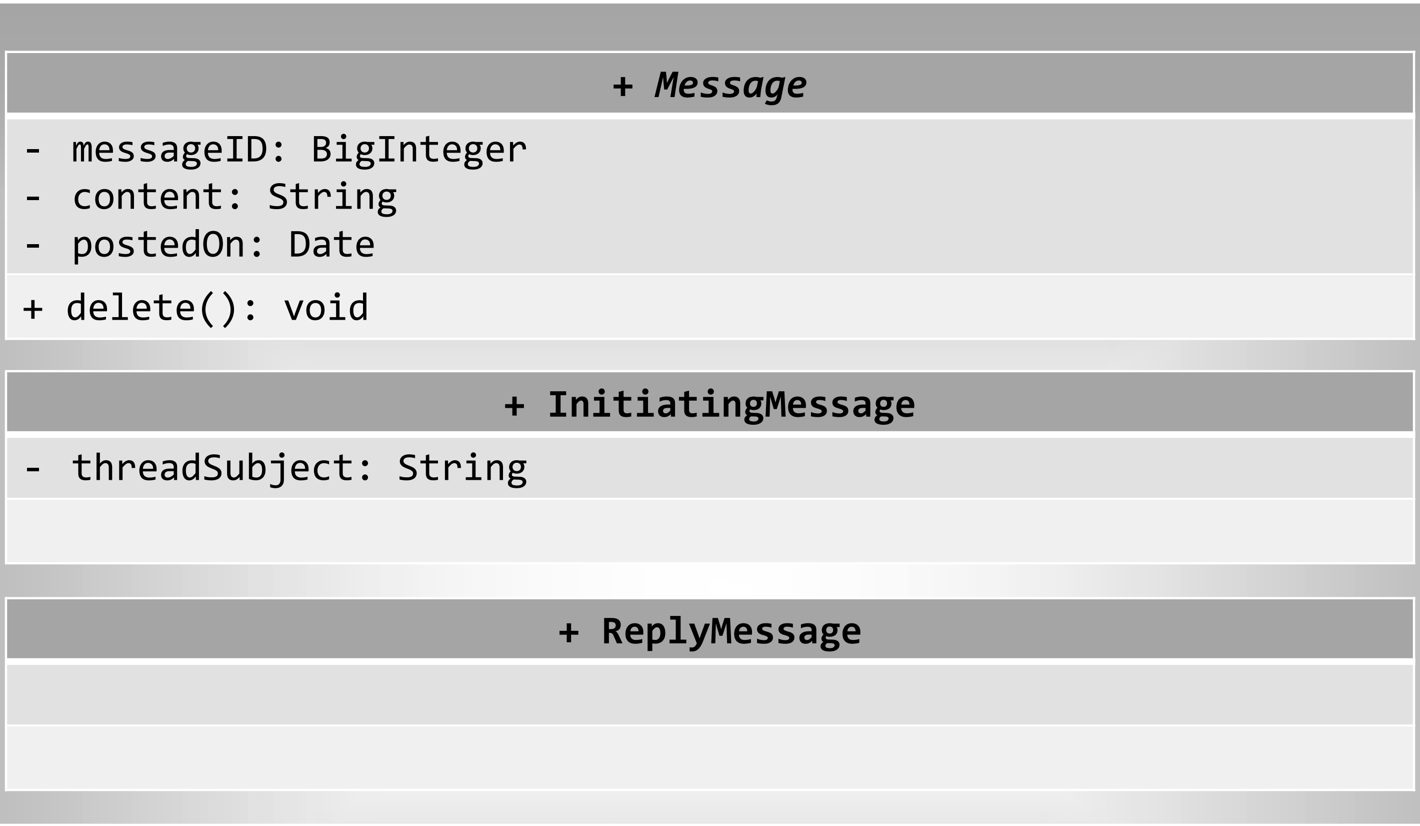
+ sendJoinInvite(user: User): void
+ sendModeratorInvite(member: Member): void
+ sendOwnerInvite(member: Member): void
+ deleteForum(): void

Class	Data Structure
Owner	Member member;
Owner	Forum forum;



```
+ DiscussionThread
- threadID: BigInteger
+ showMessages(): List<Message> messages
+ postMessage(message ReplyMessage): void
+ deleteThread(): void
```

Class	Data Structure
DiscussionThread	InitiatingMessage initiatingMessage;
DiscussionThread	List<Message> messages;



+ *Message*

- messageID: BigInteger
- content: String
- postedOn: Date

+ delete(): void

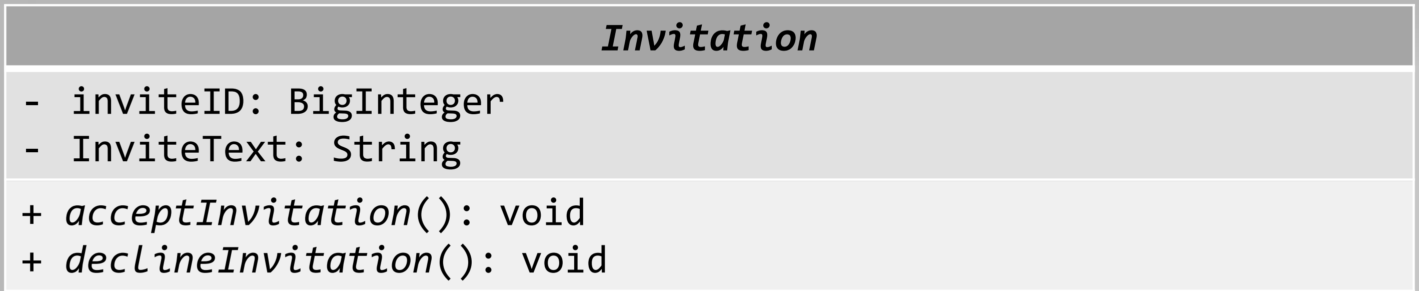
+ InitiatingMessage

- threadSubject: String

+ ReplyMessage

Reply message stores reference to the message that it replied to.

Class	**Data Structure**
Message	Member messageAuthor;
Message	DiscussionThread parentThread;
ReplyMessage	Message replyTo;

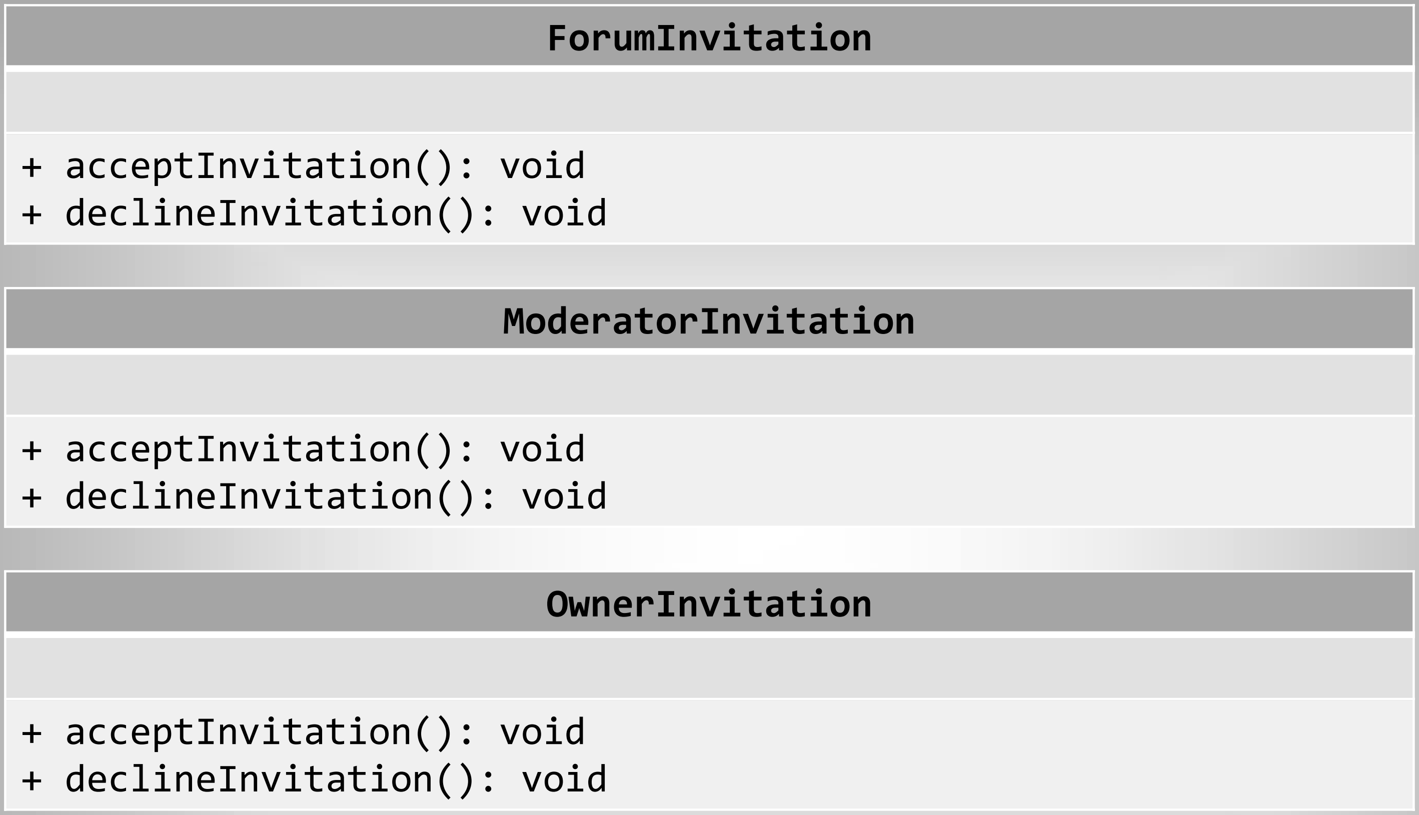


Invitation

- inviteID: BigInteger
- InviteText: String

+ *acceptInvitation*(): void
+ *declineInvitation*(): void

Class	**Data Structure**
Invitation	Forum forum;



```
ForumInvitation

+ acceptInvitation(): void
+ declineInvitation(): void
```

```
ModeratorInvitation

+ acceptInvitation(): void
+ declineInvitation(): void
```

```
OwnerInvitation

+ acceptInvitation(): void
+ declineInvitation(): void
```

Class	Data Structure
ForumInvitation	User user;
ModeratorInvitation	Member member;
OwnerInvitation	Member member;