



REST-API-Spezifikation

Schnittstellen-Dokumentation

Datum: 15.01.2021

Inhalt

1. Änderungshistorie.....	4
2. Anmeldefunktionen.....	5
2.1 Sitzungsschlüsselanfordern.....	5
2.2 Anmelden.....	5
2.3 Abmelden.....	5
2.4 Benutzerinformationen.....	6
3. Flugdatenerfassung.....	7
3.1 Flug anlegen.....	7
3.2 Flug bearbeiten.....	8
3.3 Flug löschen.....	9
3.4 Flug auslesen.....	9
3.5 Alle Flüge von heute auslesen.....	10
3.6 Alle Flüge für einen Tag auslesen.....	10
3.7 Die letzten Flüge für ein LFZ auslesen.....	11
3.8 Meine letzten Flüge auslesen.....	11
3.9 Die letzten Flüge für eine Person auslesen.....	11
3.10 Zuletzt geänderte Flüge auslesen.....	12
4. Kalender.....	13
4.1 Öffentlichen Kalender auslesen.....	13
5. Mitglieder.....	14
5.1 Auslesen der Mitgliederliste.....	14
6. Reservierungen.....	16
6.1 Aktuelle Reservierungen auslesen.....	16
7. Instandhaltung.....	17
7.1 Aktuelle Zellenzeiten eines LFZs auslesen.....	17
8. Finanzen - Buchungen.....	18
8.1 Buchung anlegen.....	18
8.2 Einzelne Buchung auslesen.....	18
8.3 Alle Buchungen mit heutigem Buchungsdatum auslesen.....	19
8.4 Alle Buchungen zu einem Jahr auslesen.....	19
8.5 Alle Buchungen für einem Zeitraum auslesen.....	19
9. Arbeitsstunden.....	20
9.1 Arbeitsstunden auslesen.....	20
9.2 Arbeitsstunden anlegen.....	21
9.3 Arbeitsstundenkategorien auslesen.....	21

10. Allgemeiner Verkauf.....	22
10.1 Artikelliste auslesen.....	22
10.2 Verkauf anlegen.....	22
11. Kontakt.....	23

1. Änderungshistorie

Datum	Änderung
01.06.2015	Erstfassung
31.03.2016	Neues Felder bei der Flugdatenerfassung (Abrechnungsmodus, Abgerechnet über)
28.06.2016	Anlage von F-Schlepps möglich (Neue Felder towcallsign, towpilotname, towheight und towtime)
29.08.2016	Angabe der uid anstelle des Namens bei Flügen möglich. Neue Felder (uidpilot, uidattendant, uidattendant2, uidattendant3 und towuidpilot)
25.10.2016	Offblock und onblock bei Flügen ergänzt
10.03.2017	pilotmemberid und pilotattendant bei Flügen ergänzt
13.06.2017	Motorzeiten bei Flügen ergänzt
03.11.2017	Auslesen der Mitgliederliste
28.04.2018	Neue Felder in der Mitgliederliste (Rundmailempfänger, Funktionen, Sparten)
04.05.2018	Neuer Endpunkt zum Auslesen von Reservierungen
22.10.2018	Neuer Endpunkt zum Auslesen der letzten Flüge zu einem Callsign Neuer Endpunkt zum Auslesen der letzten Flüge zu einer Person (uid)
07.11.2018	Ausgabe der weitere Eigenschaften bei Mitgliedern
18.12.2018	Neues Feld beim Auslesen von Flügen: planetype
22.03.2019	Neuer Endpunkt zum Auslesen der aktuellen Zellenzeiten eines LFZs
19.05.2019	AppKey bei Anmeldungen
14.08.2019	Anlegen und auslesen von Buchungen
13.09.2019	Neuer Parameter bei der Anmeldung: Angabe des aktuellen Secrets bei Verwendung der Zwei-Faktor-Authentifizierung
12.10.2019	F-Schlepps können nun auch bei der Flugaktualisierung angelegt werden
05.12.2019	Alle Buchungen für einen Zeitraum auslesen.
13.01.2020	Nicht beendete Lehrpläne am Mitglied (Mitgliederliste)
19.06.2020	Neuer Parameter need_2fa in der Fehlermeldung nach einer Anmeldung
16.09.2020	Sachkonten bei Mitgliederdaten
10.11.2020	Neuer Endpunkt zum Auslesen von zuletzt geänderten Flügen
02.01.2021	Auslesen und Anlegen von Arbeitsstunden
15.01.2021	Anlegen von Verkäufen

2. Anmeldefunktionen

2.1 Sitzungsschlüsselanfordern

URL	interface/rest/auth/accesstoken
Methode	GET
Parameter	Keine
Antwort	[accesstoken]
HttpStatus	200 = OK
Kommentar	Liefert das Accesstoken für alle weiteren Aufrufe

2.2 Anmelden

URL	interface/rest/auth/signin
Methode	POST
Parameter	<code>[accesstoken]:[string]</code> <code>[username]:[string]</code> - Benutzername <code>[password]:[string]</code> - Passwort <code>[appkey]:[string]</code> - Eindeutiger Applikationsschlüssel <code>[cid]:[int]</code> - Eindeutige Nummer des Vereins (Optional) <code>[auth_secret]:[string]</code> - Zwei-Faktor-Authentifizierung (Optional) Hinweis: Das Passwort muss vorher verschlüsselt werden: <code>md5(password)</code> Hinweis: Die eindeutige cid wird nur benötigt, wenn der Benutzer in mehreren Vereinen existiert. Der für die Anmeldung erforderliche appkey kann über den Support angefragt werden. Die max. Anzahl an Requests (gemeint sind alle Aufrufe) ist auf 1000 je Tag limitiert. Die kommerzielle Nutzung der Schnittstelle ist grundsätzlich untersagt.
Antwort	Keine
HttpStatus	200 = Anmeldung erfolgreich 403 = Fehlerhafte Anmeldung <code>[error]:[string]</code> Fehlerbedschreibung <code>[need_2fa]:[int]</code> 1= Wenn der User, der sich anmelden wollte, eine Zwei-Faktor-Auth aktiviert hat.
Kommentar	Anmeldung eines Benutzers

2.3 Abmelden

URL	interface/rest/auth/signout
Methode	DELETE
Parameter	[accesstoken]
Antwort	Keine
HttpStatus	200 = Abmeldung erfolgreich
Kommentar	Benutzerabmeldung

2.4 Benutzerinformationen

URL	interface/rest/auth/getuser	
Methode	POST	
Parameter	[accessToken]	
Antwort	[uid]:[int] - Eindeutige Id im Vereinsflieger [firstname]:[string] - Vorname [lastname]:[string] - Nachname [memberid]:[string] -Mitgliedsnr [status]:[string] - Mitgliedsstatus [roles]: [stringarray] - Zugeordnete Rollen [email]:string - E-Mail-Adresse	
HttpStatus	200 = Ok 401 = Nicht Authentifiziert	
Kommentar	Daten zum angemeldeten Benutzer ermitteln	

3. Flugdatenerfassung

3.1 Flug anlegen

URL	interface/rest/flight/add
Methode	POST
Parameter	<p>[accesstoken]</p> <p>[callsign]:[string] - LFZ-Kennzeichen</p> <p>Optionale Parameter:</p> <p>[pilotname]:[string] - Name des Piloten</p> <p>[uidpilot]:[int] - uid des Piloten</p> <p>[attendantname] :[string] - Name des Begleiters / FI</p> <p>[uidattendant]:[int] - uid des Begleiters / FI</p> <p>[attendantname2] :[string] - Name des 2. Begleiters</p> <p>[uidattendant2]:[int] - uid des 2. Begleiters</p> <p>[attendantname3] :[string] - Name des 3. Begleiters</p> <p>[uidattendant3]:[int] - uid des 3. Begleiters</p> <p>[starttype] :[string] - Startart (E, W oder F)</p> <p>[departuretime]:[datetime] - Startzeit in UTC. Format(YYYY-mm-dd HH:ii)</p> <p>[departurelocation]:[string] - Name oder ICAO oder Koordinaten (LAT,LON)</p> <p>[arrivaltime]:[datetime] - Landezeit in UTC. Format(YYYY-mm-dd HH:ii)</p> <p>[arrivallocation]:[string] - Landeort</p> <p>[landingcount]:[int] - Anzahl der Landungen (default= 1)</p> <p>[ftid]:[int] - Flugart im Standard=10 oder 8-Schulflug</p> <p>[km]:[int] - Strecke in Km</p> <p>[chargemode]:[int] - Abrechnungsmodus(1=keine,2=Pilot,3=Begleiter,4=Gast, 5=Pilot+Begleiter, 7=Anderes Mitglied)</p> <p>[uidcharge]:[int] – id des zahlenden Mitglieds</p> <p>[comment]:[string] – Bemerkung</p> <p>[wid]:[int] – ID der Winde</p> <p>[towcallsign]:[string] – Callsign des Schleppflugzeugs</p> <p>[towpilotname]:[string] – Pilot des Schleppflugzeugs</p> <p>[towuidpilot]:[int] – uid des Schlepppiloten</p> <p>[towtime]:[int] – Schleppzeit in Minuten</p> <p>[towheight]:[int] – Schlepphöhe in Meter</p> <p>[offblock]:[time] - Offblock in UTC im Format(HH:ii)</p> <p>[onblock]:[time] - Onblock in UTC im Format(HH:ii)</p> <p>[motorstart]:[time/float] - Motorzählerstand bei Start. Angabe entweder in HH:ii oder als Industriestunde mit max 5 Nachkommastellen.</p> <p>[motorend]:[time/float] - Motorzählerstand bei Landung. Angabe entweder in HH:ii oder als Industriestunde mit max 5 Nachkommastellen.</p>
Antwort	Daten zum neu angelegten Flug (Details siehe Funktion „Flug auslesen“)
HttpStatus	<p>200 =Erfolgreich</p> <p>400= Ungültige Anfrage</p> <p>401= Nicht Authentifiziert</p>
Kommentar	Anlage eines neuen Fluges

3.2 Flug bearbeiten

URL	interface/rest/flight/edit/[flid]
Methode	PUT
Parameter	<p>Url - Parameter [flid] - Eindeutige Flugnummer - Angabe in der URL</p> <p>Post - Parameter [accesstoken]</p> <p>Optionale Post - Parameter: [callsign]:[string] - LFZ-Kennzeichen [pilotname]:[string] - Name des Piloten [uidpilot]:[int] - uid des Piloten [attendantname]:[string] - Name des Begleiters / FI [uidattendant]:[int] - uid des Begleiters / FI [attendantname2]:[string] - Name des 2. Begleiters [uidattendant2]:[int] - uid des 2. Begleiters [attendantname3]:[string] - Name des 3. Begleiters [uidattendant3]:[int] - uid des 3. Begleiters [starttype]:[string] - Startart (E, W oder F) [departuretime]:[datetime] - Startzeit in UTC. Format(YYYY-mm-dd HH:ii) [departurelocation]:[string] - Name oder ICAO oder Koordinaten (LAT,LON) [arrivaltime]:[datetime] - Landezeit in UTC. Format(YYYY-mm-dd HH:ii) [arrivallocation]:[string] - Landeort [landingcount]:[int] - Anzahl der Landungen (default= 1) [ftid]:[int] - Flugart im Standard=10 oder 8-Schulflug [km]:[int] - Strecke in Km [chargemode]:[int] - Abrechnungsmodus(1=keine,2=Pilot,3=Begleiter,4=Gast, 5=Pilot+Begleiter, 7=Anderes Mitglied) [uidcharge]:[int] – id des zahlenden Mitglieds [comment]:[string] – Bemerkung [towcallsign]:[string] – Callsign des Schleppflugzeugs [towpilotname]:[string] – Pilot des Schleppflugzeugs [towuidpilot]:[int] – uid des Schlepppiloten [towtime]:[int] – Schleppzeit in Minuten [towheight]:[int] – Schlepphöhe in Meter [offblock]:[time] - Offblock in UTC im Format(HH:ii) [onblock]:[time] - Onblock in UTC im Format(HH:ii) [motorstart]:[time/float] - Motorzählerstand bei Start. Angabe entweder in HH:ii oder als Industriestunde mit max 5 Nachkommastellen. [motorend]:[time/float] - Motorzählerstand bei Landung. Angabe entweder in HH:ii oder als Industriestunde mit max 5 Nachkommastellen.</p>
Antwort	Daten zum bearbeiteten Flug (Details siehe Funktion „Flug auslesen“)
HttpStatus	200 =Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert
Kommentar	Bearbeiten eines Fluges

3.3 Flug löschen

URL	interface/rest/flight/delete/[flid]
Methode	DELETE
Parameter	Url – Parameter [flid] - Eindeutige Flugnummer - Angabe in der URL Post - Parameter [accesstoken]
Antwort	Keine
HttpStatus	200= Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert
Kommentar	Flug löschen

3.4 Flug auslesen

URL	interface/rest/flight/get/[flid]
Methode	POST
Parameter	Url - Parameter [flid] - Eindeutige Flugnummer - Angabe in der URL Post - Parameter [accesstoken]
Antwort	[flid]:[int] - Eindeutige Flugnummer [createtime]:[datetime] - Datum der Erstellung [modifytime]:[datetime] - Datum der letzten Änderung [apid]:[int] - Eindeutige LFZ-Nummer [callsign]:[string] - LFZ-Kennzeichen [uidpilot]:[int] - Eindeutige Benutzernummer des Piloten [pilotname]:[string] - Pilot [pilotmemberid]:[int] – Mitgliedsnummer des Piloten im Verein [uidattendant]:[int] – Eindeutige Benutzernummer des Begleiters / FI [attendantname]:[string] - Begleiter / FI [attendantmemberid]:[int] – Mitgliedsnummer des Begleiters im Verein [uidattendant2]:[int] – Eindeutige Benutzernummer des 2. Begleiters [attendantname2]:[string] – Begleiter [uidattendant3]:[int] – Eindeutige Benutzernummer des 3. Begleiters [attendantname3]:[string] – Begleiter [uidfi]:[int] - Eindeutige Benutzernummer (Flugauftrag von) [finame]:[string] – Flugauftrag von [uidcharge]:[int] – Eindeutige Benutzernummer (Anderes Mitglied zahlt) [dateofflight]:[date] - Flugdatum [departuretime]:[time] - Abflugzeit [departurelocation]:[string] – Startort [arrivaltime]:[time] - Landezeit [arrivallocation]:[string] - Landeort

[flighttime]:[int] - Flugzeit
 [landingcount]:[int] - Anzahl der Landungen
 [starttype]:[int] – Startart (1=Eigenstart,3=F-Schlepp,5=Winde,7=Gummiseil,9=Fahrzeug)
 [comment]:[string] – Kommentar
 [flightmode]:[int] – Flugmodus (1=Lokal, 2=Abflug, 3=Landung, 4=Fremd)
 [ftid]:[int] – Id der Flugart (Siehe Stammdaten Flugarten)
 [offblock]:[time] - OffBlock-Zeit
 [onblock]:[time] - OnBlock-Zeit
 [blocktime]:[int] - Blockzeit
 [flidtow]:[int] – Eindeutige Flugnummer des zugehörigen Schleppfluges
 [chargemode]:[int] – Abrechnungsmodus(1=Keine, 2=Pilot, 3=Begleiter, 4=Gastflug, 5=Pilot+Begleiter, 6=Gastflug (Pilot zahlt), 7=Anderes Mitglied, 8=Über SF, 9=Landegebühr)
 [wid]:[int] - Id der Winde
 [planetype]:[varchar] – Luftfahrzeugart (Segelflugzeug, Reisemotorsegler, Ultraleicht ...)
 [invoiceinfo]:array – Fluggebühren (nach der Abrechnung verfügbar)

Invoiceinfo:

[uid]:[int] – Benutzernummer
 [adid]:[int] – ID der Buchungsnummer
 [name]:[string] - Name
 [value]:[float] – Brutto Abrechnungsbetrag

HttpStatus	200= Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert
Kommentar	Flug auslesen

3.5 Alle Flüge von heute auslesen

URL	interface/rest/flight/list/today
Methode	POST
Parameter	Post - Parameter [accesstoken]
Antwort	Auflistung aller Flüge des aktuellen Tages
HttpStatus	200= Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert
Kommentar	Auslesen aller Flüge die heute gestartet/gelandet sind.

3.6 Alle Flüge für einen Tag auslesen

URL	interface/rest/flight/list/date
Methode	POST
Parameter	Post – Parameter [accesstoken] [dateparam]:[date] – Datum im Format YYYY-mm-dd
Antwort	Auflistung aller Flüge für einen Tag
HttpStatus	200= Erfolgreich 400= Ungültige Anfrage

	401= Nicht Authentifiziert
Kommentar	Auslesen aller Flüge die an einem bestimmten Tag gestartet/gelandet sind.

3.7 Die letzten Flüge für ein LFZ auslesen

URL	interface/rest/flight/list/plane
Methode	POST
Parameter	Post – Parameter [accesstoken] [callsign]:[string] – Callsign des LFZ [count]:[int] – Anzahl der Flüge (1 – 100)
Antwort	Auflistung der letzten X Flüge zu einem Callsign
HttpStatus	200= Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert

3.8 Meine letzten Flüge auslesen

URL	interface/rest/flight/list/myflights
Methode	POST
Parameter	Post – Parameter [accesstoken] [count]:[int] – Anzahl der Flüge (1 – 1000)
Antwort	Auflistung der letzten X Flüge zu einer Person. Es werden die Flüge als Pilot und die Flüge als FI ausgegeben.
HttpStatus	200= Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert

3.9 Die letzten Flüge für eine Person auslesen

URL	interface/rest/flight/list/user
Methode	POST
Parameter	Post – Parameter [accesstoken] [uid]:[int] – Eindeutige Id der Person [count]:[int] – Anzahl der Flüge (1 – 100)
Antwort	Auflistung der letzten X Flüge zu einer Person. Es werden die Flüge als Pilot und die Flüge als FI ausgegeben.
HttpStatus	200= Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert

3.10 Zuletzt geänderte Flüge auslesen

URL	interface/rest/flight/list/modified
Methode	POST
Parameter	Post – Parameter [accesstoken] [days]:[int] – Anzahl der Tage (1 – 28)
Antwort	Auflistung der Flüge die in den letzten X Tagen geändert wurden.
HttpStatus	200= Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert

4. Kalender

4.1 Öffentlichen Kalender auslesen

URL	interface/rest/calendar/list/public
Methode	POST
Parameter	Post - Parameter [accesstoken] - optional, eine Anmeldung ist nicht notwendig [hpageaccesscode] - Homepageaccesscode. Dieser ist zu finden in der Administration/Homepageerweiterungen/Kalender
Antwort	[title]:[string] - Titel [datefrom]:[datetime] - ab [dateto]:[datetime] - bis [wholeday]:[int] - ganztägig = 1 [location]:[string] - Ort [comment]:[string] - Kommentar [daterange]:[string] - Zeitraum im Klartext [duration]:[string] - Dauer im Klartext
HttpStatus	200= Erfolgreich 400= Ungültige Anfrage
Kommentar	Auslesen des öffentlichen Kalenders

5. Mitglieder

5.1 Auslesen der Mitgliederliste

URL	interface/rest/user/list
Methode	POST
Parameter	Post - Parameter [accesstoken] Hinweis: Das Recht ‚Mitgliederdaten bearbeiten‘ wird benötigt.
Antwort	[uid]:[int] - Interne ID [title]:[string] - Titel [firstname]:[string] - Vorname [lastname]:[string] - Nachname [careof]:[string] - Adresszusatz [street]:[string] - Straße [zipcode]:[string] - PLZ [town]:[string] - Ort [email]:[string] - Mailadresse [gender]:[string] - Geschlecht (m, w) [birthday]:[date] - Geburtsdatum [birthplace]:[string] - Geburtsort [homenumber]:[string] - Telefon (privat) [mobilenumber]:[string] - Mobil (privat) [phonenumber]:[string] - Telefon (gesch.) [phonenumber2]:[string] - Mobil (gesch.) [carlicenseplate]:[string] - Autokennzeichen [identification]:[string] - Ausweisnummer [natoid]:[string] - NatoID [policecert_validto]:[date] - Führungszeugnis [ice_contact1]:[string] - Notfallkontakt 1 [ice_contact2]:[string] - Notfallkontakt 2 [memberid]:[int] - MitgliedsNr [memberbegin]:[date] - Eintrittsdatum [memberend]:[date] - Austrittsdatum [lettertitle]:[string] - Briefanrede [memberstatus]:[string] - Mitgliedsstatus [country]:[string] - Land [bankaccountname]:[string] - Kontoinhaber [directdebitauth]:[int] - Lastschrift ja = 1 [iban]:[string] - IBAN [bic]:[string] - BIC [mandate]:[string] - Mandat [roles]: [stringarray] - Zugeordnete Rollen [mandatedate]:[date] - Mandatsdatum [mailrecipient]:[int] - Rundmailempfänger [sector]:[stringarray] - Zugeordnete Sparten [functions]:[stringarray] - Zugeordnete Funktionen [educations]:[stringarray] - Nicht beendete Lehrpläne [prop(x)]:[array] - Weitere Eigenschaften, wobei (x) der interne Index der Eigenschaft ist. Das Array beinhaltet die Bezeichnung und den Wert der Eigenschaft [accounts]:[array] - Liste der Sachkonten. [keymanagement]:[array] - Liste der zugeordneten Schlüssel
HttpStatus	200= Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert
Kommentar	Auslesen der Mitgliederliste

6. Reservierungen

6.1 Aktuelle Reservierungen auslesen

URL	interface/rest/reservation/list/actice
Methode	POST
Parameter	Post - Parameter [accesstoken]
Antwort	[prid]:[int] - ReservierungsId [datefrom]:[datetime] – Beginn der Reservierung [dateto]:[datetime] – Ende der Reservierung [comment]:[string] – Kommentar [freeseats]:[int] – Freie Sitze [user]:[string] – Pilot [fi]:[string] - Fluglehrer [type]:[string] –Art der Reservierung [ressource]:[string] – Bezeichnung der Ressource [daterange]:[string] – Zeitraum im Klartext [duration]:[string] – Dauer im Klartext
HttpStatus	200= Erfolgreich 400= Ungültige Anfrage
Kommentar	Auslesen der Reservierungen

7. Instandhaltung

7.1 Aktuelle Zellenzeiten eines LFZs auslesen

URL	interface/rest/maintenance/airplane/[callsign]
Methode	POST
Parameter	Post - Parameter [accesstoken] [callsign]:string: - Callsign des LFZs
Antwort	[motortime]:[float] – Aktueller Motorzählerstand [flighttime]:[float] – Aktueller Flugzeitenstand [landingcount]:[int] – Aktuelle Anzahl der Landungen
HttpStatus	200= Erfolgreich 400= Ungültige Anfrage
Kommentar	Auslesen der Zellenzeit zu einem LFZ

8. Finanzen - Buchungen

8.1 Buchung anlegen

URL	interface/rest/account/add
Methode	POST
Parameter	[accesstoken] [bookingdate] :[date] – Buchungsdatum im Format (YYYY-mm-dd) [value]:[float] – Bruttobetrag(muss > 0,00 sein) [salestax]:[float] – Mehrwertsteuer in % (gültiger MwSt- Eintrag aus den Stammdaten) [debitaccount]:[string] – Sachkontennummer Sollkonto [creditaccount]:[string] - Sachkontennummer Habenkonto [taxaccount]:[string] – Sachkontennummer Steuerkonto [accountreference]:[string] – Belegart (Kurzbezeichnung aus den Stammdaten) [accountreferenceid]:[int] – Belegnr [bookingtext] :[string] – Buchungstext
Antwort	Neu angelegte Buchung
HttpStatus	200 =Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert
Kommentar	Anlage einer neuen Buchung. Voraussetzung ist der Buchhaltungsmodus Version2

8.2 Einzelne Buchung auslesen

URL	interface/rest/account/get/[adid]
Methode	POST
Parameter	Url - Parameter [adid] - Eindeutige Buchungsnummer - Angabe in der URL Post - Parameter [accesstoken]
Antwort	[adid]:[int] – Buchungsnummer [createtime]:[datetime] – Erstellungsdatum [modifytime] :[datetime] – Letztes Änderungsdatum [bookingdate] :[date] - Buchungsdatum [bookingtext] :[string] – Buchungstext [value]:[float] – Wert [netvalue]:[float] - Nettowert [salestax]:[float] – Mehrwertsteuer in % [debitaccount]:[string] - Sollkonto [creditaccount]:[string] - Habenkonto [taxaccount]:[string] – Steuerkonto [accountreference]:[string] – Belegart [accountreferenceid]:[int] – Belegnr [bookingerror]:[int] – Wert =1 -> Buchung ist fehlerhaft
HttpStatus	200= Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert

Kommentar	Buchung auslesen
-----------	------------------

8.3 Alle Buchungen mit heutigem Buchungsdatum auslesen

URL	interface/rest/account/list/today
Methode	POST
Parameter	Post - Parameter [accesstoken]
Antwort	Auflistung aller Buchungen mit heutigem Buchungsdatum
HttpStatus	200= Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert
Kommentar	Auslesen aller Buchungen, die das aktuelle Datum als Buchungsdatum haben.

8.4 Alle Buchungen zu einem Jahr auslesen

URL	interface/rest/account/list/year
Methode	POST
Parameter	Post - Parameter [accesstoken] [year]:[int] Buchungsjahr
Antwort	Auflistung aller Buchungen zu einem Jahr
HttpStatus	200= Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert
Kommentar	Auslesen aller Buchungen, die in einem bestimmten Jahr liegen.

8.5 Alle Buchungen für einen Zeitraum auslesen

URL	interface/rest/account/list/daterange
Methode	POST
Parameter	Post - Parameter [accesstoken] [datefrom]:[date] - Datum im Format YYYY-mm-dd [dateto]:[date] – Datum im Format YYYY-mm-dd
Antwort	Auflistung aller Buchungen für einen Zeitraum
HttpStatus	200= Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert
Kommentar	Auslesen aller Buchungen deren Buchungsdatum in einem bestimmten Zeitraum liegen.

9. Arbeitsstunden

9.1 Arbeitsstunden auslesen

URL	interface/rest/workhours/list/daterange
Methode	POST
Parameter	Post - Parameter [accesstoken] [datefrom]:[date] - Datum im Format YYYY-mm-dd [dateto]:[date] – Datum im Format YYYY-mm-dd
Antwort	[whid]:[int] - Eindeutige ID [createtime]:[Datetime] - Erstellungsdatum [modifytime]:[Datetime] - Letzte Änderung [uid]:[int] - ID des Person [username]:[string] - Name [jobdate]:[Date] - Arbeitsstundendatum [jobtext]:[string] - Durchgeführte Tätigkeit [hours]:[Time] - Dauer der Arbeit [timefrom]:[Time] - Zeitraum von [timeto]:[Time] - Zeitraum bis [status]:[int] - Stautus [statusinfo]:[string] - Statusbezeichnung [category]:[int] - Kategorie [categoryinfo]:[string] - Kategoriebezeichnung [comment]:[string] - Kommentar
HttpStatus	200= Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert
Kommentar	Auslesen aller Arbeitsstunden deren Datum in einem bestimmten Zeitraum liegen.

9.2 Arbeitsstunden anlegen

URL	interface/rest/workhours/add
Methode	POST
Parameter	[accesstoken] [uid]:[int] – Id der Person [jobdate]:[date] – Datum im Format (YYYY-mm-dd) [jobtext]:[string] –Tätigkeit [hours]:[float] – Stunden im Format HH:ii [category]:[int] – Id der Arbeitsstundenkategorie Optionale Felder: [timefrom]:[time] - Uhrzeit ab im Format HH:ii [timeto]:[time] - Uhrzeit ab im Format HH:ii [status]:[int] – 1= Angelgt, 2=Akzeptiert, 3=abgelehnt [comment]:[string] – Kommentar
Antwort	Neu angelegter Datensatz
HttpStatus	200 =Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert
Kommentar	Anlage eines neuen Arbeitsstundendatensatzes

9.3 Arbeitsstundenkategorien auslesen

URL	interface/rest/workhourcategories/list
Methode	POST
Parameter	Post - Parameter [accesstoken]
Antwort	[category]:[int] – Eindeutige Id der Kategorie [name]:[string] - Bezeichnung
HttpStatus	200= Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert
Kommentar	Auslesen aller Arbeitsstundenkategorien

10. Allgemeiner Verkauf

10.1 Artikelliste auslesen

URL	interface/rest/articles/list
Methode	POST
Parameter	Post - Parameter [accesstoken]
Antwort	[articleid]:[string] – Artikelnr [designation]:[string] – Artikelbeschreibung [unitttype]:[string] - Artikeleinheit [caid]:[Sachkonto] – Sachkonto [prices]:[array] [validfrom]:[date] – Gültig ab [validto]:[date] – Gültig bis [salestax]:[float] – Mehrwertsteuer [unitprice]:[float] - Bruttopreis
HttpStatus	200= Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert
Kommentar	Auslesen aller Artikel

10.2 Verkauf anlegen

URL	interface/rest/sale/add
Methode	POST
Parameter	[accesstoken] [bookingdate]:[date] – Datum im Format YYYY-mm-dd [articleid]:[string] – Artikelnummer [amount]:[float] - Menge Optionale Felder: [memberid]:[int] - Mitgliedsnummer des Käufers [callsign]:[string] – Callsign bzw. Verwendung [salestax]:[float] - Steuer [totalprice]:[float] - Bruttopreis [counter]:[float] - Zählerstand [comment]:[string] – Kommentar [ccid]:[string] – Gebührenbereich
Antwort	Neu angelegter Datensatz
HttpStatus	200 =Erfolgreich 400= Ungültige Anfrage 401= Nicht Authentifiziert
Kommentar	Anlage eines neuen Verkaufs

11. Kontakt

Vereinsflieger.de

Anne-Frank-Straße 5

35428 Langgöns

E-Mail: info@vereinsflieger.de