Brug af Api til sletning af brugere i Prepare

Forudsætning

Oprettelse af jeres koncern i Identity server på forespørgsel via support@firstagenda.com

Token

Token til API'et requestes derefter via:

Access Token URL:

https://identity.firstagenda.com/connect/token

Auth URL:

https://identity.firstagenda.com/connect/authorize

Client ID:

Provided_ID

Client Secret:

Provided_Secret

Authentication

API'et er sikret ved hjælp af Bearer tokens, der er udstedt af vores identitetsserver, som kan tilgås her:

https://identity.firstagenda.com

API'et bruger OpenID Connect til godkendelsesflowet (som er et supersæt af OAuth2), og vi udsteder klienter, der tillader, at client credential-flowet bruges til at hente et adgangs-token.

Det, Client-ID vi giver, vil være knyttet til en bestemt kunde og deres data, således at når du kalder API'et, returneres kun tilgængelige data for den pågældende kunde. Disse oplysninger er integreret i token.

Når du konstruerer POST-request efter OAuth2 -client credential-flowet, skal du bruge følgende detaljer:

Grant type: Client Credentials

Access Token URL: https://identity.qa.firstagenda.com/connect/token

Client ID: Kunden angiver det Client ID, som FirstAgenda har leveret

Client Secret: Kunden angiver det Client Secret, som FirstAgenda har leveret

Scope: Skal være FirstAgenda Integrations API – dette giver adgang til Integration API

Example Post call:


```
POST /connect/token
Host: https://identity.firstagenda.com

grant_type=client_credentials
&client_id=xxxxxx
&client_secret=xxxxxx
&scope=prepare_integrationapi
```

Som vil returnere et json-object med formularen:

```
{
   "access_token" :  "XXXXXX" ,
   "expires_in" :  3600 ,
   "token_type" :  "Bearer"
}
```

Værdien af access_token skal derefter inkluderes i alle kald til Integration API indenfor Authorization header med en værdi af Bearer ${access_token}, f.eks. Bearer XXXXXX:

```
GET /api/publication/agenda/123
Host: https://integrationapi.firstagenda.com
Accept: application/json
```

Hvor ofte skal token opdateres

Ud fra ovenstående token-eksempel, har token en bestemt levetid beskrevet som expires_in (selve token er et JWT -token, som kan afkodes. I det findes et tidsstempel i epokformat for, hvornår tokenet udløber).

Tokenet bør derfor bruges til alle anmodninger, der er lavet, indtil tokenet er ved at udløbe, når et nyt token skal hentes.

Det er ikke nødvendigt at hente et nyt token for hvert kald, da token i sig selv ikke har en bestemt kontekst (bortset fra at være i kundens kontekst).
Vi anbefaler at opdatere token lidt, før det udløber, f.eks. 5 minutter før, ellers kan der ske en HTTP 401, hvis det tilfældigt udløber, mens det bruges til langvarige operationer (måske trækkes der flere sider med resultater og den udløber mellem siderne)

EndPoint – Fjern brugere

https://prepare.firstagenda.com/api/v1/integration/account/synchronization/DeleteUser?organisationUid=[Guid]&email=[string]