

SIP

Dies ist die Dokumentation der Sip-Webapp. Alle Komponenten sind in der Seitenleiste zu finden, Dokumentation zu allen Skripten hier.

- SIP
 - [src/main.js](#)
 - [src/store/index.js](#)
 - [src/router/index.js](#)

src/main.js

Das Skript main.js etabliert einige Konstanten, die durch den Vue prototype für alle Komponenten erreichbar sind.

```
Vue.prototype.$notificationTimeout = 5000;  
Vue.prototype.$apiHost = ""  
Vue.prototype.$uploadHost = ""
```

js

Ein Event-Hub wird erstellt:

```
//global event bus  
Vue.prototype.$eventHub = new Vue();
```

js

Außerdem definiert es einige Methoden, ebenfalls über den Vue.prototype:

Method	Description	Parameters
--------	-------------	------------

Method	Description	Parameters
\$getApiUrl	Findet eine URL der API mit Protokoll.	protocol
\$getUploadUrl	Findet eine URL zum Upload-Directory mit Protokoll.	protocol
\$getAvatarUrl	Findet die URL eines Avatars anhand eines Object und seines Typs: "user" oder "group".	type, obj
\$initApp	Holt funktionsrelevante Daten (user, userProfile, blockedUsers, blockedBy) vom Server und markiert die App im store als initialisiert, sobald alle Daten angekommen sind.	-

Eine Hilfsmethode wird lokal definiert:

Method	Description	Parameters
createUserWatcher	Aktiviert eine global WebSockets Verbindung nach <code>/users/{userId}/websockets</code> und registriert callbacks für Fehler und Nachrichten. Wenn eine neue chatMesssage ankommt, wird sie durch den globalen Event-Hub geschickt.	protocol

src/store/index.js

Definiert den Store und Setter-Mutations für alle Attribute.

Store Element	Bedeutung
token	Das Token der aktuelle Session
userId	Die eigene userId
user	Der eigene User
ws	Das WebSocket Objekt, welches den User beobachtet
initialized	Ob alle funktionsnotwendigen Informationen geladen sind

Store Element	Bedeutung
blockedUsers	List aller vom eigenen Nutzer blockierten Nutzer
blockedBy	List aller Nutzer, die den eigenen Nutzer blockiert haben
userProfile	UserProfile Objekt des eigenen Nutzers

src/router/index.js

Definiert alle Routen:

```
const routes = [  
  {  
    path: '/login',  
    name: 'Login',  
    component: Login  
  },  
  {  
    path: '/register',  
    name: 'Register',  
    component: Register  
  },  
  {  
    path: '/home',  
    component: Home,  
    children: [  
      {  
        path: '',  
        redirect: 'direct-chats',  
        name: 'Home',  
      },  
      {  
        path: 'direct-chats',
```

js

```
        component: HomeDirectChats,
        name: "HomeDirectChats",
    },
    {
        path: 'groups',
        component: HomeGroups,
        name: "HomeGroups",
    }
]
},
{
    path: '/chat/:chatId',
    component: DirectChat,
    name: 'DirectChat'
},
{
    path: '/group/:groupId/chat/:chatId',
    component: Group,
    name: 'GroupChat'
},
{
    path: '/group/:groupId',
    component: Group,
    name: 'Group'
},
{
    path: '/group/:groupId/edit',
    component: GroupEdit,
    children: [
        {
            path: 'overview',
            component: GroupEditOverview,
            name: 'GroupEditOverview',
        },
        {
            path: 'roles',
            component: GroupEditRolesView,
            name: "GroupEditRolesView",
        }
    ]
}
```

```
    },
    {
      path: 'permissions',
      component: GroupEditPermissionsView,
      name: "GroupEditPermissionsView",
    },
    {
      path: 'users',
      component: GroupEditUsersView,
      name: "GroupEditUsersView",
    }
  ]
},
{
  path: '/edit',
  component: Edit,
  children: [
    {
      path: 'profile',
      component: EditProfile,
      name: 'EditProfile'
    },
    {
      path: 'settings',
      component: EditSettings,
      name: 'EditSettings'
    },
  ]
},
{
  path: '',
  name: 'FrontPage',
  component: FrontPage,
}
]
```

Definiert außerdem einen globalen `forEach Navigation-Guard`. Dieser überprüft, falls die Navigation zu

`/home...`

`/chat...`

`/group...`

`/edit...`

führt, ob ein Token im Store vorhanden ist. Falls nein, wird im Cookie gesucht. Wird dort eins gefunden, wird es beim Server überprüft. Ist dies erfolgreich, wird die App initialisiert.

Falls die Anwendung initialisiert ist, die Navigation aber zu einer anderen Route führt, wird die Anwendung terminiert.

App

Leere Rahmen-Komponente. Enthält

`<router-view>`

, welches die Anwendung beinhaltet.

AddUserConfirmDialog

Bittet den Nutzer um Bestätigung, dass er einen Nutzer zu einer Gruppe/Rolle hinzufügen will.

Events

Event Name	Description	Parameters
added	Wenn der Nutzer hinzugefügt wurde.	User, falls er zu einer Rolle und nicht zu einer Gruppe hinzugefügt wurde.

Methods

Method	Description	Parameters
show	Zeigt den Dialog. Wird dies als show(user) aufgerufen, fragt die Komponente, ob der Nutzer zu einer Gruppe hinzugefügt werden soll. Wird dies als show(user, role) aufgerufen, fragt die Komponente, ob der Nutzer zu einer Rolle hinzugefügt werden soll.	user role
close	Schließt den Dialog.	-
add	Fügt den Nutzer, abhängig davon, wie "show" aufgerufen wurde, beim Server entweder zu einer Gruppe oder einer Rolle hinzu.	-

ChatCard

Zeigt für Direkt-Unterhaltungen auf einer Karte sowohl den anderen Nutzer als auch die zuletzt gesendet Nachricht an.

Props

Name	Description	Type	Required	Default
chat	Der angezeigte Chat	Object	false	-

ChatEditorDialog

Erlaubt das Bearbeiten von Gruppenchats.

Events

Event Name	Description	Parameters
chat-updated	Emittiert den aktualisierten Chat	chat
chat-created	Emittiert den erstellten Chat	chat

Methods

Method	Description	Parameters
show	Öffnet den Dialog. Falls ein "chat"-Paramter angegeben ist, wird der Dialog im Bearbeitungsmodus geöffnet, andernfalls wird er der Erstellungsmodus gewählt.	chat
submit	Aktualisiert/erstellt, abhängig vom Modus, beim Server einen Chat.	-
close	Schließt den Dialog	-

ChatWindow

Sobald erstellt, lädt diese Komponente die Berichtigungen zu einem Chat sowie die neuesten Nachrichten, die Anzahl ist definiert durch `userProfile.messageChunkSize` im store. Falls der Nutzer nicht lesen oder schreiben darf, werden jeweils schriftzüge angezeigt. Gegebenenfalls verschwindet das Eingabefeld. Nachrichten werden gruppiert, falls sie vom selben Nutzer stammen, am selben Tag geschrieben

Events

Event Name	Description	Parameters
show-user	Wenn das Profilbild oder der Name eines Nutzers angeklickt wird.	user

Methods

Method	Description	Parameters
updateMessages	Lädt neue Nachrichten und setzt <code>hasOldest</code> und <code>hasNewest</code> entsprechend. Ruft "addMessages" mit der Serverantwort auf.	-

Method	Description	Parameters
sendMessage	Sendet eine Nachricht an den Server, falls "editing" false ist und sie nicht ausschliesslich ausschließlich aus Leerzeichen besteht. Aktualisiert eine Nachricht beim Server, falls "editing" true ist und sie nicht ausschliesslich ausschließlich aus Leerzeichen besteht.	-
scrollDownMessages	Scrollt den Nachrichtencontainer nach unten.	-
addMessages	Fügt Nachrichten lokal hinzu. Nimmt Rücksicht auf userProfile.maxLoadedMessages im store und behält Scroll-Position bei. Achtung: nimmt an, dass sich der gesamte Array entweder vor oder hinter den geladenen Nachrichten befindet.	-
copyToClipboard	Kopiert einen Text zur Zwischenablage.	text
deleteMessage	Löscht eine Nachricht beim Server.	chatMessage
removeMessage	Entfernt eine Nachricht lokal.	chatMessage
editMessage	Leert das Eingabefeld und aktiviert Bearbeitungsmodus.	chatMessage
updateMessage	Aktualisiert eine Nachricht lokal.	chatMessage
createWatcher	Baut WebSockets Verbindung auf, um den Chat zu beobachten. Registriert callback für neue Nachrichten.	-
onNewMessage	Callback für den globalen Event-Hub bei Event "new-message".	chatMessage
getPermissions	Lädt Permissions.	-

GroupCard

Zeigt den Namen, das Bild und die Beschreibung einer Gruppe auf einer Karte.

Props

Name	Description	Type	Required	Default
group	Die angezeigte Gruppe.	Object	false	-

GroupCreatorDialog

Erlaubt das Erstellen von Gruppen samt Namen, Beschreibung und Bild.

Events

Event Name	Description	Parameters
open-group-id	Emittiert die der erstellten Gruppe.	group

Methods

Method	Description	Parameters
updateFile	Ändert das imgFile und sendet ein post-request an "/api/images/groups/pictures" um einen neuen Picture-Code zu erhalten.	file

Computed

Computed	Type	Description	From Store
avatarUrl	-	Generiert eine Object-Url anhand des imgFile.	No

LoadingScreen

Zeigt das Logo mittig auf einem dunkelgrauen Hintergrund.

LoginForm

Hier kann sich der Nutzer anmelden.

Methods

Method	Description	Parameters
onSubmit	Führt Anmeldung durch. Falls dies fehlschlägt wird eine Fehlermeldung gezeigt.	-

MessageAlerts

Eine Snackbar, die die neueste erhaltene Nachricht anzeigt. Falls eine neue Nachricht ankommt, wird der Timer zurückgesetzt. Keine neue Snackbar wird angezeigt. Registriert einen Event-Listener für den globalen Event-Hub.

Events

Event Name	Description	Parameters
open-chat	Wenn der entsprechende Knopf geklickt wird.	chat

Methods

Method	Description	Parameters
onNewMessage	Zeigt die Snackbar bzw. setzt das Timeout zurück und zeigt die neue Nachricht an.	chatMessage
openChat	Schließt den Dialog und emittiert ein "open-chat" event.	-

MessageOptionsMenu

Ein Menü mit Aktionen für Nachrichten.

Props

Name	Description	Type	Required	Default
chatMessage	Die Nachricht, für die das Menü angezeigt wird.	Object	false	-

Events

Event Name	Description	Parameters
copy-to-clipboard	Wenn der Nutzer den "Copy" Menüeintrag anklickt.	-
edit-message	Wenn der Nutzer den "Edit" Menüeintrag anklickt.	-
delete-message	Wenn der Nutzer den "Delete" Menüeintrag anklickt.	-

RegisterForm

Erlaubt Nutzerregistrierung in 3 Schritten:

- 1. Name + Login-Informationen
- 2. Profil
- 3. Einstellungen

Methods

Method	Description	Parameters
updateFile	Ändert das imgFile und sendet ein post-request an "/api/images/users/pictures" um einen neuen Picture-Code zu erhalten.	file
onSubmit	Erstellt den Nutzer beim Server. Falls dies gelingt, wird die Anwendung nach "/login" geroutet.	-
testEmail	Testet ob eine E-Mail gültig ist und gibt ggf. eine Fehlermeldung zurück.	v

Computed

Computed	Type	Description	From Store
avatarUrl	-	Generiert eine Object-Url anhand des imgFile.	No

RemoveUserConfirmDialog

Bittet den Nutzer um Bestätigung, dass er einen Nutzer aus einer Gruppe/Rolle entfernen will.

Events

Event Name	Description	Parameters
removed	Wenn der Nutzer erfolgreich entfernt wurde	user

Methods

Method	Description	Parameters
show	Zeigt den Dialog. Wird dies als show(user) aufgerufen, fragt die Komponente, ob der Nutzer aus einer Gruppe entfernt werden soll. Wird dies als show(user, role) aufgerufen, fragt die Komponente, ob der Nutzer aus einer Rolle entfernt werden soll.	user role
close	Schließt den Dialog.	-
remove	Entfernt den Nutzer, abhängig davon, wie "show" aufgerufen wurde, beim Server entweder aus einer Gruppe oder aus einer Rolle.	-

UserFinderDialog

Lässt den Nutzer andere Nutzer anhand ihrer ID suchen.

Events

Event Name	Description	Parameters
selected-user	-	-

Methods

Method	Description	Parameters
show	Setzt den Dialog zurück und zeigt ihn.	-
search	Sucht den Nutzer beim Server. Zeigt eine Fehlermeldung, falls nicht gefunden.	-
close	Schließt den Dialog.	-

UserProfileDialog

Zeigt das Profil eines Nutzers in einem Dialog an. Erlaubt, den Nutzer zu blockieren und einen Chat mit ihm zu öffnen.

Events

Event Name	Description	Parameters
open-direct-chat	Wenn der Nutzer auf "MESSAGE" klickt um den Direct-Chat zu öffnen.	chat
open-group	Wenn der Nutzer eine Gruppe anklickt.	group

Methods

Method	Description	Parameters
openDirectChat	Lädt einen Direct-Chat zwischen dem gezeigten Nutzer und dem eingeloggten Nutzer vom Server und emittiert 'open-direct-chat' mit ihm.	-
openGroup	Emittiert 'open-group' mit einem Parameter group und schließt den Dialog.	group
show	Setzt die Komponente zurück und zeigt den Dialog.	-
blockUser	Blockiert einen Nutzer beim Server.	-

Method	Description	Parameters
unblockUser	Hebt eine Blockierung beim Server auf.	-

Computed

Computed	Type	Description	From Store
userIsBlocked	-	Ob der gezeigt Nutzer blockiert ist.	No

DirectChat

Diese Komponente zeigt eine Direkt-Unterhaltung an und erlaubt Interaktion. Auf der linken Seite der Anwendung befindet sich ein Navigation Drawer, der es erlaubt, zwischen Chats zu wechseln. Die App-Leiste zeigt Informationen wie den Online-Status, sowie, falls nicht online, das Datum, an dem der Nutzer zuletzt online war, an. Registriert einen Event-Listener für den globalen Event-Hub, der auf neue Nachrichten reagiert. Der Navigation drawer verschwindet und lässt sich optional öffnen, sollte der Bildschirm weniger als 600px breit sein.

Methods

Method	Description	Parameters
showUser	Zeigt den UserProfileDialog mit dem entsprechenden User.	user
openChat	Öffnet eine beliebige Art von Chat, überprüft ob Chat ein Gruppen-/Direkt-Chat ist und routet die Anwendung entweder zu entsprechenden URL oder ruft "openDirectChat" auf.	chat
openDirectChat	Passt die Komponente an um einen anderen Chat anzuzeigen.	chat
openGroup	Routet die Anwendung zur "Group"-Komponente mit der entsprechenden Id.	group
findNotSelf	Findet den User eines Chats, der nicht mit der im Store gespeicherten userId übereinstimmt und fügt ihn dem Chat als "notSelf" hinzu.	chat
onNewMessage	Verschiebt den Chat einer neuen Nachricht in der Liste nach oben.	message

Method	Description	Parameters
showUserFinder	Zeigt den UserFinderDialog.	-
fetchChats	Sendet eine Server-Abfrage um alle Chats zu erhalten und ändert die Komponente um diese anzuzeigen.	-

Edit

Zeigt einen Navigation Drawer auf der linken Bildschirmseite, ein

```
<router-view>
```

mit den Sub-Routes "EditProfile", "EditSettings" auf der rechten Seite. Der Navigation drawer verschwindet und lässt sich optional öffnen, sollte der Bildschirm weniger als 600px breit sein.

Methods

Method	Description	Parameters
getTitle	Generiert den Titel, basierend auf einem Schlüsselwort aus der Route.	keyword

EditProfile

Erlaubt, das Profil des eigenen Nutzers zu bearbeiten, das Passwort zu ändern oder den Account zu löschen.

Methods

Method	Description	Parameters
updateFile	Ändert das imgFile und sendet ein post-request an "/api/images/users/pictures" um einen neuen Picture-Code zu erhalten.	file
updateUser	Aktualisiert den User beim Server.	-
changePass	Aktualisiert das Passwort beim Server.	-
openDeleteDialog	Öffnet einen Dialog, der das Löschen des Nutzers bestätigen lässt.	-
deleteUser	Löscht den User beim Server	-

Computed

Computed	Type	Description	From Store
avatarUrl	-	Generiert eine Object-Url anhand des imgFile.	No

EditSettings

Erlaubt, die User-Einstellungen zu bearbeiten.

Methods

Method	Description	Parameters
onSubmit	Aktualisiert die Einstellungen in Form eines "UserProfile" beim Server.	-

FrontPage

Zeigt das sip-Logo und Partikel im Hintergrund mit
<https://github.com/marcbruederlin/particles.js> 

Group

Diese Komponente zeigt eine Gruppe an und erlaubt Interaktion. Auf der linken Seite der Anwendung befindet sich ein Navigation Drawer, der es erlaubt, zwischen Chats und Gruppen zu wechseln. Falls der Nutzer administrative Rechte hat, kann er hier auch Chats hinzufügen, bearbeiten und entfernen. Auf der rechten Seite der Anwendung befindet sich ein Navigation Drawer, der alle Gruppenmitglieder, nach Rollen sortiert, anzeigt. Die App-Leiste zeigt den Namen des aktuellen Chats, sowie ein Icon für den zweiten Navigation Drawer und, falls der User administrative Rechte hat, ein Icon als Link zu den Gruppeneinstellungen an. Die Navigation Drawers verschwindet und lassen sich optional öffnen, sollte der Bildschirm weniger als 600px breit sein.

Methods

Method	Description	Parameters
toggleUserDrawer	Zeigt den rechten Navigation Drawer.	-
showUser	Zeigt den UserProfileDialog für einen User.	user
openDirectChat	Routet die Anwendung zum entsprechenden Chat.	chat
openGroup	Passt die Komponente an um eine gewisse Gruppe zu zeigen.	group
openChat	Öffnet eine beliebige Art von Chat, überprüft ob Chat ein Gruppen-/Direkt-Chat ist und routet die Anwendung entweder zu entsprechenden URL oder öffnet einen Gruppen-Chat auf.	chat
initGroup	Lädt alle Chats, Rollen und Nutzer ohne Rolle einer Gruppe.	-

Method	Description	Parameters
resetView	Setzt die Komponente zurück.	-
getGroups	Lädt alle Gruppen eines Nutzers.	-
openGroupld	Öffnet eine Gruppe anhand ihrer ID.	id
showGroupCreator	Zeigt den "GroupCreatorDialog".	-
createChat	Zeigt den ChatEditorDialog im Erstellungs-Modus.	-
editChat	Zeigt den ChatEditorDialog im Bearbeitungs-Modus mit dem "chat" Parameter.	chat
deleteChat	Löscht einen Chat beim Server.	chat

Computed

Computed	Type	Description	From Store
isAdmin	-	Bestimmt anhand der geladenen Rollen und der geladenen Gruppe ob der User administrative Rechte hat.	No
rolesWithMembers	-	Findet alle Rollen die Mitglieder haben.	No

GroupEdit

Zeigt einen Navigation Drawer auf der linken Bildschirmseite, ein

```
<router-view>
```

mit den Sub-Routes "GroupEditOverview", "GroupEditPermissionsView", "GroupEditRolesView", "GroupEditUsersView" auf der rechten Seite. Der Navigation drawer verschwindet und lässt sich optional öffnen, sollte der Bildschirm weniger als 600px breit sein.

Methods

Method	Description	Parameters
getTitle	Generiert den Titel, basierend auf einem Schluesselwort aus der Route.	keyword

GroupEditOverview

Erlaubt das Bearbeiten grundlegender Eigenschaften einer Gruppe wie z.B. ihren Namen und das Löschen einer Gruppe.

Methods

Method	Description	Parameters
updateFile	Ändert das imgFile und sendet ein post-request an "/api/images/groups/pictures" um einen neuen Picture-Code zu erhalten.	file
submit	Aktualisiert die Gruppe beim Server.	-
openDeleteDialog	Öffnet einen Dialog, der das Löschen der Gruppe bestätigen lässt.	-
deleteGroup	Löscht die Gruppe beim Server.	-

Computed

Computed	Type	Description	From Store
avatarUrl	-	Generiert eine Object-Url anhand des imgFile.	No

GroupEditPermissionsView

Erlaubt das Hinzufügen, Bearbeiten und Entfernen von Berechtigungen.

Methods

Method	Description	Parameters
fetchRoles	Lädt alle Rollen einer Gruppe.	-
fetchChats	Lädt alle Chats einer Gruppe.	-
fetchPermissions	Lädt alle Permissions eines Chats und findet die Permission ohne Role, die "rule". Alle anderen bilden die "exceptions".	-
showRoleFinder	Zeigt einen Dialog mit einer Gruppe.	-
addException	Fügt einen neue Permission beim Server hinzu.	role
updatePermission	Aktualisiert eine Permission beim Server.	permission
deleteException	Löscht eine Permission beim Server.	permission

Computed

Computed	Type	Description	From Store
----------	------	-------------	------------

Computed	Type	Description	From Store
rolesWithoutExceptions	-	Findet alle Rollen, die keine Berechtigung für den aktuellen Chat haben.	No

Watch

Name	Description	Parameters
chatId	Wenn ein neuer chat ausgewählt wird, werden die Berechtigungen neu geladen.	-

GroupEditRolesView

Erlaubt, Rollen zu kreieren, zu bearbeiten und zu löschen.

Methods

Method	Description	Parameters
fetchRoles	Lädt alle Rollen der Gruppe.	-
submit	Aktualisiert eine Rolle beim Server, falls sie eine ID hat, falls nicht, muss die Rolle neu sein und wird neu hinzugefügt.	-
createRole	Erstellt lokal eine neue Rolle ohne ID.	-
deleteRole	Löscht eine Rolle beim Server.	-
removeUser	Zeigt den "RemoveUserConfirmDialog" mit einem "user" Parameter.	user
addUser	Zeigt den "AddUserConfirmDialog" mit einem "user" Parameter.	user
showUserFinder	Zeigt einen Dialog mit allen Gruppenmitgliedern, die nicht Teil der Rolle sind. Dieser erlaubt es, Nutzer zur Rolle hinzuzufügen.	-
localRemoveUserFromRole	Entfernt einen Nutzer aus dem "user"-Parameter lokal aus einer Rolle.	user
localAddUserToRole	Fügt einen Nutzer aus dem "user"-Parameter lokal zu einer Rolle hinzu.	user

Method	Description	Parameters
fetchUsers	Lädt alle Mitglieder einer Gruppe.	-
swapRolesUpwards	Verschiebt eine Rolle mit index von Parameter "idx" nach oben. Überprüft nicht, ob dies möglich ist.	idx
swapRolesDownwards	Verschiebt eine Rolle mit index von Parameter "idx" nach unten. Überprüft nicht, ob dies möglich ist.	idx
showPriorityEditor	Zeigt einen Dialog mit einer Liste aller Rollen. Dieser erlaubt, die Reihenfolge dieser Rollen zu ändern.	-
updateRolePriorities	Aktualisiert die Prioritäten der Rollen beim Server.	-

Computed

Computed	Type	Description	From Store
notMembers	-	Findet alle Nutzer, die nicht Teil der aktuellen Rolle sind.	No

Watch

Name	Description	Parameters
roleId	Wenn eine neue Rolle ausgewählt wird, wird diese auch zur Bearbeitung bereitgestellt.	-

GroupEditUsersView

Erlaubt, Nutzer zu Gruppen hinzuzufügen und sie zu entfernen.

Methods

Method	Description	Parameters
removeUser	Zeigt den RemoveUserConfirmDialog.	user
addUser	Zeigt den AddUserConfirmDialog.	user
fetchUsers	Lädt alle Nutzer.	-
fetchGroup	Lädt die aktuelle Gruppe.	-
showUserFinder	Zeigt den UserFinderDialog.	-

Home

Der Home-Screen zeigt eine App Bar und eine Liste von Chats oder Gruppen in einem

```
<router-view>
```

mit "HomeDirectChats", "HomeGroups".

Methods

Method	Description	Parameters
openChat	Routet die Anwendung zu einem Chat.	chat
fetchUser	Lädt den eigenen User.	-
logout	Loggt den User beim Server aus.	-

HomeDirectChats

Zeigt alle Direct-Chats des eigenen Users in Form von Karten

Methods

Method	Description	Parameters
showUserDialog	Zeigt den UserProfileDialog.	user
openDirectChat	Routet die Anwendung zu einem Chat.	chat
openGroup	Routet die Anwendung zu einer Group.	group
showUserFinder	Zeigt den UserFinderDialog.	-

HomeGroups

Zeigt alle Gruppen des eigenen Users in Form von Karten

Methods

Method	Description	Parameters
showGroupCreator	Zeigt den GroupCreatorDialog.	-
openGroupId	Zeigt eine Gruppe anhand ihrer ID.	id

LoginPage

Zeigt das LoginForm. Initialisiert die App, falls ein gültiges Token im Cookie gefunden werden kann.

RegisterPage

Zeigt das RegisterForm. Initialisiert die App, falls ein gültiges Token im Cookie gefunden werden kann.