

sip-webservice 1.0 API

Packages

Package	Description
com.realet.sip	
com.realet.sip.GsonTypeAdapter	

Package com.realet.sip

Class Summary

Class	Description
AuthResource	RestEasy Resource-Klasse, die für Authentifizierung zuständig ist.
Chat	Die Chat Klasse ist ein Behälter für ChatMessages .
ChatMessage	Diese Klasse repräsentiert eine einzelne ChatMessage .
ChatMessagesFacade	Manager-Klasse für ChatMessages
ChatMessagesResource	RestEasy Resource-Klasse für ChatMessages .
ChatsFacade	Manager-Klasse für Chats
ChatsResource	RestEasy Resource-Klasse für Chats .
ChatWebSocketManagement	Manager-Klasse für Session-Objekte, die für einen Chat registriert sind.
ChatWebSocketResource	Diese Klasse akzeptiert WebSockets-Verbindungen nach "/chats/{chatId}/websockets".
CORSPreflightHandler	Beantworte all OPTIONS-Anfragen mit den korrekten HTTP-Header und einer 200 OK Antwort, um CORS zu erlauben.
FileUploadForm	Behälter-Klasse, die den "file"-Parameter eines multipart/form-data Requests aufnimmt.
Group	Die Group Klasse modelliert eine Gruppe.
GroupsFacade	Manager-Klasse für Groups
GroupsResource	RestEasy Resource-Klasse für Groups .
ImagesResource	RestEasy Resource-Klasse für den Upload von Bildern.
Permission	Die Permission -Klasse ist ein Konfigurations-Objekt für Lese-/Schreibzugriff auf einen Chat .
PermissionsFacade	Manager-Klasse für Permissions
PermissionsResource	RestEasy Resource-Klasse für Permissions .
Role	Eine Role gehört zu einer Group .
RolesFacade	Manager-Klasse für Roles
RolesResource	RestEasy Resource-Klasse für Roles .

SessionsFacade	Manager-Klasse für Sessions
User	Die User -Klasse repräsentiert einen Nutzer der Applikation.
UserProfile	Ein UserProfile gehört zu einem User und speichert persönliche Konfigurationen des Nutzers.
UserProfilesFacade	Manager-Klasse für UserProfiles
UserProfilesResource	RestEasy Resource-Klasse für UserProfiles .
UsersFacade	Manager-Klasse für Users
UsersResource	RestEasy Resource-Klasse für Users .
UserWebsocketManagement	Manager-Klasse für Session-Objekte, die für einen User registriert sind.
UserWebsocketResource	Diese Klasse akzeptiert WebSockets-Verbindungen nach "/users/{userId}/websockets".

Package com.realet.sip**Class AuthResource**

`java.lang.Object`
`com.realet.sip.AuthResource`

```
@Path("/auth")
public class AuthResource
extends Object
```

RestEasy Resource-Klasse, die für Authentifizierung zuständig ist.

Constructor Summary**Constructors**

Constructor	Description
<code>AuthResource()</code>	

Method Summary**All Methods Instance Methods Concrete Methods**

Modifier and Type	Method	Description
<code>javax.ws.rs.core.Response</code>	<code>login (String token)</code>	Findet eine <code>Session</code> anhand des "token"-Parameters und entfernt sie aus der Datenbank.
<code>javax.ws.rs.core.Response</code>	<code>login (String email, String password)</code>	Erstellt eine <code>Session</code> und speichert sie in der Datenbank, falls die Authentifizierung erfolgreich verläuft.
<code>javax.ws.rs.core.Response</code>	<code>validate (String token)</code>	Überprüft ob eine Session existiert

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait`

```
public AuthResource()
```

Method Detail

login

```
@POST @Path("/login") @Consumes("application/x-www-form-urlencoded")
@Produces("application/json") public javax.ws.rs.core.Response login
(@FormParam("email") String email, @FormParam("password") String password)
```

Erstellt eine **Session** und speichert sie in der Datenbank, falls die Authentifizierung erfolgreich verläuft.

Parameters:

email -

password -

Returns:

Status Code 200 mit JSON-Objekt {"token":<Session.token>,"userId":<User.id>} Status Code 403, falls die Authentifizierung fehlschlägt

login

```
@POST @Path("/logout") @Consumes("application/x-www-form-urlencoded")
public javax.ws.rs.core.Response login(@FormParam("token") String token)
```

Findet eine **Session** anhand des "token"-Parameters und entfernt sie aus der Datenbank.

Parameters:

token -

Returns:

Status Code 200 Status Code 403, falls keine Session mit diesem **Session.token** gefunden werden konnte.

validate

token -

Returns:

Status Code 200 mit {"userId":<User.id>} Status Code 403, falls keine Session mit diesem Session.token existiert.

Package com.realet.sip**Class Chat**

```
java.lang.Object
  com.realet.sip.Chat
```

```
@Entity
public class Chat
extends Object
```

Die **Chat** Klasse ist ein Behälter für **ChatMessages**. Ein **Chat** kann entweder zu einer **Group** gehören, in diesem Fall hat er einen **name**, oder zwei **Users** haben, wodurch er eine Direkte-Unterhaltung modelliert. Definiert eineNamedQuery "Chat.findByUsers", welchen eine Direkt-Unterhaltung zweier **User** findet. Akzeptiert zwei Parameter: "user1", "user2".

Definiert eineNamedQuery "Chat.findByGroup", welche alle **Chats** einer **Group** findet. Akzeptiert einen Parameter: "group".

Definiert eine NamedNativeQuery "Chat.findDirectChatsByUser", welche alle Direkt-Unterhaltungen eines **Users** findet und diese nach **ChatMessage.sent** der zuletzt gesendeten **ChatMessage** sortiert. Akzeptiert einen Parameter: 1: **User.id**.

Ein JPA-Entity für eine Tabelle names "Chats".

Field Summary**Fields**

Modifier and Type	Field	Description
private Group	group	
private long	id	
private List<ChatMessage>	messages	
private String	name	Dieses Attribut existiert nur, wenn der Chat zu einer Group gehört (group ist gesetzt).
private List<Permission>	permissions	
private User	user1	
private User	user2	

Constructor Summary

Chat(**Group** group, **User** user1, **User** user2, **String** name)

Method Summary

[All Methods](#) [Instance Methods](#) [Concrete Methods](#)

Modifier and Type	Method	Description
boolean	<code>equals(Object obj)</code>	Diese Methode vergleicht nur die <code>id</code> beider Objekte.
Group	<code>getGroup()</code>	
long	<code>getId()</code>	
<code>List<ChatMessage></code>	<code>getMessages()</code>	
String	<code>getName()</code>	
<code>List<Permission></code>	<code>getPermissions()</code>	
User	<code>getUser1()</code>	
User	<code>getUser2()</code>	
int	<code>hashCode()</code>	
void	<code>setGroup(Group group)</code>	
void	<code>setId(long id)</code>	
void	<code>setMessages (<code>List<ChatMessage></code> messages)</code>	
void	<code>setName(String name)</code>	
void	<code>setPermissions (<code>List<Permission></code> permissions)</code>	
void	<code> setUser1(User user1)</code>	
void	<code> setUser2(User user2)</code>	

Methods inherited from class `java.lang.Object`

`clone`, `finalize`, `getClass`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

Field Detail

group

```
private Group group
```

user1

```
private User user1
```

user2

```
private User user2
```

name

```
private String name
```

Dieses Attribut existiert nur, wenn der Chat zu einer Group gehört (group ist gesetzt).

messages

```
private List<ChatMessage> messages
```

permissions

```
private List<Permission> permissions
```

Constructor Detail

Chat

```
public Chat(Group group,
            User user1,
            User user2,
            String name)
```

Method Detail**getId**

```
public long getId()
```

setId

```
public void setId(long id)
```

getGroup

```
public Group getGroup()
```

setGroup

```
public void setGroup(Group group)
```

getUser1

```
public User getUser1()
```

getUser2

```
public User getUser2()
```

setUser2

```
public void setUser2(User user2)
```

hashCode

```
public int hashCode()
```

Overrides:

`hashCode` in class `Object`

equals

```
public boolean equals(Object obj)
```

Diese Methode vergleicht nur die `id` beider Objekte.

Overrides:

`equals` in class `Object`

getName

```
public String getName()
```

setName

```
public void setName(String name)
```

setMessages

```
public void setMessages(List<ChatMessage> messages)
```

getPermissions

```
public List<Permission> getPermissions()
```

setPermissions

```
public void setPermissions(List<Permission> permissions)
```

Package com.realet.sip

Class ChatMessage

```
java.lang.Object
  com.realet.sip.ChatMessage
```

```
@Entity
public class ChatMessage
extends Object
```

Diese Klasse repräsentiert eine einzelne [ChatMessage](#). Sie enthält Text und gehört zu einem [Chat](#). Eine [ChatMessage](#) hat einen [author](#) und enthält Informationen darüber, wann sie geschrieben, und das letzte Mal bearbeitet wurde. Ein JPA-Entity für eine Tabelle names "ChatMessages".

Field Summary

Fields

Modifier and Type	Field	Description
private User	author	
private Chat	chat	
private String	content	
private Date	edited	
private Date	expires	Nicht implementiert.
private long	id	
private Date	sent	

Constructor Summary

Constructors

Constructor	Description
ChatMessage()	
ChatMessage(String content, Date expires, Date sent, Chat chat, User author, Date edited)	

Type

boolean	<code>equals(Object obj)</code>	Diese Methode vergleicht nur die <code>id</code> beider Objekte.
User	<code>getAuthor()</code>	
Chat	<code>getChat()</code>	
String	<code>getContent()</code>	
Date	<code>getEdited()</code>	
Date	<code>getExpires()</code>	
long	<code>getId()</code>	
Date	<code>getSent()</code>	
int	<code>hashCode()</code>	
void	<code>setAuthor(User author)</code>	
void	<code>setChat(Chat chat)</code>	
void	<code>setContent(String content)</code>	
void	<code>setEdited(Date edited)</code>	
void	<code>setExpires(Date expires)</code>	
void	<code>setId(long id)</code>	
void	<code>setSent(Date sent)</code>	

Methods inherited from class java.lang.Object

`clone, finalize, getClass, notify, notifyAll, toString, wait, wait`

Field Detail**id**

`private long id`

content

expires

```
private Date expires
```

Nicht implementiert.

sent

```
private Date sent
```

chat

```
private Chat chat
```

author

```
private User author
```

edited

```
private Date edited
```

Constructor Detail

ChatMessage

```
public ChatMessage(String content,  
                   Date expires,  
                   Date sent,  
                   Chat chat,  
                   User author,  
                   Date edited)
```

Method Detail**getId**

```
public long getId()
```

setId

```
public void setId(long id)
```

getContent

```
public String getContent()
```

setContent

```
public void setContent(String content)
```

getExpires

```
public Date getExpires()
```

setExpires

```
public void setExpires(Date expires)
```

setChat

```
public void setChat(Chat chat)
```

getAuthor

```
public User getAuthor()
```

setAuthor

```
public void setAuthor(User author)
```

getSent

```
public Date getSent()
```

setSent

```
public void setSent(Date sent)
```

getEdited

```
public Date getEdited()
```

setEdited

```
public void setEdited(Date edited)
```

hashCode in class Object

equals

```
public boolean equals(Object obj)
```

Diese Methode vergleicht nur die `id` beider Objekte.

Overrides:

[equals](#) in class [Object](#)

Package com.realet.sip

Class ChatMessagesFacade

`java.lang.Object`
`com.realet.sip.ChatMessagesFacade`

```
public class ChatMessagesFacade
extends Object
```

Manager-Klasse für ChatMessages.

Field Summary

Fields

Modifier and Type	Field	Description
(package private) static javax.persistence.EntityManagerFactory	<code>emf</code>	Die EntityManagerFactory, die für alle Operationen von ChatMessagesFacade verwendet wird.

Constructor Summary

Constructors

Constructor	Description
<code>ChatMessagesFacade()</code>	

Method Summary

All Methods Static Methods Concrete Methods

Modifier and Type	Method	Description
static void	<code>add(ChatMessage chatMessage)</code>	
static <code>List<ChatMessage></code>	<code>find(Chat chat, int count, long before, long after, long unblockedById, long reverseBlocking)</code>	Benutzt die CriteriaQuery API um dynamisch ChatMessages nach bestimmten Kriterien zu finden.

SUMMARY: NESTED | FIELD | CONSTR | METHOD

DETAIL: FIELD | CONSTR | METHOD

static void	<code>remove(ChatMessage chatMessage)</code>
static void	<code>update(ChatMessage chatMessage)</code>
static ChatMessage	<code>updateContent(ChatMessage chatMessage)</code>

Aktualisiert den `ChatMessage.content` einer `ChatMessage` und setzt den `ChatMessage.edited` Timestamp auf die aktuelle Zeit.

Methods inherited from class java.lang.Object

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`

Field Detail

emf

`static javax.persistence.EntityManagerFactory emf`

Die EntityManagerFactory, die für alle Operationen von `ChatMessagesFacade` verwendet wird.

Constructor Detail

ChatMessagesFacade

`public ChatMessagesFacade()`

Method Detail

initialize

findById

```
public static Optional<ChatMessage> findById(long id)
```

find

```
public static List<ChatMessage> find(Chat chat, int count, long before, long after,  
long unblockedById, long reverseBlocking)
```

Benutzt die CriteriaQuery API um dynamisch ChatMessages nach bestimmten Kriterien zu finden.
Diese werden aufsteigend nach ChatMessage.id sortiert.

Parameters:

chat - In welchem Chat gesucht werden soll

count - Wie viele ChatMessages gesammelt werden sollen.

before - ID der ChatMessage, vor deren Sendedatum die gesuchten ChatMessages gesendet werden
sein sollen. Wird ignoriert wenn der gegebene Wert 0 ist.

after - ID der ChatMessage, nach deren Sendedatum die gesuchten ChatMessages gesendet werden
sein sollen. Wird ignoriert wenn der gegebene Wert 0 ist.

unblockedById - Spezifiziert die ID des Nutzers der die Abfrage stellt und schließt alle Nachrichten,
deren Autor von dem ihm blockiert ist, aus.

reverseBlocking - Legt fest, ob Nachrichten, deren Autor den Nutzer der die Abfrage stellt blockiert
haben, eingeschlossen werden sollen.

Returns:

Gefundene ChatMessages

add

```
public static void add(ChatMessage chatMessage)
```

update

```
public static void update(ChatMessage chatMessage)
```

Timestamp auf die aktuelle Zeit.

Returns:**remove**

```
public static void remove(ChatMessage chatMessage)
```

Package com.realet.sip

Class ChatMessagesResource

```
java.lang.Object
  com.realet.sip.ChatMessagesResource
```

```
@Path("/chat-messages")
public class ChatMessagesResource
extends Object
```

RestEasy Resource-Klasse für ChatMessages.

Constructor Summary

Constructors

Constructor	Description
<code>ChatMessagesResource()</code>	

Method Summary

[All Methods](#) [Instance Methods](#) [Concrete Methods](#)

Modifier and Type	Method	Description
<code>javax.ws.rs.core.Response</code>	<code>removeMessage(long id, String token)</code>	Entfernt eine ChatMessage anhand ihrer ChatMessage.id.
<code>javax.ws.rs.core.Response</code>	<code>updateChatMessage (ChatMessage chatMessage, String token)</code>	Aktualisiert den ChatMessage.content einer ChatMessage.

Methods inherited from class java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait,
wait, wait
```

Constructor Detail

Method Detail

removeMessage

```
@DELETE @Path("/{id}") public javax.ws.rs.core.Response removeMessage(@PathParam("id") long id, @HeaderParam("Authorization") String token)
```

Entfernt eine `ChatMessage` anhand ihrer `ChatMessage.id`. Sendet die entsprechende Information an alle WebSockets, die den Chat beobachten.

Returns:

Status Code 200, Status Code 404, falls keine `ChatMessage` mit dieser `ChatMessage.id` existiert, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt.

updateChatMessage

```
@PUT @Path("/") @Consumes("application/json") public javax.ws.rs.core.Response updateChatMessage(ChatMessage chatMessage, @HeaderParam("Authorization") String token)
```

Aktualisiert den `ChatMessage.content` einer `ChatMessage`. Sendet die entsprechende Information an alle WebSockets, die den Chat beobachten. Entfernt Leerzeichen am Anfang und Ende des `ChatMessage.content`.

Returns:

Status Code 200 Status Code 404, falls keine `ChatMessage` mit derselben `ChatMessage.id` gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status Code 400, falls der `ChatMessage.content` der neuen `ChatMessage` leer oder nicht vorhanden ist, oder nur Leerzeichen enthält.

Package com.realet.sip

Class ChatsFacade

```
java.lang.Object
  com.realet.sip.ChatsFacade
```

```
public class ChatsFacade
extends Object
```

Manager-Klasse für Chats.

Field Summary

Fields

Modifier and Type	Field	Description
(package private) static javax.persistence.EntityManagerFactory	<code>emf</code>	Die EntityManagerFactory, die für alle Operationen von <code>ChatsFacade</code> verwendet wird.

Constructor Summary

Constructors

Constructor	Description
<code>ChatsFacade()</code>	

Method Summary

[All Methods](#) [Static Methods](#) [Concrete Methods](#)

Modifier and Type	Method	Description
static void	<code>add(Chat chat)</code>	
static <code>List<Chat></code>	<code>findByGroup(Group group)</code>	
static <code>Optional<Chat></code>	<code>findById(long id)</code>	

[LIST<Chat>](#)

Unterhaltungen eines [Users](#) und sortiert sie nach dem Sendedatum der zuletzt gesendeten [ChatMessage](#).

```
static void initialize  
        (javax.persistence.EntityManagerFactory emf)  
  
static void remove(Chat chat)  
  
static void update(Chat chat)
```

Setzt [emf](#)

Methods inherited from class java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait,  
wait, wait
```

Field Detail

emf

```
static javax.persistence.EntityManagerFactory emf
```

Die EntityManagerFactory, die für alle Operationen von [ChatsFacade](#) verwendet wird.

Constructor Detail

ChatsFacade

```
public ChatsFacade()
```

Method Detail

findById

```
public static Optional<Chat> findById(long id)
```

findByUsers

```
public static Optional<Chat> findByUsers(User user1, User user2)
```

Findet Direkt-Unterhaltung zweier User.

findDirectChatsByUser

```
public static List<Chat> findDirectChatsByUser(User user)
```

Findet alle Direkt-Unterhaltungen eines Users und sortiert sie nach dem Sendedatum der zuletzt gesendeten ChatMessage.

findByGroup

```
public static List<Chat> findByGroup(Group group)
```

add

```
public static void add(Chat chat)
```

update

```
public static void update(Chat chat)
```

remove

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

ALL CLASSES

SEARCH:

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Copyright © 2021. All rights reserved.

Package com.realet.sip

Class ChatsResource

```
java.lang.Object
  com.realet.sip.ChatsResource
```

```
@Path("/chats")
public class ChatsResource
extends Object
```

RestEasy Resource-Klasse für Chats.

Constructor Summary

Constructors

Constructor	Description
ChatsResource()	

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
javax.ws.rs.core.Response	addChatMessage (ChatMessage chatMessage, long chatId, String token)	Fügt eine ChatMessage zu einem Chat hinzu.
javax.ws.rs.core.Response	addPermission (long chatId, Permission permission, String token)	Fügt eine Permission für einen Chat hinzu.
javax.ws.rs.core.Response	deleteChat (long chatId, String token)	Entfernt einen Chat und alle seine ChatMessages.
javax.ws.rs.core.Response	getChatMessages (long chatId, int count, long beforeId, long afterId, long userId, long reverseBlocking, String token)	Findet ChatMessages eines Chats mittels <code>ChatMessagesFacade.find(com.realet.sip.Chat, int, long, long, long, long)</code> .
javax.ws.rs.core.Response	getChatPermissions (long chatId, long userId, String token)	Findet alle Permissions eines Chats.
javax.ws.rs.core.Response	getChats (long user1Id,	Findet eine Direkt-Unterhaltung zweier User.

Methods inherited from class java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

Constructor Detail**ChatsResource**

```
public ChatsResource()
```

Method Detail**getChats**

```
@GET @Path("") @Produces("application/json") public javax.ws.rs.core.Response getChats
(@QueryParam("user1") long user1Id, @QueryParam("user2") long user2Id,
@HeaderParam("Authorization") String token)
```

Findet eine Direkt-Unterhaltung zweier User.

Parameters:

user1Id -

user2Id -

token -

Returns:

Status Code 200 mit Chat, serialisiert durch ChatAdapter, Status Code 404, falls ein User nicht vorhanden ist, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt,

getChatMessages

```
@GET @Path("/{chatId}/chat-messages") @Produces("application/json")
public javax.ws.rs.core.Response getChatMessages(@PathParam("chatId") long chatId,
@QueryParam("count") int count, @QueryParam("before") long beforeId, @QueryParam("after")
long afterId, @QueryParam("authorUnblockedBy") long userId, @QueryParam("reverseBlocking")
long reverseBlocking, @HeaderParam("Authorization") String token)
```

Findet ChatMessages eines Chats mittels ChatMessagesFacade.find(com.realet.sip.Chat, int, long, long, long, long).

afterId -

userId -

reverseBlocking -

token -

Returns:

Status Code 200 mit `ChatMessages`, serialisiert durch `ChatMessageAdapter` mit einer `ChatMessageAdapter.verbosity` von 1, Status Code 404, falls der `Chat` oder eine angegebene einschränkende `ChatMessage` nicht vorhanden ist, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status Code 400, falls "count" nicht zwischen 1 und 1000 liegt oder sowohl "beforeId" und "afterId" ungleich 0 sind.

addChatMessage

```
@POST @Path("/{chatId}/chat-messages") @Consumes("application/json")
public javax.ws.rs.core.Response addChatMessage(ChatMessage chatMessage, @PathParam("chatId")
long chatId, @HeaderParam("Authorization") String token)
```

Fügt eine `ChatMessage` zu einem `Chat` hinzu. Sendet diese `ChatMessage` an alle WebSockets, die für `User` registriert sind, welche diesen `Chat` lesen können. Entfernt Leerzeichen am Anfang und Ende des `ChatMessage.content`.

Parameters:

chatMessage -

chatId -

token -

Returns:

Status Code 201, Status Code 404, falls der `Chat` nicht vorhanden ist, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status Code 400, falls der `ChatMessage.content` nicht vorhanden oder leer ist, oder nur aus Leerzeichen besteht.

updateChat

```
@PUT @Path("") @Consumes("application/json") public javax.ws.rs.core.Response updateChat
(Chat inputChat, @HeaderParam("Authorization") String token)
```

Aktualisiert einen `Chat`. Entfernt Leerzeichen am Anfang und Ende des `Chat.name`.

Parameters:

inputChat -

token -

Returns:

Status Code 200, Status Code 404, falls der `Chat` nicht vorhanden ist, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status Code 400, falls der `ChatMessage.content` nicht vorhanden oder leer ist, oder nur aus Leerzeichen besteht, oder der `Chat` eine Direkt-Unterhaltung ist.

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

```
long chatId, @HeaderParam("Authorization") String token)
```

Entfernt einen Chat und alle seine ChatMessages.

Parameters:

chatId -

token -

Returns:

Status Code 200, Status Code 404, falls der Chat nicht vorhanden ist, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status Code 400, falls der Chat eine Direkt-Unterhaltung ist.

getChatPermissions

```
@GET @Path("/{chatId}/permissions") @Produces("application/json")
public javax.ws.rs.core.Response getChatPermissions(@PathParam("chatId") long chatId,
@QueryParam("user") long userId, @HeaderParam("Authorization") String token)
```

Findet alle Permissions eines Chats.

Parameters:

chatId -

userId -

token -

Returns:

Status Code 200 mit Permissions, serialisiert durch PermissionAdapter, Status Code 404, falls der Chat nicht vorhanden ist, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status Code 400, falls der Chat eine Direkt-Unterhaltung ist.

addPermission

```
@POST @Path("/{chatId}/permissions") @Consumes("application/json")
public javax.ws.rs.core.Response addPermission(@PathParam("chatId") long chatId,
Permission permission, @HeaderParam("Authorization") String token)
```

Fügt eine Permission für einen Chat hinzu.

Parameters:

chatId -

permission -

token -

Returns:

Status Code 200, Status Code 404, falls der Chat nicht vorhanden ist, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status Code 400, falls die Permission keine Permission.role hat, die Permission.role nicht zur Chat.group gehört oder bereits eine Permission für die Permission.role und den Chat existiert.

ALL CLASSES

SEARCH:

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD
ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD
Copyright © 2021. All rights reserved.

Package com.realet.sip

Class ChatWebSocketManagement

`java.lang.Object`
 com.realet.sip.ChatWebSocketManagement

```
public class ChatWebSocketManagement
extends Object
```

Manager-Klasse für Session-Objekte, die für einen Chat registriert sind.

Field Summary

Fields

Modifier and Type	Field	Description
(package private) static <code>HashMap<Long,</code> <code>ArrayList<javax.websocket.Session>></code>	<code>sessions</code>	Format: chatId \Rightarrow Session.

Constructor Summary

Constructors

Constructor	Description
<code>ChatWebSocketManagement()</code>	

Method Summary

All Methods Static Methods Concrete Methods

Modifier and Type	Method	Description
static void	<code>addSession(long chatId,</code> <code>javax.websocket.Session session)</code>	
static <code>ArrayList<javax.websocket.Session></code>	<code>getSessions(long chatId)</code>	
static void	<code>removeSession(long chatId,</code> <code>javax.websocket.Session session)</code>	

Field Detail

sessions

```
static HashMap<Long,ArrayList<javax.websocket.Session>> sessions
```

Format: chatId => Session.

Constructor Detail

ChatWebsocketManagement

```
public ChatWebsocketManagement()
```

Method Detail

getSessions

```
public static ArrayList<javax.websocket.Session> getSessions(long chatId)
```

addSession

```
public static void addSession(long chatId, javax.websocket.Session session)
```

removeSession

```
public static void removeSession(long chatId, javax.websocket.Session session)
```

ALL CLASSES

SEARCH:

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Copyright © 2021. All rights reserved.

Package com.realet.sip**Class ChatWebSocketResource**

`java.lang.Object`
`com.realet.sip.ChatWebSocketResource`

```
public class ChatWebSocketResource
extends Object
```

Diese Klasse akzeptiert WebSockets-Verbindungen nach "/chats/{chatId}/websockets". Diese werden benutzt, um live Ereignisse aus einen bestimmten Chat an den Client zu übermitteln. Das schliesst neue Nachrichten aus, da diese durch die `UserWebSocketResource` gesendet werden.

Dies geschieht in folgendem Format:

Nachricht geändert: "updated: <ChatMessage als JSON>"

Nachricht geändert: "removed: <ChatMessage als JSON>"

Constructor Summary**Constructors**

Constructor	Description
<code>ChatWebSocketResource()</code>	

Method Summary**All Methods Instance Methods Concrete Methods**

Modifier and Type	Method	Description
void	<code>onClose(javax.websocket.Session session, javax.websocket.CloseReason closeReason, long chatId)</code>	Entfernt die WebSockets-Session aus dem <code>ChatWebSocketManagement</code> .
void	<code>onError(javax.websocket.Session session, Throwable throwable, long chatId)</code>	Entfernt die WebSockets-Session aus dem <code>ChatWebSocketManagement</code> .
void	<code>OnMessage(javax.websocket.Session session, String message, long chatId)</code>	Ueberprueft Gueltigkeit und Vorrechte einer im Format "Bearer <token>" uebermittelten <code>Session</code> und registriert die WebSockets-Session im <code>ChatWebSocketManagement</code> .
void	<code>onOpen(javax.websocket.Session session,</code>	Ueberprueft, ob der jeweilige <code>Chat</code>

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait

Constructor Detail

ChatWebsocketResource

```
public ChatWebsocketResource()
```

Method Detail

onOpen

```
public void onOpen(javax.websocket.Session session, long chatId)
```

Ueberprueft, ob der jeweilige Chat existiert und sendet andernfalls eine entsprechende Nachricht, woraufhin die Verbindung geschlossen wird.

OnMessage

```
public void OnMessage(javax.websocket.Session session, String message, long chatId)
```

Ueberprueft Gueltigkeit und Vorrechte einer im Format "Bearer <token>" übermittelten Session und registriert die WebSockets-Session im ChatwebsocketManagement. Sendet gegebenenfalls eine Fehlermeldung zurück.

onClose

```
public void onClose(javax.websocket.Session session, javax.websocket.CloseReason closeReason, long chatId)
```

Entfernt die WebSockets-Session aus dem ChatwebsocketManagement.

[SUMMARY: NESTED | FIELD | CONSTR | METHOD](#) [DETAIL: FIELD | CONSTR | METHOD](#)`long chatId)`

Entfernt die WebSockets-Session aus dem ChatWebsocketManagement.

[SUMMARY: NESTED | FIELD | CONSTR | METHOD](#) [DETAIL: FIELD | CONSTR | METHOD](#)

Copyright © 2021. All rights reserved.

Package com.realet.sip

Class CORSPreflightHandler

java.lang.Object
com.realet.sip.CORSPreflightHandler

```
@Path("/")
public class CORSPreflightHandler
extends Object
```

Beantwortet alle OPTIONS-Anfragen mit den korrekten HTTP-Headers und einer 200 OK Antwort, um CORS zu erlauben.

Constructor Summary

Constructors

Constructor	Description
CORSPreflightHandler()	

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
javax.ws.rs.core.Response	options()	

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait

Constructor Detail

CORSPreflightHandler

```
public CORSPreflightHandler()
```

~~Method Detail~~**options**

```
@OPTIONS @Path("{path : .*}") public javax.ws.rs.core.Response options()
```

Package com.realet.sip

Class FileUploadForm

java.lang.Object
com.realet.sip.FileUploadForm

```
public class FileUploadForm  
extends Object
```

Behälter-Klasse, die den "file"-Parameter eines multipart/form-data Requests aufnimmt.

Field Summary

Fields

Modifier and Type	Field	Description
private byte[]	data	

Constructor Summary

Constructors

Constructor	Description
FileUploadForm()	

Method Summary

[All Methods](#) [Instance Methods](#) [Concrete Methods](#)

Modifier and Type	Method	Description
byte[]	getData()	
void	setData(byte[] data)	

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#)

data

```
private byte[] data
```

Constructor Detail**FileUploadForm**

```
public FileUploadForm()
```

Method Detail**getData**

```
public byte[] getData()
```

setData

```
public void setData(byte[] data)
```

Package com.realet.sip**Class Group**

```
java.lang.Object
  com.realet.sip.Group
```

```
@Entity
public class Group
extends Object
```

Die **Group** Klasse modelliert eine Gruppe. Eine Gruppe ist eine Sammlung von **Roles**, **Users** und **Chats**. **Chats**, die zu einer **Group** gehören, können Namen haben.

Definiert eine NamedQuery "Group.findShared", welche alle gemeinsamen Gruppen zweier **User** findet.

Akzeptiert zwei Parameter: "user1", "user2".

Ein JPA-Entity für eine Tabelle names "Groups".

Field Summary**Fields**

Modifier and Type	Field	Description
private List<Chat>	chats	
private String	description	
private long	id	
private String	name	
private User	owner	
private String	picture	Dateiname des Bilds ohne Pfad oder Endung.
private List<Role>	roles	
private Set<User>	users	

Constructor Summary**Constructors**

Constructor	Description
Group()	
Group(String name, String description, String picture, User owner)	

Modifier and Type	Method	Description
boolean	<code>equals(Object obj)</code>	Diese Methode vergleicht nur die <code>id</code> beider Objekte.
<code>List<Chat></code>	<code>getChats()</code>	
<code>String</code>	<code>getDescription()</code>	
long	<code>getId()</code>	
<code>String</code>	<code>getName()</code>	
<code>User</code>	<code>getOwner()</code>	
<code>String</code>	<code>getPicture()</code>	
<code>List<Role></code>	<code>getRoles()</code>	
<code>Set<User></code>	<code>getUsers()</code>	
int	<code>hashCode()</code>	
void	<code>setChats(List<Chat> chats)</code>	
void	<code>setDescription(String description)</code>	
void	<code>setId(long id)</code>	
void	<code>setName(String name)</code>	
void	<code>setOwner(User owner)</code>	
void	<code>setPicture(String picture)</code>	
void	<code>setRoles(List<Role> roles)</code>	
void	<code>setUsers(Set<User> users)</code>	

Methods inherited from class `java.lang.Object`

`clone, finalize, getClass, notify, notifyAll, toString, wait, wait, wait`

Field Detail

`id``private long id`

```
private String name
```

description

```
private String description
```

picture

```
private String picture
```

Dateiname des Bilds ohne Pfad oder Endung.

owner

```
private User owner
```

users

```
private Set<User> users
```

roles

```
private List<Role> roles
```

chats

```
private List<Chat> chats
```

SUMMARY: NESTED | FIELD | CONSTR | METHOD

DETAIL: FIELD | CONSTR | METHOD

```
String description,  
String picture,  
User owner)
```

Group

```
public Group()
```

Method Detail

getId

```
public long getId()
```

setId

```
public void setId(long id)
```

getName

```
public String getName()
```

setName

```
public void setName(String name)
```

getDescription

```
public String getDescription()
```

getPicture

```
public String getPicture()
```

setPicture

```
public void setPicture(String picture)
```

getOwner

```
public User getOwner()
```

setOwner

```
public void setOwner(User owner)
```

getUsers

```
public Set<User> getUsers()
```

setUsers

```
public void setUsers(Set<User> users)
```

getRoles

```
public List<Role> getRoles()
```

getChats

```
public List<Chat> getChats()
```

setChats

```
public void setChats(List<Chat> chats)
```

hashCode

```
public int hashCode()
```

Overrides:

[hashCode](#) in class [Object](#)

equals

```
public boolean equals(Object obj)
```

Diese Methode vergleicht nur die [id](#) beider Objekte.

Overrides:

[equals](#) in class [Object](#)

Package com.realet.sip

Class GroupsFacade

```
java.lang.Object
  com.realet.sip.GroupsFacade
```

```
public class GroupsFacade
extends Object
```

Manager-Klasse für Groups.

Field Summary

Fields

Modifier and Type	Field	Description
(package private) static javax.persistence.EntityManagerFactory	emf	Die EntityManagerFactory, die für alle Operationen von GroupsFacade verwendet wird.

Constructor Summary

Constructors

Constructor	Description
GroupsFacade()	

Method Summary

[All Methods](#) [Static Methods](#) [Concrete Methods](#)

Modifier and Type	Method	Description
static void	add(Group group)	
static	findById(long id) Optional<Group>	
static	findShared(User user1, User user2) List<Group>	Findet alle gemeinsamen Groups zweier User.

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait

Field Detail

emf

static javax.persistence.EntityManagerFactory emf

Die EntityManagerFactory, die für alle Operationen von [GroupsFacade](#) verwendet wird.

Constructor Detail

GroupsFacade

public GroupsFacade()

Method Detail

initialize

public static void initialize(javax.persistence.EntityManagerFactory emf)

Setzt emf.

findById

public static Optional<Group> findById(long id)

```
public static List<Group> findShared(User user1, User user2)
```

Findet alle gemeinsamen Groups zweier User.

add

```
public static void add(Group group)
```

update

```
public static void update(Group group)
```

remove

```
public static void remove(Group group)
```

Package com.realet.sip**Class GroupsResource**

`java.lang.Object`
`com.realet.sip.GroupsResource`

```
@Path("/groups")
public class GroupsResource
extends Object
```

RestEasy Resource-Klasse für [Groups](#).

Constructor Summary**Constructors**

Constructor	Description
GroupsResource()	

Method Summary**All Methods** **Instance Methods** **Concrete Methods**

Modifier and Type	Method	Description
<code>javax.ws.rs.core.Response</code>	<code>addUser(long groupId, User inputUser, String token)</code>	Fügt einen User zu einer Group hinzu.
<code>javax.ws.rs.core.Response</code>	<code>createGroup(Group group, String token)</code>	Erstellt eine Group .
<code>javax.ws.rs.core.Response</code>	<code>deleteGroup(long groupId, String token)</code>	Löscht eine Group und alle dazugehörigen Chats und Roles .
<code>javax.ws.rs.core.Response</code>	<code>getBasicUsers(long groupId, String token)</code>	Findet alle User ohne Role in einer Group .
<code>javax.ws.rs.core.Response</code>	<code>getChats(long groupId, String token)</code>	Findet alle Chats einer Group .
<code>javax.ws.rs.core.Response</code>	<code>getGroup(long groupId, String token)</code>	Findet eine Group .
<code>javax.ws.rs.core.Response</code>	<code>getRoles(long groupId,</code>	Findet alle Roles einer

SUMMARY: NESTED FIELD CONSTR METHOD	DETAIL: FIELD CONSTR METHOD
<code>java.ws.rs.core.Response removeUser (long groupId, long userId, String token)</code>	Entfernt einen User aus einer Group.
<code>javax.ws.rs.core.Response updateChat (long groupId, Chat inputChat, String token)</code>	Erstellt einen Chat und fügt ihn zu einer Group hinzu.
<code>javax.ws.rs.core.Response updateGroup (Group group, String token)</code>	Aktualisiert eine Group.
<code>javax.ws.rs.core.Response updatePriorities (long groupId, String requestBody, String token)</code>	Verändert Role.priority aller Roles in einer Group.
<code>javax.ws.rs.core.Response updateRole (long groupId, Role role, String token)</code>	Erstellt eine Role und fügt sie zu einer Group hinzu.

Methods inherited from class java.lang.Object

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait,
wait, wait`

Constructor Detail

GroupsResource

```
public GroupsResource()
```

Method Detail

getGroup

```
@GET @Path("/{groupId}") @Produces("application/json")
public javax.ws.rs.core.Response getGroup(@PathParam("groupId") long groupId,
@HeaderParam("Authorization") String token)
```

Findet eine Group.

Parameters:

groupId -

getChats

```
@GET @Path("/{groupId}/chats") @Produces("application/json")
public javax.ws.rs.core.Response getChats(@PathParam("groupId") long groupId,
@HeaderParam("Authorization") String token)
```

Findet alle [Chats](#) einer Group.

Parameters:

groupId -

token -

Returns:

Status Code 200 mit [Chats](#), serialisiert durch [ChatAdapter](#), Status Code 404, falls die [Group](#) nicht gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt.

getBasicUsers

```
@GET @Path("/{groupId}/basic-users") @Produces("application/json")
public javax.ws.rs.core.Response getBasicUsers(@PathParam("groupId") long groupId,
@HeaderParam("Authorization") String token)
```

Findet alle [User](#) ohne [Role](#) in einer Group.

Parameters:

groupId -

token -

Returns:

Status Code 200 mit [Usern](#), serialisiert durch [UserAdapter](#), Status Code 404, falls die [Group](#) nicht gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt.

getUsers

```
@GET @Path("/{groupId}/users") @Produces("application/json")
public javax.ws.rs.core.Response getUsers(@PathParam("groupId") long groupId,
@HeaderParam("Authorization") String token)
```

Returns:

Status Code 200 mit [Usern](#), serialisiert durch [UserAdapter](#), Status Code 404, falls die [Group](#) nicht gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt.

getRoles

```
@GET @Path("/{groupId}/roles") @Produces("application/json")
public javax.ws.rs.core.Response getRoles(@PathParam("groupId") long groupId,
@HeaderParam("Authorization") String token)
```

Findet alle [Roles](#) einer Group.

Parameters:

groupId -

token -

Returns:

Status Code 200 mit [Roles](#), serialisiert durch [RoleAdapter](#) mit einer [RoleAdapter.verbosity](#) von 1, Status Code 404, falls die [Group](#) nicht gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt.

createGroup

```
@POST public javax.ws.rs.core.Response createGroup(Group group,
@HeaderParam("Authorization") String token)
```

Erstellt eine [Group](#). Erstellt automatisch einen [Chat](#) und eine [Permission](#) mit [Permission.canRead == true](#) und [Permission.canWrite == true](#). Entfernt Leerzeichen am Anfang und Ende des [Group.name](#).

Parameters:

group -

token -

Returns:

Status Code 201, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status code 400, falls der [Group.name](#) nicht vorhanden oder leer ist, oder nur aus Leerzeichen besteht.

updateChat

SUMMARY: NESTED | FIELD | CONSTR | METHOD

DETAIL: FIELD | CONSTR | METHOD

Erstellt einen `Chat` und fügt ihn zu einer `Group` hinzu. Erstellt automatisch eine `Permission` mit `Permission.canRead == true` und `Permission.canWrite == true`. Entfernt Leerzeichen am Anfang und Ende des `Chat.name`.

Parameters:`groupId` -`inputChat` -`token` -**Returns:**

Status Code 201, Status Code 404, falls die `Group` nicht gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status code 400, falls der `Chat.name` nicht vorhanden oder leer ist, oder nur aus Leerzeichen besteht.

updateGroup

```
@PUT @Consumes("application/json") public javax.ws.rs.core.Response updateGroup
(@PathParam("group") Group group, @HeaderParam("Authorization") String token)
```

Aktualisiert eine `Group`. `Group.owner` wird in jedem Fall beibehalten. Entfernt Leerzeichen am Anfang und Ende des `Group.name`.

Parameters:`group` -`token` -**Returns:**

Status Code 201, Status Code 404, falls keine `Group` mit derselben `Group.id` gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status code 400, falls der `Group.name` nicht vorhanden oder leer ist, oder nur aus Leerzeichen besteht.

addUser

```
@POST @Path("/{groupId}/users") public javax.ws.rs.core.Response addUser
(@PathParam("groupId") long groupId, User inputUser, @HeaderParam("Authorization")
String token)
```

Fügt einen `User` zu einer `Group` hinzu. Diese Methode verändert den in der Datenbank gespeicherten `User` nicht.

Parameters:`groupId` -`inputUser` -

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD
Falls der **User** bereits Teil dieser **Group** ist.

removeUser

```
@DELETE @Path("/{groupId}/users/{userId}")
public javax.ws.rs.core.Response removeUser(@PathParam("groupId") long groupId,
@PathParam("userId") long userId, @HeaderParam("Authorization") String token)
```

Entfernt einen User aus einer Group. Der User existiert weiterhin.

Parameters:

groupId -

userId -

token -

Returns:

Status Code 200, Status Code 404, falls die Group oder der User nicht gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status code 400, falls der User nicht Teil dieser Group oder der Group.owner ist.

updateRole

```
@POST @Path("/{groupId}/roles") @Consumes("application/json")
public javax.ws.rs.core.Response updateRole(@PathParam("groupId") long groupId,
Role role, @HeaderParam("Authorization") String token)
```

Erstellt eine Role und fügt sie zu einer Group hinzu.

Parameters:

groupId -

role -

token -

Returns:

Status Code 200, Status Code 404, falls die Group nicht gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status code 400, falls die Role.color nicht vorhanden oder kein gültiger Hex-Code ist, oder der Role.name nicht vorhanden oder leer ist, oder nur aus Leerzeichen besteht.

updatePriorities

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD
Role.id ALLER Group.roles sortiert nach Priorität (wichtigste Rollen zuerst).

Parameters:

groupId -

requestBody -

token -

Returns:

Status Code 200, Status Code 404, falls die **Group** oder eine der **Roles** nicht gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status code 400, falls Array-Elemente doppelt erscheinen, nicht als Long zu parsen sind, oder die zu ihnen gehörenden **Roles** nicht Teil der **Group** sind.

deleteGroup

```
@DELETE @Path("/{groupId}") public javax.ws.rs.core.Response deleteGroup  
(@PathParam("groupId") long groupId, @HeaderParam("Authorization") String token)
```

Löscht eine **Group** und alle dazugehörigen **Chats** und **Roles**.

Parameters:

groupId -

token -

Returns:

Status Code 200, Status Code 404, falls die **Group** nicht gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt.

Package com.realet.sip**Class ImagesResource**

`java.lang.Object`
`com.realet.sip.ImagesResource`

```
@Path("/images")
public class ImagesResource
extends Object
```

RestEasy Resource-Klasse für den Upload von Bildern.

Constructor Summary**Constructors**

Constructor	Description
<code>ImagesResource()</code>	

Method Summary**All Methods Instance Methods Concrete Methods**

Modifier and Type	Method	Description
<code>javax.ws.rs.core.Response uploadGroupPicture (FileUploadForm form)</code>		Speichert ein Bild im durch die System-Property "com.realet.sip.uploadDir" beschriebenen Upload-Verzeichnis unter "pic/group/".
<code>javax.ws.rs.core.Response uploadUserPicture (FileUploadForm form)</code>		Speichert ein Bild im durch die System-Property "com.realet.sip.uploadDir" beschriebenen Upload-Verzeichnis unter "pic/user/".

Methods inherited from class java.lang.Object

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait`

```
public IImagesResource()
```

Method Detail

uploadUserPicture

```
@POST @Path("/users/pictures") @Consumes("multipart/form-data")
@Produces("application/json") public javax.ws.rs.core.Response uploadUserPicture
(FileUploadForm form)
```

Speichert ein Bild im durch die System-Property "com.realet.sip.uploadDir" beschriebenen Upload-Verzeichnis unter "pic/user/". Generiert einen einzigartigen MD5-Hash als Dateinamen und konvertiert das Bild zu einer JPG-Datei.

Parameters:

form -

Returns:

Status Code 201 mit einem JSON-Objekt {"picture":{}}, Status Code 500, falls das Schreiben der Datei fehlschlägt. Status Code 415, falls der jeweilige Bild-Typ nicht von ImageIO unterstützt wird. Status Code 400, falls das Bild größer als 30MB ist oder das Bild leer ist.

uploadGroupPicture

```
@POST @Path("/groups/pictures") @Consumes("multipart/form-data")
@Produces("application/json") public javax.ws.rs.core.Response uploadGroupPicture
(FileUploadForm form)
```

Speichert ein Bild im durch die System-Property "com.realet.sip.uploadDir" beschriebenen Upload-Verzeichnis unter "pic/group/". Generiert einen einzigartigen MD5-Hash als Dateinamen und konvertiert das Bild zu einer JPG-Datei.

Parameters:

form -

Returns:

Status Code 201 mit einem JSON-Objekt {"picture":{}}, Status Code 500, falls das Schreiben der Datei fehlschlägt. Status Code 415, falls der jeweilige Bild-Typ nicht von ImageIO unterstützt wird. Status Code 400, falls das Bild größer als 30MB ist oder das Bild leer ist.

ALL CLASSES

SEARCH:

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Copyright © 2021. All rights reserved.

Package com.realet.sip**Class Permission**

```
java.lang.Object
    com.realet.sip.Permission
```

```
@Entity
public class Permission
extends Object
```

Die **Permission**-Klasse ist ein Konfigurations-Objekt für Lese-/Schreibzugriff auf einen **Chat**. Es beschreibt entweder die grundlegenden Zugangsberechtigungen zu einem **Chat** (falls **role** nicht gesetzt ist), oder andernfalls die Zugangsberechtigungen einer **Role** zu einem **Chat**. **Permissions** funktionieren nach OR-Prinzip. Wenn also eine **Permission** Zugriff erlaubt, ist es irrelevant, wie viele **Permissions** dies nicht tun.

Definiert eine NamedQuery "Permission.findByRoleAndChat", welche eine **Permission** einer **Role** zu einem **Chat** findet. Akzeptiert zwei Parameter: "role", "chat".

Definiert eine NamedNativeQuery "Permission.findGroupChatPermissions", welche 1-3 relevante **Permissions** eines **Users** zu einem **Chat** findet; Einen Eintrag mit **canRead** == true, falls vorhanden, einen Eintrag mit **canWrite** == true, falls vorhanden und die Grundberechtigungen zu einem **Chat**, was alle notwendigen Informationen bereitstellt um die Berechtigungen eines **Users** zu erfahren. Akzeptiert einen Parameter: 1: **User.id**, 2: **Chat.id**.

Ein JPA-Entity für eine Tabelle names "Permissions".

Field Summary**Fields**

Modifier and Type	Field	Description
private boolean	canRead	
private boolean	canWrite	
private Chat	chat	
private long	id	
private Role	role	

Constructor Summary**Constructors**

Constructor	Description
Permission()	

Method Summary

[All Methods](#) [Instance Methods](#) [Concrete Methods](#)

Modifier and Type	Method	Description
boolean	<code>equals(Object obj)</code>	Diese Methode vergleicht nur die <code>id</code> beider Objekte.
<code>Chat</code>	<code>getChat()</code>	
<code>long</code>	<code>getId()</code>	
<code>Role</code>	<code>getRole()</code>	
<code>int</code>	<code>hashCode()</code>	
boolean	<code>isCanRead()</code>	
boolean	<code>isCanWrite()</code>	
void	<code>setCanRead(boolean canRead)</code>	
void	<code>setCanWrite(boolean canWrite)</code>	
void	<code>setChat(Chat chat)</code>	
void	<code>setId(long id)</code>	
void	<code>setRole(Role role)</code>	

Methods inherited from class java.lang.Object

`clone, finalize, getClass, notify, notifyAll, toString, wait, wait`

Field Detail

<code>id</code>

```
private long id
```

<code>role</code>

cnat

```
private Chat chat
```

canRead

```
private boolean canRead
```

canWrite

```
private boolean canWrite
```

Constructor Detail

Permission

```
public Permission(Role role,  
                  Chat chat,  
                  boolean canRead,  
                  boolean canWrite)
```

Permission

```
public Permission()
```

Method Detail

hashCode

```
public int hashCode()
```

equals

```
public boolean equals(Object obj)
```

Diese Methode vergleicht nur die `id` beider Objekte.

Overrides:

`equals` in class `Object`

getId

```
public long getId()
```

getRole

```
public Role getRole()
```

setRole

```
public void setRole(Role role)
```

setId

```
public void setId(long id)
```

getChat

```
public Chat getChat()
```

setChat

```
public void setChat(Chat chat)
```

```
public boolean isCanRead()
```

setCanRead

```
public void setCanRead(boolean canRead)
```

isCanWrite

```
public boolean isCanWrite()
```

setCanWrite

```
public void setCanWrite(boolean canWrite)
```

Package com.realet.sip**Class PermissionsFacade**

`java.lang.Object`
`com.realet.sip.PermissionsFacade`

`public class PermissionsFacade
extends Object`

Manager-Klasse für Permissions.

Field Summary**Fields**

Modifier and Type	Field	Description
(package private) static <code>javax.persistence.EntityManagerFactory</code>	<code>emf</code>	Die EntityManagerFactory, die für alle Operationen von <code>PermissionsFacade</code> verwendet wird.

Constructor Summary**Constructors**

Constructor	Description
<code>PermissionsFacade()</code>	

Method Summary

[All Methods](#) [Static Methods](#) [Concrete Methods](#)

Modifier and Type	Method	Description
static void	<code>add(Permission permission)</code>	
static	<code>findById(long id) Optional<Permission></code>	
static	<code>findByRoleAndChat(Role role, Chat chat) Optional<Permission></code>	
static	<code>findGroupChatPermissions(long chatId,</code>	Findet eine Liste

einem Chat einer Group hinreichend beschreiben.

static void	initialize (javax.persistence.EntityManagerFactory emf)	Setzt emf.
static void	remove(Permission permission)	
static void	update(Permission permission)	

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#)

Field Detail

emf

static javax.persistence.EntityManagerFactory emf

Die EntityManagerFactory, die für alle Operationen von [PermissionsFacade](#) verwendet wird.

Constructor Detail

PermissionsFacade

public PermissionsFacade()

Method Detail

initialize

findById

```
public static Optional<Permission> findById(long id)
```

findGroupChatPermissions

```
public static List<Permission> findGroupChatPermissions(long chatId, long userId)
throws IllegalArgumentException
```

Findet eine Liste von maximal 3 Permissions, die die Berechtigung eines Nutzers zu einem Chat einer Group hinreichend beschreiben.

Throws:

`IllegalArgumentException`

findByRoleAndChat

```
public static Optional<Permission> findByRoleAndChat(Role role, Chat chat)
```

add

```
public static void add(Permission permission)
```

update

```
public static void update(Permission permission)
```

remove

```
public static void remove(Permission permission)
```

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

ALL CLASSES

SEARCH:

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Package com.realet.sip

Class PermissionsResource

`java.lang.Object`
`com.realet.sip.PermissionsResource`

```
@Path("/permissions")
public class PermissionsResource
extends Object
```

RestEasy Resource-Klasse für Permissions.

Constructor Summary

Constructors

Constructor	Description
<code>PermissionsResource()</code>	

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
<code>javax.ws.rs.core.Response</code>	<code>deletePermission (long permissionId, String token)</code>	Entfernt eine Permission anhand ihrer <code>Permission.id</code> .
<code>javax.ws.rs.core.Response</code>	<code>updatePermission (Permission permission, String token)</code>	Aktualisiert eine Permission.

Methods inherited from class java.lang.Object

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait`

Constructor Detail

Method Detail

updatePermission

```
@PUT @Consumes("application/json")
public javax.ws.rs.core.Response updatePermission(Permission permission,
@HeaderParam("Authorization") String token)
```

Aktualisiert eine `Permission`. `Permission.chat` und `Permission.role` werden in jedem Fall beibehalten.

Parameters:

`permission` -

`token` -

Returns:

Status Code 200 Status Code 404, falls keine `Permission` mit derselben `Permission.id` gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt.

deletePermission

```
@DELETE @Path("/{permissionId}")
public javax.ws.rs.core.Response deletePermission(@PathParam("permissionId") long permissionId, @HeaderParam("Authorization") String token)
```

Entfernt eine `Permission` anhand ihrer `Permission.id`.

Parameters:

`permissionId` -

`token` -

Returns:

Status Code 200, Status Code 404, falls keine `Permission` mit dieser `Permission.id` existiert, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status Code 400, falls die `Permission` keine `Permission.role` hat. In diesem Fall ist die `Permission` die Grundregel für diesen Chat und kann nicht gelöscht werden.

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

ALL CLASSES

SEARCH:

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Copyright © 2021. All rights reserved.

Package com.realet.sip**Class Role**

```
java.lang.Object
  com.realet.sip.Role
```

```
@Entity
public class Role
extends Object
```

Eine **Role** gehört zu einer **Group**. **Roles** regulieren Zugriff auf **Chats**.

Definiert eine NamedQuery "Role.findGroupRolesOrderedByPriority", welche alle **Roles** einer **Group** findet und aufsteigend nach **priority** sortiert. Akzeptiert einen Parameter: "group".

Definiert eine NamedQuery "Role.findUserGroupRoles", welche alle **Roles** eines **Users** in einer **Group** findet. Akzeptiert zwei Parameter: "group", "user".

Definiert eine NamedQuery "Role.findAdminRolesByUserAndGroup", welche alle **Roles** eines **Users** in einer **Group**, die **admin==true** haben, findet. Akzeptiert zwei Parameter: "group", "user".

Ein JPA-Entity für eine Tabelle names "Roles".

Field Summary**Fields**

Modifier and Type	Field	Description
private boolean	admin	
private String	color	Hex code.
private Group	group	
private long	id	
private String	name	
private List<Permission>	permissions	
private long	priority	
private Set<User>	users	

Constructor Summary**Constructors**

Constructor	Description
Role()	

Method Summary

[All Methods](#) [Instance Methods](#) [Concrete Methods](#)

Modifier and Type	Method	Description
boolean	<code>equals(Object obj)</code>	Diese Methode vergleicht nur die <code>id</code> beider Objekte.
<code>String</code>	<code>getColor()</code>	
<code>Group</code>	<code>getGroup()</code>	
long	<code>getId()</code>	
<code>String</code>	<code>getName()</code>	
<code>List<Permission></code>	<code>getPermissions()</code>	
long	<code>getPriority()</code>	
<code>Set<User></code>	<code>getUsers()</code>	
int	<code>hashCode()</code>	
boolean	<code>isAdmin()</code>	
void	<code>setAdmin(boolean admin)</code>	
void	<code>setColor(String color)</code>	
void	<code>setGroup(Group group)</code>	
void	<code>setId(long id)</code>	
void	<code>setName(String name)</code>	
void	<code>setPermissions (List<Permission> permissions)</code>	
void	<code>setPriority(long priority)</code>	
void	<code>setUsers(Set<User> users)</code>	

Methods inherited from class `java.lang.Object`

`clone, finalize, getClass, notify, notifyAll, toString, wait, wait, wait`

Field Detail

name

```
private String name
```

color

```
private String color
```

Hex code. Ex.: "#FFFFFF"

users

```
private Set<User> users
```

group

```
private Group group
```

admin

```
private boolean admin
```

priority

```
private long priority
```

permissions

```
private List<Permission> permissions
```

Role

```
public Role()
```

Role

```
public Role(String name,  
           String color,  
           Group group,  
           boolean admin,  
           long priority)
```

Method Detail**getId**

```
public long getId()
```

setId

```
public void setId(long id)
```

getName

```
public String getName()
```

setName

```
public void setName(String name)
```

setColor

```
public void setColor(String color)
```

getGroup

```
public Group getGroup()
```

setGroup

```
public void setGroup(Group group)
```

getUsers

```
public Set<User> getUsers()
```

setUsers

```
public void setUsers(Set<User> users)
```

hashCode

```
public int hashCode()
```

Overrides:

[hashCode](#) in class [Object](#)

equals

```
public boolean equals(Object obj)
```

getPermissions

```
public List<Permission> getPermissions()
```

setPermissions

```
public void setPermissions(List<Permission> permissions)
```

getPriority

```
public long getPriority()
```

setPriority

```
public void setPriority(long priority)
```

isAdmin

```
public boolean isAdmin()
```

setAdmin

```
public void setAdmin(boolean admin)
```

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

ALL CLASSES

SEARCH:

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Package com.realet.sip**Class RolesFacade**

`java.lang.Object`
`com.realet.sip.RolesFacade`

```
public class RolesFacade
extends Object
```

Manager-Klasse für Roles.

Field Summary**Fields**

Modifier and Type	Field	Description
(package private) static javax.persistence.EntityManagerFactory	<code>emf</code>	Die EntityManagerFactory, die für alle Operationen von <code>RolesFacade</code> verwendet wird.

Constructor Summary**Constructors**

Constructor	Description
<code>RolesFacade()</code>	

Method Summary

[All Methods](#) [Static Methods](#) [Concrete Methods](#)

Modifier and Type	Method	Description
static void	<code>add(Role role)</code>	
static <code>List<Role></code>	<code>findAdminRolesByUserAndGroup(User user,</code> <code>Group group)</code>	
static <code>Optional<Role></code>	<code>findById(long id)</code>	
static <code>List<Role></code>	<code>findGroupRolesOrderedByPriority(Group group)</code>	

```
static void update(Role role)
```

Methods inherited from class java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait,  
wait, wait
```

Field Detail

emf

```
static javax.persistence.EntityManagerFactory emf
```

Die EntityManagerFactory, die für alle Operationen von RolesFacade verwendet wird.

Constructor Detail

RolesFacade

```
public RolesFacade()
```

Method Detail

initialize

```
public static void initialize(javax.persistence.EntityManagerFactory emf)
```

Setzt emf.

findById

findGroupRolesOrderedByPriority

```
public static List<Role> findGroupRolesOrderedByPriority(Group group)
```

findUserGroupRoles

```
public static List<Role> findUserGroupRoles(Group group, User user)
```

findAdminRolesByUserAndGroup

```
public static List<Role> findAdminRolesByUserAndGroup(User user, Group group)
```

add

```
public static void add(Role role)
```

update

```
public static void update(Role role)
```

remove

```
public static void remove(Role role)
```

Package com.realet.sip**Class RolesResource**

`java.lang.Object`
`com.realet.sip.RolesResource`

```
@Path("/roles")
public class RolesResource
extends Object
```

RestEasy Resource-Klasse für `Roles`.

Constructor Summary**Constructors**

Constructor	Description
<code>RolesResource()</code>	

Method Summary**All Methods Instance Methods Concrete Methods**

Modifier and Type	Method	Description
<code>javax.ws.rs.core.Response addUser(long roleId,</code> <code> User inputUser, String token)</code>		Fügt einen <code>User</code> zu einer <code>Role</code> hinzu.
<code>javax.ws.rs.core.Response deleteRole(long roleId,</code> <code> String token)</code>		Entfernt eine <code>Role</code> anhand ihrer <code>Role.id</code> .
<code>javax.ws.rs.core.Response removeUser(long roleId,</code> <code> long userId, String token)</code>		Entfernt einen <code>User</code> aus einer <code>Group</code> .
<code>javax.ws.rs.core.Response updateRole(Role role,</code> <code> String token)</code>		Aktualisiert eine <code>Role</code> .

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`

RolesResource

```
public RolesResource()
```

Method Detail**updateRole**

```
@PUT @Consumes("application/json") public javax.ws.rs.core.Response updateRole  
(Role role, @HeaderParam("Authorization") String token)
```

Aktualisiert eine `Role`. `Role.priority` und `Role.group` werden in jedem Fall beibehalten. Entfernt Leerzeichen am Anfang und Ende des `Role.name`.

Parameters:

`role` -

`token` -

Returns:

Status Code 200 Status Code 404, falls keine `Role` mit derselben `Role.id` gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status code 400, falls die `Role.color` nicht vorhanden oder kein gültiger Hex-Code ist, oder der `Role.name` nicht vorhanden oder leer ist, oder nur aus Leerzeichen besteht.

deleteRole

```
@DELETE @Path("/{roleId}") public javax.ws.rs.core.Response deleteRole  
(@PathParam("roleId") long roleId, @HeaderParam("Authorization") String token)
```

Entfernt eine `Role` anhand ihrer `Role.id`.

Parameters:

`roleId` -

`token` -

Returns:

Status Code 200, Status Code 404, falls keine `Role` mit dieser `Role.id` existiert, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt.

[SUCHEN](#)

Fügt einen [User](#) zu einer [Role](#) hinzu. Diese Methode verändert den in der Datenbank gespeicherten [User](#) nicht.

Parameters:[roleId](#) -[inputUser](#) -[token](#) -**Returns:**

Status Code 200, Status Code 404, falls der [User](#) oder die [Role](#) nicht gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status code 400, falls der [User](#) bereits Teil dieser [Role](#) oder nicht Teil der [Role.group](#) ist.

removeUser

```
@DELETE @Path("/{roleId}/users/{userId}")
public javax.ws.rs.core.Response removeUser(@PathParam("roleId") long roleId,
@PathParam("userId") long userId, @HeaderParam("Authorization") String token)
```

Entfernt einen [User](#) aus einer [Group](#). Der [User](#) existiert weiterhin.

Parameters:[roleId](#) -[userId](#) -[token](#) -**Returns:**

Status Code 200, Status Code 404, falls die [Group](#) [Group](#) nicht gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status code 400, falls der [User](#) nicht Teil dieser [Role](#) ist.

Package com.realet.sip

Class ServiceApplication

```
java.lang.Object
    javax.ws.rs.core.Application
        com.realet.sip.ServiceApplication
```

```
@ApplicationPath("/")
public class ServiceApplication
extends javax.ws.rs.core.Application
```

Die HauptKlasse der Anwendung. Registriert alle RESTEasy Resource-Klassen und initialisiert alle Facade-Klassen.

Constructor Summary

Constructors

Constructor	Description
ServiceApplication()	

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
Set<Class<?>>	getClasses()	

Methods inherited from class javax.ws.rs.core.Application

[getProperties](#), [getSingletons](#)

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#)

```
public ServiceApplication()
```

Method Detail

getClasses

```
public Set<Class<?>> getClasses()
```

Overrides:

getClasses in class javax.ws.rs.core.Application

Package com.realet.sip**Class Session**

`java.lang.Object`
`com.realet.sip.Session`

```
@Entity
public class Session
extends Object
```

Die `Session`-Klasse modelliert eine Sitzung.

Definiert eine NamedQuery "Session.getAllUserSessionsExcept", welche alle Sessions eines User, die nicht ein bestimmtes token haben, findet. Akzeptiert zwei Parameter: "user", "token".
Ein JPA-Entity für eine Tabelle names "Sessions".

Field Summary**Fields**

Modifier and Type	Field	Description
private Date	<code>expires</code>	
private String	<code>token</code>	MD5-Hash, um Einzigartigkeit zu garantieren und das Erraten eines Tokens praktisch unmöglich zu machen.
private User	<code>user</code>	

Constructor Summary**Constructors**

Constructor	Description
<code>Session()</code>	
<code>Session(String token, User user, Date expires)</code>	

Method Summary

SUMMARY: NESTED | FIELD | CONSTR | METHOD

DETAIL: FIELD | CONSTR | METHOD

Diese Methode vergleicht nur das LOKALEN Beide Objekte.

Date	<code>getExpires()</code>
String	<code>getToken()</code>
User	<code>getUser()</code>
int	<code>hashCode()</code>
void	<code>setExpires (Date expires)</code>
void	<code>setUser(User user)</code>

Methods inherited from class java.lang.Object

`clone, finalize, getClass, notify, notifyAll, toString, wait, wait, wait`

Field Detail

token

`private String token`

MD5-Hash, um Einzigartigkeit zu garantieren und das Erraten eines Tokens praktisch unmöglich zu machen.

user

`private User user`

expires

`private Date expires`

Constructor Detail

Session

```
public Session(String token,  
               User user,  
               Date expires)
```

Method Detail**getUser**

```
public User getUser()
```

setUser

```
public void setUser(User user)
```

getExpires

```
public Date getExpires()
```

setExpires

```
public void setExpires(Date expires)
```

getToken

```
public String getToken()
```

Overrides:[hashCode](#) in class [Object](#)**equals**

```
public boolean equals(Object obj)
```

Diese Methode vergleicht nur das `token` beider Objekte.

Overrides:[equals](#) in class [Object](#)

Package com.realet.sip

Class SessionsFacade

`java.lang.Object`
`com.realet.sip.SessionsFacade`

```
public class SessionsFacade
extends Object
```

Manager-Klasse für Sessions.

Field Summary

Fields

Modifier and Type	Field	Description
(package private) static javax.persistence.EntityManagerFactory	<code>emf</code>	Die EntityManagerFactory, die für alle Operationen von SessionsFacade verwendet wird.

Constructor Summary

Constructors

Constructor	Description
<code>SessionsFacade()</code>	

Method Summary

[All Methods](#) [Static Methods](#) [Concrete Methods](#)

Modifier and Type	Method	Description
static void	<code>add(Session session)</code>	
static	<code>findByToken(String token)</code> <code>Optional<Session></code>	
static long	<code>findUserIdByToken(String token)</code>	Findet eine User.id eines Session.user einer Session mit

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

```
static void      remove(Session session)
```

```
static void      removeAllUserSessionsExcept(String token,  
                                              User user)
```

Entfernt alle Sessions mit einem bestimmten Session.user, außer eine Session mit einem bestimmten Session.token.

Methods inherited from class java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait,  
wait, wait
```

Field Detail

emf

```
static javax.persistence.EntityManagerFactory emf
```

Die EntityManagerFactory, die für alle Operationen von SessionsFacade verwendet wird.

Constructor Detail

SessionsFacade

```
public SessionsFacade()
```

Method Detail

initialize

findByToken

```
public static Optional<Session> findByToken(String token)
```

findUserIdByToken

```
public static long findUserIdByToken(String token) throws IllegalAccessException
```

Findet eine `User.id` eines `Session.user` einer `Session` mit einem bestimmten `Session.token`.

Throws:

`IllegalAccessException` - Falls keine Session mit dem gegebenen Token gefunden werden kann.

removeAllUserSessionsExcept

```
public static void removeAllUserSessionsExcept(String token, User user)
```

Entfernt alle Sessions mit einem bestimmten `Session.user`, außer einer `Session` mit einem bestimmten `Session.token`.

add

```
public static void add(Session session)
```

remove

```
public static void remove(Session session)
```

Package com.realet.sip**Class User**

```
java.lang.Object
com.realet.sip.User
```

```
@Entity
public class User
extends Object
```

Die `User`-Klasse repräsentiert einen Nutzer der Applikation. Ein `User` enthält die Anmelde-, sowie Anzeigeeinformationen eines Nutzers.

Definiert eineNamedQuery "User.findByEmail", welche einen `User` anhand seiner `email` findet. Akzeptiert einen Parameter: "email".

Definiert eine NamedNativeQuery "User.findBasicGroupMembers", welche alle `User` ohne `Role` einer `Group` findet. Akzeptiert einen Parameter: 1: `Group.id`.

Definiert eine NamedNativeQuery "User.findGroupChatReaders", welche alle `User`, die einen bestimmten `Chat` lesen können, findet. Akzeptiert einen Parameter: 1: `Chat.id`.

Ein JPA-Entity für eine Tabelle names "Users".

Field Summary**Fields**

Modifier and Type	Field	Description
private Set<User>	<code>blockedBy</code>	
private Set<User>	<code>blockedUsers</code>	
private List<ChatMessage>	<code>chatMessages</code>	
private List<Chat>	<code>directChats1</code>	
private List<Chat>	<code>directChats2</code>	
private String	<code>email</code>	
private Set<Group>	<code>groups</code>	
private long	<code>id</code>	
private String	<code>info</code>	
private boolean	<code>isOnline</code>	
private Date	<code>lastSeen</code>	
private List<Group>	<code>ownedGroups</code>	
private String	<code>pass</code>	Bcrypt-Hash mit 10 Runden.

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

private String**status****private String****username**

Kann keine Leerzeichen enthalten.

private List<UserProfile>**userProfiles**

Constructor Summary

Constructors

Constructor	Description
User()	
User(String username, String email, String pass, String info, String status, Date lastSeen, boolean isOnline, String profilePicture)	

Method Summary

All Methods **Instance Methods** **Concrete Methods**

Modifier and Type	Method	Description
boolean	equals(Object obj)	Diese Methode vergleicht nur die <code>id</code> beider Objekte.
Set<User>	getBlockedBy()	
Set<User>	getBlockedUsers()	
List<ChatMessage>	getChatMessages()	
List<Chat>	getDirectChats1()	
List<Chat>	getDirectChats2()	
String	getEmail()	
Set<Group>	getGroups()	
long	getId()	
String	 getInfo()	
Date	getLastSeen()	
List<Group>	getOwnedGroups()	
String	getPass()	
String	getProfilePicture()	

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

String **getUsername()**

List<UserProfile> **getUserProfiles()**

int **hashCode()**

boolean **isOnline()**

void **setBlockedBy**
(Set<User> blockedBy)

void **setBlockedUsers**
(Set<User> blockedUsers)

void **setChatMessages**
(List<ChatMessage> chatMessages)

void **setDirectChats1**
(List<Chat> directChats1)

void **setDirectChats2**
(List<Chat> directChats2)

void **setEmail(String email)**

void **setGroups(Set<Group> groups)**

void **setId(long id)**

void **setInfo(String info)**

void **setLastSeen(Date lastSeen)**

void **setOnline(boolean isOnline)**

void **setOwnedGroups**
(List<Group> ownedGroups)

void **setPass(String pass)**

void **setProfile_picture**
(String profilePicture)

void **setProfilePicture**
(String profilePicture)

void **setRoles(Set<Role> roles)**

void **setSessions**
(List<Session> sessions)

void **setStatus(String status)**

void **setUsername(String username)**

void **setUserProfiles**
(List<UserProfile> userProfiles)

Field Detail**id**

```
private long id
```

username

```
private String username
```

Kann keine Leerzeichen enthalten.

email

```
private String email
```

pass

```
private String pass
```

Bcrypt-Hash mit 10 Runden.

info

```
private String info
```

status

```
private String status
```

isOnline

```
private boolean isOnline
```

profilePicture

```
private String profilePicture
```

Dateiname des Bilds ohne Pfad oder Endung.

userProfiles

```
private List<UserProfile> userProfiles
```

blockedUsers

```
private Set<User> blockedUsers
```

blockedBy

```
private Set<User> blockedBy
```

groups

```
private Set<Group> groups
```

roles

```
private Set<Role> roles
```

directChats1

```
private List<Chat> directChats1
```

directChats2

```
private List<Chat> directChats2
```

chatMessages

```
private List<ChatMessage> chatMessages
```

sessions

```
private List<Session> sessions
```

Constructor Detail**User**

```
public User()
```

User

```
public User(String username,  
           String email,  
           String pass,  
           String info,  
           String status,
```

Method Detail**getId**

```
public long getId()
```

setId

```
public void setId(long id)
```

getUsername

```
public String getUsername()
```

setUsername

```
public void setUsername(String username)
```

 getInfo

```
public String getInfo()
```

setInfo

```
public void setInfo(String info)
```

getProfilePicture

SUMMARY: NESTED | FIELD | CONSTR | METHOD

DETAIL: FIELD | CONSTR | METHOD

setProfile_picture

```
public void setProfile_picture(String profilePicture)
```

getEmail

```
public String getEmail()
```

setEmail

```
public void setEmail(String email)
```

getPass

```
public String getPass()
```

setPass

```
public void setPass(String pass)
```

getGroups

```
public Set<Group> getGroups()
```

setGroups

```
public void setGroups(Set<Group> groups)
```

getRoles

setRoles

```
public void setRoles(Set<Role> roles)
```

getOwnedGroups

```
public List<Group> getOwnedGroups()
```

setOwnedGroups

```
public void setOwnedGroups(List<Group> ownedGroups)
```

getDirectChats1

```
public List<Chat> getDirectChats1()
```

setDirectChats1

```
public void setDirectChats1(List<Chat> directChats1)
```

getDirectChats2

```
public List<Chat> getDirectChats2()
```

setDirectChats2

```
public void setDirectChats2(List<Chat> directChats2)
```

getChatMessages

setChatMessages

```
public void setChatMessages(List<ChatMessage> chatMessages)
```

getSessions

```
public List<Session> getSessions()
```

setSessions

```
public void setSessions(List<Session> sessions)
```

getStatus

```
public String getStatus()
```

setStatus

```
public void setStatus(String status)
```

getLastSeen

```
public Date getLastSeen()
```

setLastSeen

```
public void setLastSeen(Date lastSeen)
```

isOnline

setOnline

```
public void setOnline(boolean isOnline)
```

setProfilePicture

```
public void setProfilePicture(String profilePicture)
```

getBlockedUsers

```
public Set<User> getBlockedUsers()
```

setBlockedUsers

```
public void setBlockedUsers(Set<User> blockedUsers)
```

getBlockedBy

```
public Set<User> getBlockedBy()
```

setBlockedBy

```
public void setBlockedBy(Set<User> blockedBy)
```

hashCode

```
public int hashCode()
```

Overrides:

[hashCode](#) in class [Object](#)

setUserProfiles

```
public void setUserProfiles(List<UserProfile> userProfiles)
```

equals

```
public boolean equals(Object obj)
```

Diese Methode vergleicht nur die `id` beider Objekte.

Overrides:

`equals` in class `Object`

Package com.realet.sip**Class UserProfile**

`java.lang.Object`
`com.realet.sip.UserProfile`

```
@Entity
public class UserProfile
extends Object
```

Ein `UserProfile` gehört zu einem `User` und speichert persönliche Konfigurationen des Nutzers.
 Ein JPA-Entity für eine Tabelle names "UserProfiles".

Field Summary**Fields**

Modifier and Type	Field	Description
private	<code>id</code> <code>Long</code>	
private	<code>maxLoadedMessages</code> <code>int</code>	Anweisung an die Client-Anwendung, wie viele Nachrichten gleichzeitig geladen werden sollten.
private	<code>messageChunkSize</code> <code>int</code>	Anweisung an die Client-Anwendung, wie viele neue Nachrichten geladen werden sollten, wenn der Nutzer zum Ende der geladenen Nachrichten gescrollt hat.
private	<code>reverseBlocking</code> <code>boolean</code>	Anweisung an die Client-Anwendung, ob Nachrichten von <code>Usern</code> , die den eigenen <code>User</code> blockiert haben, gezeigt werden sollten.
private	<code>user</code> <code>User</code>	

Constructor Summary**Constructors**

Constructor	Description
<code>UserProfile()</code>	
<code>UserProfile(User user, boolean reverseBlocking, int maxLoadedMessages,</code>	

All Methods [Instance Methods](#) [Concrete Methods](#)

Modifier and Type	Method	Description
Long	<code>getId()</code>	
int	<code>getMaxLoadedMessages()</code>	
int	<code>getMessageChunkSize()</code>	
User	<code>getUser()</code>	
boolean	<code>isReverseBlocking()</code>	
void	<code>setId(Long id)</code>	
void	<code>setMaxLoadedMessages(int maxLoadedMessages)</code>	
void	<code>setMessageChunkSize(int messageChunkSize)</code>	
void	<code>setReverseBlocking(boolean reverseBlocking)</code>	
void	<code>setUser(User user)</code>	

Methods inherited from class java.lang.Object

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait`

Field Detail**id**

```
private Long id
```

user

```
private User user
```

reverseBlocking

maxLoadedMessages

```
private int maxLoadedMessages
```

Anweisung an die Client-Anwendung, wie viele Nachrichten gleichzeitig geladen werden sollten.

messageChunkSize

```
private int messageChunkSize
```

Anweisung an die Client-Anwendung, wie viele neue Nachrichten geladen werden sollten, wenn der Nutzer zum Ende der geladenen Nachrichten gescrollt hat.

Constructor Detail**UserProfile**

```
public UserProfile(User user,
                  boolean reverseBlocking,
                  int maxLoadedMessages,
                  int messageChunkSize)
```

UserProfile

```
public UserProfile()
```

Method Detail**getUser**

```
public User getUser()
```

isReverseBlocking

```
public boolean isReverseBlocking()
```

setReverseBlocking

```
public void setReverseBlocking(boolean reverseBlocking)
```

getMaxLoadedMessages

```
public int getMaxLoadedMessages()
```

setMaxLoadedMessages

```
public void setMaxLoadedMessages(int maxLoadedMessages)
```

getMessageChunkSize

```
public int getMessageChunkSize()
```

getId

```
public Long getId()
```

setId

```
public void setId(Long id)
```

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

ALL CLASSES

SEARCH:

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Copyright © 2021. All rights reserved.

Package com.realet.sip

Class UserProfilesFacade

`java.lang.Object`
`com.realet.sip.UserProfilesFacade`

```
public class UserProfilesFacade
extends Object
```

Manager-Klasse für `UserProfiles`.

Field Summary

Fields

Modifier and Type	Field	Description
(package private) static javax.persistence.EntityManagerFactory	<code>emf</code>	Die EntityManagerFactory, die für alle Operationen von <code>UserProfilesFacade</code> verwendet wird.

Constructor Summary

Constructors

Constructor	Description
<code>UserProfilesFacade()</code>	

Method Summary

[All Methods](#) [Static Methods](#) [Concrete Methods](#)

Modifier and Type	Method	Description
static void	<code>add(UserProfile userProfile)</code>	
static <code>Optional<UserProfile></code>	<code>findById(long id)</code>	
static void	<code>initialize</code> (<code>javax.persistence.EntityManagerFactory emf</code>)	Setzt <code>emf</code> .
static void	<code>remove(UserProfile userProfile)</code>	

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait,  
wait, wait
```

Field Detail

emf

```
static javax.persistence.EntityManagerFactory emf
```

Die EntityManagerFactory, die für alle Operationen von [UserProfilesFacade](#) verwendet wird.

Constructor Detail

UserProfilesFacade

```
public UserProfilesFacade()
```

Method Detail

initialize

```
public static void initialize(javax.persistence.EntityManagerFactory emf)
```

Setzt emf.

findById

```
public static Optional<UserProfile> findById(long id)
```

update

```
public static void update(UserProfile userProfile)
```

remove

```
public static void remove(UserProfile userProfile)
```

Package com.realet.sip

Class UserProfilesResource

java.lang.Object
com.realet.sip.UserProfilesResource

```
@Path("/user-profiles")
public class UserProfilesResource
extends Object
```

RestEasy Resource-Klasse für UserProfiles.

Constructor Summary

Constructors

Constructor	Description
UserProfilesResource()	

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
javax.ws.rs.core.Response	updateUserProfile (UserProfile userProfile, String token)	Aktualisiert ein UserProfile.

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail

UserProfilesResource

Method Detail

updateUserProfile

```
@PUT @Consumes("application/json")
public javax.ws.rs.core.Response updateUserProfile(UserProfile userProfile,
@HeaderParam("Authorization") String token)
```

Aktualisiert ein UserProfile.

Parameters:

userProfile -

token -

Returns:

Status Code 200, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status code 400, falls UserProfile.maxLoadedMessages < UserProfile.messageChunkSize ist oder die UserProfile#id nicht mit der UserProfile.id des UserProfile des UserProfile.user übereinstimmt.

Package com.realet.sip**Class UsersFacade**

`java.lang.Object`
`com.realet.sip.UsersFacade`

```
public class UsersFacade
extends Object
```

Manager-Klasse für `Users`.

Field Summary**Fields**

Modifier and Type	Field	Description
(package private) static <code>javax.persistence.EntityManagerFactory</code>	<code>emf</code>	Die EntityManagerFactory, die für alle Operationen von <code>UsersFacade</code> verwendet wird.

Constructor Summary**Constructors**

Constructor	Description
<code>UsersFacade()</code>	

Method Summary

[All Methods](#) [Static Methods](#) [Concrete Methods](#)

Modifier and Type	Method	Description
static void	<code>add(User user)</code>	
static <code>List<User></code>	<code>findAll()</code>	
static <code>List<User></code>	<code>findBasicGroupMembers(long groupId)</code>	Findet alle <code>User</code> einer <code>Group</code> , die in dieser

```
static      findById(long id)
```

```
Optional<User>
```

```
static      findByName(String name)
```

```
List<User>
```

```
static      findGroupChatReaders(long chatId,  
List<User>      long groupId)
```

Findet alle [User](#), die
einen bestimmten Chat
lesen können.

```
static void   initialize  
              (javax.persistence.EntityManagerFactory emf)
```

Setzt [emf](#).

```
static void   remove(User user)
```

```
static void   update(User user)
```

Methods inherited from class [java.lang.Object](#)

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait,  
wait, wait
```

Field Detail

```
emf
```

```
static javax.persistence.EntityManagerFactory emf
```

Die EntityManagerFactory, die für alle Operationen von [UsersFacade](#) verwendet wird.

Constructor Detail

```
UsersFacade
```

```
public UsersFacade()
```

```
public static void initialize(javax.persistence.EntityManagerFactory emf)
```

Setzt emf.

findById

```
public static Optional<User> findById(long id)
```

findAll

```
public static List<User> findAll()
```

findByName

```
public static List<User> findByName(String name)
```

findByEmail

```
public static Optional<User> findByEmail(String email)
```

findGroupChatReaders

```
public static List<User> findGroupChatReaders(long chatId, long groupId)
```

Findet alle User, die einen bestimmten Chat lesen können.

findBasicGroupMembers

```
public static List<User> findBasicGroupMembers(long groupId)
```

Findet alle User einer Group, die in dieser Group kein Teil einer Role sind.

update

```
public static void update(User user)
```

remove

```
public static void remove(User user)
```

Package com.realet.sip**Class UsersResource**

`java.lang.Object`
`com.realet.sip.UsersResource`

```
@Path("/users")
public class UsersResource
extends Object
```

RestEasy Resource-Klasse für `Users`.

Constructor Summary**Constructors**

Constructor	Description
<code>UsersResource()</code>	

Method Summary**All Methods Instance Methods Concrete Methods**

Modifier and Type	Method	Description
<code>javax.ws.rs.core.Response addBlockedUser (long userId, User input, String token)</code>		Fügt einen <code>User</code> zu <code>User.blockedUsers</code> eines anderen <code>Users</code> hinzu.
<code>javax.ws.rs.core.Response addUser(User user)</code>		Erstellt einen <code>User</code> .
<code>javax.ws.rs.core.Response getBlockedBy (long userId, String token)</code>		Findet alle <code>User</code> , von denen ein <code>User</code> blockiert ist.
<code>javax.ws.rs.core.Response getBlockedUsers (long userId, String token)</code>		Findet alle <code>User</code> , die ein <code>User</code> blockiert hat.
<code>javax.ws.rs.core.Response getDirectChats (long userId, String token)</code>		Findet alle Direkt-Unterhaltungen eines <code>Users</code> .
<code>javax.ws.rs.core.Response getGroups(long userId, String token)</code>		Findet alle <code>Groups</code> eines <code>Users</code> .

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD
`String token)`

`javax.ws.rs.core.Response getSharedGroups
 (long user1Id,
 long user2Id,
 String token)`

Findet alle gemeinsamen Groups zweier User.

`javax.ws.rs.core.Response getUser(long id)`

Findet einen User.

`javax.ws.rs.core.Response removeBlockedUser
 (long userId,
 long blockedUserId,
 String token)`

Entfernt einen User aus User.blockedUsers eines anderen Users.

`javax.ws.rs.core.Response updateUser(long userId,
 String token)`

Löscht einen User und alle Groups von denen er der Group.owner ist, sowie seine Sessions.

`javax.ws.rs.core.Response updateUser(User user,
 String token)`

Aktualisiert einen User.

Methods inherited from class java.lang.Object

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait`

Constructor Detail

UsersResource

`public UsersResource()`

Method Detail

getUser

```
@GET @Path("/{id}") @Produces("text/plain")
public javax.ws.rs.core.Response getUser(@PathParam("id") long id)
```

Status Code 200 mit [User](#), serialisiert durch [UserAdapter](#), Status Code 404, falls der [User](#) nicht gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt.

getDirectChats

```
@GET @Path("/{userId}/direct-chats") @Produces("application/json")
public javax.ws.rs.core.Response getDirectChats(@PathParam("userId") long userId,
@HeaderParam("Authorization") String token)
```

Findet alle Direkt-Unterhaltungen eines [Users](#).

Parameters:

`userId` -

`token` -

Returns:

Status Code 200 mit [Chats](#), serialisiert durch [ChatAdapter](#), Status Code 404, falls der [User](#) nicht gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt.

getGroups

```
@GET @Path("/{userId}/groups") @Produces("application/json")
public javax.ws.rs.core.Response getGroups(@PathParam("userId") long userId,
@HeaderParam("Authorization") String token)
```

Findet alle [Groups](#) eines [Users](#).

Parameters:

`userId` -

`token` -

Returns:

Status Code 200 mit [Groups](#), serialisiert durch [GroupAdapter](#), Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt.

getSharedGroups

```
@GET @Path("/{user1Id}/shared-groups/{user2Id}") @Produces("application/json")
public javax.ws.rs.core.Response getSharedGroups(@PathParam("user1Id")
```

user1Id -

user2Id -

token -

Returns:

Status Code 200 mit [Groups](#), serialisiert durch [GroupAdapter](#), Status Code 404, falls einer der beiden [User](#) nicht gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt, Status Code 400, falls beide [User.id](#) identisch sind.

getBlockedUsers

```
@GET @Path("/{userId}/blocked-users") @Produces("application/json")
public javax.ws.rs.core.Response getBlockedUsers(@PathParam("userId") long userId,
@HeaderParam("Authorization") String token)
```

Findet alle [User](#), die ein [User](#) blockiert hat.

Parameters:

userId -

token -

Returns:

Status Code 200 mit [Usern](#), serialisiert durch [UserAdapter](#), Status Code 404, falls der [User](#) nicht gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt.

addBlockedUser

```
@POST @Path("/{userId}/blocked-users")
public javax.ws.rs.core.Response addBlockedUser(@PathParam("userId") long userId,
User input, @HeaderParam("Authorization") String token)
```

Fügt einen [User](#) zu [User.blockedUsers](#) eines anderen [Users](#) hinzu. Informiert alle WebSockets, die für diesen [User](#) registriert sind.

Parameters:

userId -

token -

Returns:

Status Code 200, Status Code 404, falls einer (oder beide) der [User](#) nicht gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt.

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

```
public javax.ws.rs.core.Response removeBlockedUser(@PathParam("userId")
long userId, @PathParam("blockedUserId") long blockedUserId,
@HeaderParam("Authorization") String token)
```

Entfernt einen **User** aus **User.blockedUsers** eines anderen **Users**. Informiert alle WebSockets, die für diesen **User** registriert sind.

Parameters:**userId** -**token** -**Returns:**

Status Code 200, Status Code 404, falls einer (oder beide) der **User** nicht gefunden werden konnte oder der **User** nicht blockiert ist, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt.

getBlockedBy

```
@GET @Path("/{userId}/blocked-by") @Produces("application/json")
public javax.ws.rs.core.Response getBlockedBy(@PathParam("userId") long userId,
@HeaderParam("Authorization") String token)
```

Findet alle **User**, von denen ein **User** blockiert ist.

Parameters:**userId** -**token** -**Returns:**

Status Code 200 mit **Users**, serialisiert durch **UserAdapter**, Status Code 404, falls der **User** nicht gefunden werden konnte, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt.

getRoles

```
@GET @Path("/{userId}/roles") @Produces("application/json")
public javax.ws.rs.core.Response getRoles(@PathParam("userId") long userId,
@QueryParam("group") long groupId, @HeaderParam("Authorization") String token)
```

Findet alle **Roles** eines **Users** in einer **Group**.

Parameters:**userId** -**token** -

addUser

```
@POST @Consumes("application/json") public javax.ws.rs.core.Response addUser(User user)
```

Erstellt einen [User](#). Kann optional außerdem ein [UserProfile](#) erstellen.

Parameters:

[user](#) -

Returns:

Status Code 201, Status Code 400, falls [User.username](#) nicht vorhanden oder leer ist, oder nur aus Leerzeichen besteht, ein [UserProfile](#) gegeben ist, bei dem [UserProfile.maxLoadedMessages](#) kleiner ist als [UserProfile.messageChunkSize](#) oder ein anderer [User](#) diese [User.email](#) bereits verwendet.

getProfile

```
@GET @Consumes("application/json") @Path("/{userId}/profile")  
public javax.ws.rs.core.Response getProfile(@PathParam("userId") long userId,  
@HeaderParam("Authorization") String token)
```

Findet das [UserProfile](#) eines [Users](#).

Parameters:

[userId](#) -

[token](#) -

Returns:

Status Code 200 mit [UserProfile](#), serialisiert durch [UserProfileAdapter](#), Status Code 403, falls das [token](#) ungültig ist oder keinen Zugriff auf diese Ressource erlaubt.

updateUser

```
@PUT @Consumes("application/json") public javax.ws.rs.core.Response updateUser(User user, @HeaderParam("Authorization") String token)
```

Aktualisiert einen [User](#). [User.email](#), [User.lastSeen](#) und [User.isOnline](#) werden in jedem Fall beibehalten. [User.pass](#) wird beibehalten, falls kein neues Passwort angegeben ist.

Parameters:

[user](#) -

updateUser

```
@DELETE @Path("/{userId}") @Consumes("application/json")
public javax.ws.rs.core.Response updateUser(@PathParam("userId") long userId,
@HeaderParam("Authorization") String token)
```

Löscht einen User und alle Groups von denen er der Group.owner ist, sowie seine Sessions.

Parameters:

userId -

token -

Returns:

Status Code 200, Status Code 403, falls das token ungültig ist oder keinen Zugriff auf diese Ressource erlaubt.

Package com.realet.sip

Class UserWebSocketManagement

`java.lang.Object`
`com.realet.sip.UserWebSocketManagement`

```
public class UserWebSocketManagement
extends Object
```

Manager-Klasse für Session-Objekte, die für einen [User](#) registriert sind.

Field Summary

Fields

Modifier and Type	Field	Description
<code>private static HashMap<Long,</code> <code>ArrayList<javax.websocket.Session>></code>	<code>sessions</code>	Format: userId => Session

Constructor Summary

Constructors

Constructor	Description
<code>UserWebSocketManagement()</code>	

Method Summary

All Methods Static Methods Concrete Methods

Modifier and Type	Method	Description
<code>static void</code>	<code>addSession(long userId,</code> <code>javax.websocket.Session session)</code>	
<code>static</code> <code>ArrayList<javax.websocket.Session></code>	<code>getSessions(long userId)</code>	
<code>static void</code>	<code>removeSession(long userId,</code> <code>javax.websocket.Session session)</code>	

Field Detail

sessions

```
private static HashMap<Long,ArrayList<javax.websocket.Session>> sessions
```

Format: userId => Session

Constructor Detail

UserWebsocketManagement

```
public UserWebsocketManagement()
```

Method Detail

getSessions

```
public static ArrayList<javax.websocket.Session> getSessions(long userId)
```

addSession

```
public static void addSession(long userId, javax.websocket.Session session)
```

removeSession

```
public static void removeSession(long userId, javax.websocket.Session session)
```

ALL CLASSES

SEARCH:

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Copyright © 2021. All rights reserved.

Package com.realet.sip**Class UserWebSocketResource**

`java.lang.Object`
`com.realet.sip.UserWebSocketResource`

```
public class UserWebSocketResource
extends Object
```

Diese Klasse akzeptiert WebSockets-Verbindungen nach "/users/{userId}/websockets". Durch diese wird übermittelt, wenn eine neue Nachricht, welche für diesen Nutzer lesbar ist gesendet wurde, sowie wenn ein Nutzer blockiert wurde oder eine Blockade aufgehoben wurde.

Dies geschieht in folgendem Format:

Neuer Nutzer blockiert: "blocked: <User als JSON>"

Blockade aufgehoben: "unblocked: <User als JSON>"

Neue Nachricht: "new-message: <ChatMessage als JSON>"

Constructor Summary**Constructors**

Constructor	Description
<code>UserWebSocketResource()</code>	

Method Summary**All Methods** **Instance Methods** **Concrete Methods**

Modifier Method and Type	Description
<code>void onClose(javax.websocket.Session session, javax.websocket.CloseReason closeReason, long userId)</code>	Entfernt die WebSockets-Session aus dem <code>UserWebSocketManagement</code> .
<code>void onError(javax.websocket.Session session, Throwable throwable, long userId)</code>	Entfernt die WebSockets-Session aus dem <code>UserWebSocketManagement</code> .
<code>void OnMessage (javax.websocket.Session session, String message, long userId)</code>	Überprüft Gültigkeit und Vorrechte einer im Format "Bearer <token>" übermittelten <code>Session</code> und registriert die WebSockets-Session im <code>UserWebSocketManagement</code> .

Methods inherited from class java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait,  
wait, wait
```

Constructor Detail**UserWebsocketResource**

```
public UserWebsocketResource()
```

Method Detail**onOpen**

```
public void onOpen(javax.websocket.Session session, long userId)
```

Ueberprueft, ob der jeweilige **User** existiert und sendet andernfalls eine entsprechende Nachricht, woraufhin die Verbindung geschlossen wird.

OnMessage

```
public void OnMessage(javax.websocket.Session session, String message,  
long userId)
```

Ueberprueft Gueltigkeit und Vorrechte einer im Format "Bearer <token>" übermittelten **Session** und registriert die WebSockets-Session im **UserWebsocketManagement**. Sendet gegebenenfalls eine Fehlermeldung zurück.

onClose

```
public void onClose(javax.websocket.Session session,  
javax.websocket.CloseReason closeReason, long userId)
```

```
public void onError(javax.websocket.Session session, Throwable throwable,  
long userId)
```

Entfernt die WebSockets-Session aus dem `UserWebsocketManagement`.

Package com.realet.sip.GsonTypeAdapter

Class Summary

Class	Description
ChatAdapter	Ein Serialisierungs-Gson-TypeAdapter für die Chat Klasse.
ChatMessageAdapter	Ein Serialisierungs-Gson-TypeAdapter für die ChatMessage Klasse.
GroupAdapter	Ein Serialisierungs-Gson-TypeAdapter für die Group Klasse.
PermissionAdapter	Ein Serialisierungs-Gson-TypeAdapter für die Permission Klasse.
RoleAdapter	Ein Serialisierungs-Gson-TypeAdapter für die Role Klasse.
UserAdapter	Ein Serialisierungs-Gson-TypeAdapter für die User Klasse.
UserProfileAdapter	Ein Serialisierungs-Gson-TypeAdapter für die UserProfile Klasse.

Package com.realet.sip.GsonTypeAdapter

Class ChatAdapter

```
java.lang.Object
  com.google.gson.TypeAdapter<Chat>
    com.realet.sip.GsonTypeAdapter.ChatAdapter
```

```
public class ChatAdapter
extends com.google.gson.TypeAdapter<Chat>
```

Ein Serialisierungs-Gson-TypeAdapter für die Chat Klasse.

Constructor Summary

Constructors

Constructor	Description
ChatAdapter()	

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
Chat	read (com.google.gson.stream.JsonReader in)	Nicht funktionelle Entserialisierungs-Methode.
void	write (com.google.gson.stream.JsonWriter out, Chat value)	Serialisierungs-Methode.

Methods inherited from class com.google.gson.TypeAdapter

[fromJson](#), [fromJson](#), [fromJsonTree](#), [nullSafe](#), [toJson](#), [toJson](#), [toJsonTree](#)

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#)

ChatAdapter

```
public ChatAdapter()
```

Method Detail**write**

```
public void write(com.google.gson.stream.JsonWriter out, Chat value) throws  
IOException
```

Serialisierungs-Methode. Schreibt folgendes in den "out" Parameter: `Chat.id`, `Chat.group` als Objekt, falls existent, `Chat.user1` als Objekt, falls existent, `Chat.user2` als Objekt, falls existent, `Chat.name`, falls existent.

Specified by:

`write` in class `com.google.gson.TypeAdapter<Chat>`

Parameters:

`out` - JsonWriter Objekt, welches das serialisierte `Chat` Objekt erhält.

`value` - Zu serialisierendes `Chat` Objekt.

Throws:

`IOException`

read

```
public Chat read(com.google.gson.stream.JsonReader in) throws IOException
```

Nicht funktionelle Entserialisierungs-Methode.

Specified by:

`read` in class `com.google.gson.TypeAdapter<Chat>`

Parameters:

`in` -

Returns:

`null`

Throws:

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

ALL CLASSES

SEARCH:

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Copyright © 2021. All rights reserved.

Package com.realet.sip.GsonTypeAdapter

Class ChatMessageAdapter

`java.lang.Object`

```
com.google.gson.TypeAdapter<ChatMessage>
com.realet.sip.GsonTypeAdapter.ChatMessageAdapter
```

```
public class ChatMessageAdapter
extends com.google.gson.TypeAdapter<ChatMessage>
```

Ein Serialisierungs-Gson-TypeAdapter für die ChatMessage Klasse.

Field Summary

Fields

Modifier and Type	Field	Description
-------------------	-------	-------------

(package verbosity private) int	o: Keine Attribute werden als Objekt serialisiert, stattdessen wird die ID des Objekts angegeben. 1: Serialisiere ChatMessage.author als Objekt, alle anderen Objekt-Attribute als IDs. 2: Serialisiere ChatMessage.author, ChatMessage.chat als Objekte.
--	---

Constructor Summary

Constructors

Constructor	Description
<code>ChatMessageAdapter(int verbosity)</code>	Constructor for ChatMessageAdapter

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
	<code>ChatMessage read(com.google.gson.stream.JsonReader in)</code>	Nicht funktionelle Entserialisierungs-Methode.
void	<code>write</code>	

```
fromJson, fromJson, fromJsonTree, nullSafe, toJson, toJson, toJsonTree
```

Methods inherited from class java.lang.Object

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait,  
wait, wait
```

Field Detail

verbosity

```
int verbosity
```

- 0: Keine Attribute werden als Objekt serialisiert, stattdessen wird die ID des Objekts angegeben.
- 1: Serialisiere ChatMessage.author als Objekt, alle anderen Objekt-Attribute als IDs.
- 2: Serialisiere ChatMessage.author, ChatMessage.chat als Objekte.

Constructor Detail

ChatMessageAdapter

```
public ChatMessageAdapter(int verbosity)
```

Constructor for ChatMessageAdapter

Parameters:

verbosity - Value for verbosity

Method Detail

write

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

`ChatMessage.chat`, abhangig von `verbosity`, `ChatMessage.author`, abhangig von `verbosity`,
`ChatMessage.edited`, falls existent.

Specified by:

`write` in class `com.google.gson.TypeAdapter<ChatMessage>`

Parameters:

`out` - `JsonWriter` object that will receive the serialized `ChatMessage` Objekt erhält.

`value` - Zu serialisierendes `ChatMessage` Objekt.

Throws:

`IOException`

read

`public ChatMessage read(com.google.gson.stream.JsonReader in) throws IOException`

Nicht funktionelle Entserialisierungs-Methode.

Specified by:

`read` in class `com.google.gson.TypeAdapter<ChatMessage>`

Parameters:

`in` -

Returns:

`null`

Throws:

`IOException`

Package com.realet.sip.GsonTypeAdapter

Class GroupAdapter

```
java.lang.Object
  com.google.gson.TypeAdapter<Group>
    com.realet.sip.GsonTypeAdapter.GroupAdapter
```

```
public class GroupAdapter
extends com.google.gson.TypeAdapter<Group>
```

Ein Serialisierungs-Gson-TypeAdapter für die `Group` Klasse.

Constructor Summary

Constructors

Constructor	Description
<code>GroupAdapter()</code>	

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
<code>Group</code>	<code>read</code> (<code>com.google.gson.stream.JsonReader</code> in)	Nicht funktionelle Entserialisierungs-Methode.
<code>void</code>	<code>write</code> (<code>com.google.gson.stream.JsonWriter</code> out, <code>Group</code> value)	Serialisierungs-Methode.

Methods inherited from class com.google.gson.TypeAdapter

`fromJson`, `fromJson`, `fromJsonTree`, `nullSafe`, `toJson`, `toJson`, `toJsonTree`

Methods inherited from class java.lang.Object

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`,
`wait`, `wait`

GroupAdapter

```
public GroupAdapter()
```

Method Detail**write**

```
public void write(com.google.gson.stream.JsonWriter out, Group value) throws  
IOException
```

Serialisierungs-Methode. Schreibt folgendes in den "out" Parameter: `Group.id`, `Group.description`, falls existent, `Group.name`, `Group.picture`, falls existent, `Group.owner`, als Objekt.

Specified by:

`write` in class `com.google.gson.TypeAdapter<Group>`

Parameters:

`out` - `JsonWriter` Objekt, welches das serialisierte `Group` Objekt erhält.

`value` - Zu serialisierendes `Group` Objekt.

Throws:

`IOException`

read

```
public Group read(com.google.gson.stream.JsonReader in) throws IOException
```

Nicht funktionelle Entserialisierungs-Methode.

Specified by:

`read` in class `com.google.gson.TypeAdapter<Group>`

Parameters:

`in` -

Returns:

`null`

Throws:

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

ALL CLASSES

SEARCH:

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Copyright © 2021. All rights reserved.

Package com.realet.sip.GsonTypeAdapter

Class PermissionAdapter

`java.lang.Object`

```
com.google.gson.TypeAdapter<Permission>
com.realet.sip.GsonTypeAdapter.PermissionAdapter
```

```
public class PermissionAdapter
extends com.google.gson.TypeAdapter<Permission>
```

Ein Serialisierungs-Gson-TypeAdapter für die `Permission` Klasse.

Constructor Summary

Constructors

Constructor	Description
<code>PermissionAdapter()</code>	

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
	<code>Permission read(com.google.gson.stream.JsonReader in)</code>	Nicht funktionelle Entserialisierungs-Methode.
void	<code>write (com.google.gson.stream.JsonWriter out, Permission value)</code>	Serialisierungs-Methode.

Methods inherited from class com.google.gson.TypeAdapter

`fromJson, fromJson, fromJsonTree, nullSafe, toJson, toJson, toJsonTree`

Methods inherited from class java.lang.Object

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait`

PermissionAdapter

```
public PermissionAdapter()
```

Method Detail**write**

```
public void write(com.google.gson.stream.JsonWriter out, Permission value) throws  
IOException
```

Serialisierungs-Methode. Schreibt folgendes in den "out" Parameter: `Permission.id`, `Permission.canRead`, `Permission.canWrite`, `Permission.role`, als Objekt, mit Attributen als IDs, falls existent.

Specified by:

`write` in class `com.google.gson.TypeAdapter<Permission>`

Parameters:

`out` - `JsonWriter` Objekt, welches das serialisierte `Permission` Objekt erhält.

`value` - Zu serialisierendes `Permission` Objekt.

Throws:

`IOException`

read

```
public Permission read(com.google.gson.stream.JsonReader in) throws IOException
```

Nicht funktionelle Entserialisierungs-Methode.

Specified by:

`read` in class `com.google.gson.TypeAdapter<Permission>`

Parameters:

`in` -

Returns:

`null`

Throws:

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

ALL CLASSES

SEARCH:

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Copyright © 2021. All rights reserved.

Package com.realet.sip.GsonTypeAdapter

Class RoleAdapter

```
java.lang.Object
  com.google.gson.TypeAdapter<Role>
    com.realet.sip.GsonTypeAdapter.RoleAdapter
```

```
public class RoleAdapter
extends com.google.gson.TypeAdapter<Role>
```

Ein Serialisierungs-Gson-TypeAdapter für die `Role` Klasse.

Field Summary

Fields

Modifier and Type	Field	Description
(package private) int	<code>verbosity</code>	0: Don't serialize <code>Role.users</code> . 1: Serialize <code>Role.users</code> as array of objects.

Constructor Summary

Constructors

Constructor	Description
<code>RoleAdapter(int verbosity)</code>	Constructor for <code>RoleAdapter</code>

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
Role	<code>read</code> (<code>com.google.gson.stream.JsonReader</code> in)	Nicht funktionelle Entserialisierungs-Methode.
void	<code>write</code> (<code>com.google.gson.stream.JsonWriter</code> out, <code>Role</code> value)	Serialisierungs-Methode.

Methods inherited from class java.lang.Object

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait`

Field Detail

verbosity

`int verbosity`

- 0: Don't serialize `Role.users`.
- 1: Serialize `Role.users` as array of objects.

Constructor Detail

RoleAdapter

`public RoleAdapter(int verbosity)`

Constructor for `RoleAdapter`

Parameters:

`verbosity` - Value for `verbosity`

Method Detail

write

`public void write(com.google.gson.stream.JsonWriter out, Role value) throws IOException`

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

write in class com.google.gson.TypeAdapter<Role>

Parameters:`out` - JsonWriter Objekt, welches das serialisierte `Role` Objekt erhält.`value` - Zu serialisierendes `Role` Objekt.**Throws:**`IOException`**read**`public Role read(com.google.gson.stream.JsonReader in) throws IOException`

Nicht funktionelle Entserialisierungs-Methode.

Specified by:

read in class com.google.gson.TypeAdapter<Role>

Parameters:`in` -**Returns:**`null`**Throws:**`IOException`

Package com.realet.sip.GsonTypeAdapter

Class UserAdapter

```
java.lang.Object
  com.google.gson.TypeAdapter<User>
    com.realet.sip.GsonTypeAdapter.UserAdapter
```

```
public class UserAdapter
extends com.google.gson.TypeAdapter<User>
```

Ein Serialisierungs-Gson-TypeAdapter für die `User` Klasse.

Constructor Summary

Constructors

Constructor	Description
<code>UserAdapter()</code>	

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
User	<code>read</code> (com.google.gson.stream.JsonReader in)	Nicht funktionelle Entserialisierungs-Methode.
void	<code>write</code> (com.google.gson.stream.JsonWriter out, <code>User</code> value)	Serialisierungs-Methode.

Methods inherited from class com.google.gson.TypeAdapter

`fromJson`, `fromJson`, `fromJsonTree`, `nullSafe`, `toJson`, `toJson`, `toJsonTree`

Methods inherited from class java.lang.Object

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`,
`wait`, `wait`

UserAdapter

```
public UserAdapter()
```

Method Detail**write**

```
public void write(com.google.gson.stream.JsonWriter out, User value) throws  
IOException
```

Serialisierungs-Methode. Schreibt folgendes in den "out" Parameter: `User.id`, `User.username`, `User.info`, falls existent, `User.profilePicture`, falls existent, `User.isOnline`, `User.status`, falls existent, `User.lastSeen`, falls existent.

Specified by:

`write` in class `com.google.gson.TypeAdapter<User>`

Parameters:

`out` - `JsonWriter` Objekt, welches das serialisierte `User` Objekt erhält.

`value` - Zu serialisierendes `User` Objekt.

Throws:

`IOException`

read

```
public User read(com.google.gson.stream.JsonReader in) throws IOException
```

Nicht funktionelle Entserialisierungs-Methode.

Specified by:

`read` in class `com.google.gson.TypeAdapter<User>`

Parameters:

`in` -

Returns:

`null`

Throws:

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

ALL CLASSES

SEARCH:

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Copyright © 2021. All rights reserved.

Package com.realet.sip.GsonTypeAdapter

Class UserProfileAdapter

```
java.lang.Object
  com.google.gson.TypeAdapter<UserProfile>
    com.realet.sip.GsonTypeAdapter.UserProfileAdapter
```

```
public class UserProfileAdapter
extends com.google.gson.TypeAdapter<UserProfile>
```

Ein Serialisierungs-Gson-TypeAdapter für die UserProfile Klasse.

Constructor Summary

Constructors

Constructor	Description
UserProfileAdapter()	

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
	<code>UserProfile read(com.google.gson.stream.JsonReader in)</code>	Nicht funktionelle Entserialisierungs-Methode.
void	<code>write (com.google.gson.stream.JsonWriter out, UserProfile value)</code>	Serialisierungs-Methode.

Methods inherited from class com.google.gson.TypeAdapter

`fromJson, fromJson, fromJsonTree, nullSafe, toJson, toJson, toJsonTree`

Methods inherited from class java.lang.Object

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait`

UserProfileAdapter

```
public UserProfileAdapter()
```

Method Detail**write**

```
public void write(com.google.gson.stream.JsonWriter out, UserProfile value) throws  
IOException
```

Serialisierungs-Methode. Schreibt folgendes in den "out" Parameter: `UserProfile.id`, `UserProfile.reverseBlocking`, `UserProfile.maxLoadedMessages`, `UserProfile.messageChunkSize`.

Specified by:

`write` in class `com.google.gson.TypeAdapter<UserProfile>`

Parameters:

`out` - `JsonWriter` Objekt, welches das serialisierte `UserProfile` Objekt erhält.

`value` - Zu serialisierendes `UserProfile` Objekt.

Throws:

`IOException`

read

```
public UserProfile read(com.google.gson.stream.JsonReader in) throws IOException
```

Nicht funktionelle Entserialisierungs-Methode.

Specified by:

`read` in class `com.google.gson.TypeAdapter<UserProfile>`

Parameters:

`in` -

Returns:

`null`

Throws:

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

ALL CLASSES

SEARCH:

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Copyright © 2021. All rights reserved.