

API Reference

TaskLauncher

API Version: v1

INDEX

1. AUTH	5
1.1 GET /auth/login	5
1.2 GET /auth/logout	5
1.3 POST /auth/reset-password	5
1.4 GET /auth/user	5
1.5 POST /passwordflow/login	6
1.6 POST /passwordflow/register	6
1.7 GET /passwordflow/refresh/{token}	8
1.8 POST /auth/signup	9
1.9 POST /auth/verify-email	9
2. BANS	11
2.1 GET /api/admin/bans	11
2.2 POST /api/admin/bans	11
2.3 GET /api/admin/bans/{id}	12
2.4 DELETE /api/admin/bans/{id}	12
2.5 POST /api/admin/bans/{id}/cancel	13
3. CONFIG	14
3.1 GET /api/worker/config	14
3.2 GET /api/admin/config	14
3.3 PUT /api/admin/config	15
4. PAYMENTS	16
4.1 GET /api/payments	16
4.2 GET /api/admin/payments	16
5. STATS	18
5.1 GET /api/stats	18
5.2 GET /api/stats/times	18
5.3 GET /api/stats/taskcountperdays	19
5.4 GET /api/admin/stats	19
5.5 GET /api/admin/stats/times	20
5.6 GET /api/admin/stats/taskcountperdays	21
6. TASKS	22
6.1 GET /api/tasks/{id}/file	22
6.2 GET /api/tasks	22
6.3 POST /api/tasks	23
6.4 GET /api/tasks/{id}	23
6.5 DELETE /api/tasks/{id}	24
6.6 PATCH /api/tasks/{id}	24
6.7 POST /api/tasks/{id}/close	25
6.8 POST /api/tasks/{id}/cancel	26

6.9	POST	/api/tasks/{id}/restart	26
6.10	GET	/api/admin/tasks	27
6.11	GET	/api/admin/tasks/{id}	27
6.12	DELETE	/api/admin/tasks/{id}	28
7.	TOKEN		29
7.1	GET	/api/admin/token	29
7.2	PUT	/api/admin/token	29
7.3	GET	/api/token	29
8.	USER		31
8.1	GET	/api/user	31
8.2	DELETE	/api/user	32
8.3	PATCH	/api/user	33
8.4	GET	/api/user/admin-contact	34
9.	WORKER		36
9.1	GET	/api/worker	36
9.2	POST	/api/worker	36

Security and Authentication

SECURITY SCHEMES

KEY	TYPE	DESCRIPTION
Bearer	http, bearer	Using the Authorization header with the Bearer scheme.

API

1. AUTH

1.1 GET /auth/login

Přihlášení přes službu Auth0, uživatel je přesměrován na Auth0, pouze z prohlížeče (cookie autentizace)

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Success

1.2 GET /auth/logout

Odhlášení uživatele (cookie autentizace)

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Success

1.3 POST /auth/reset-password

Resetování hesla (cookie autentizace)

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{  
  message string Zpráva od auth0 o změnění hesla  
}
```

1.4 GET /auth/user

Získání uživatelských dat přihlášeného uživatele na základě http kontextu

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  isAuthenticated boolean  Určuje, zda je uživatel přihlášen
  nameClaimType   string   Typ claimu se jménem
  claims [{
    Array of object:
      type string  Typ nároku
      value string Hodnota nároku
  }]
}
```

1.5 POST /passwordflow/login

Přihlašování pro desktopové nebo mobilní aplikace

REQUEST

REQUEST BODY - application/json

```
{
  name      string  Jméno nebo email uživatele
  password  string  Heslo
}
```

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  access_token  string
  refresh_token string
  id_token      string
  token_type    string
  expires_in    integer
}
```

1.6 POST /passwordflow/register

Registrace uživatele, pro desktopové nebo mobilní aplikace

REQUEST

REQUEST BODY - application/json

```
{
  firstName  string  Jméno
  lastName   string  Příjmení
  nickName   string  Přezdívka
  phoneNumber string  Telefonní číslo
  picture    string  Url adresa obrázku
  email      string  Email
  password   string
}
```

}

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  appMetadata      undefined
  email            string
  emailVerified    boolean
  phoneNumber      string
  phoneVerified    boolean
  userMetadata     undefined
  userName         string
  nickName        string
  firstName        string
  fullName         string
  lastName         string
  picture          string
  blocked          boolean
  createdAt        string
  identities [{
    Array of object:
      accessToken    string
      accessTokenSecret string
      connection     string
      expiresIn      integer
      isSocial       boolean
      profileData {
      }
      provider       string
      refreshToken   string
      userId         string
    }]
  lastIpAddress    string
  lastLogin        string
  lastPasswordReset string
  locale           string
  loginsCount      string
  updatedAt        string
  userId           string
  providerAttributes {
  }
  multifactor      [string]
  original {
    appMetadata      undefined
    email            string
    emailVerified    boolean
    phoneNumber      string
    phoneVerified    boolean
    userMetadata     undefined
    userName         string
    nickName        string
    firstName        string
    fullName         string
    lastName         string
  }
```

```

picture                string
blocked                boolean
createdAt              string
identities [{
  Array of object:
    accessToken         string
    accessTokenSecret   string
    connection          string
    expiresIn           integer
    isSocial            boolean
    profileData {
    }
    provider            string
    refreshToken        string
    userId              string
  }]
lastIpAddress          string
lastLogin              string
lastPasswordReset      string
locale                 string
loginsCount            string
updatedAt              string
userId                 string
providerAttributes {
}
multifactor            [string]
}
vip                    boolean
registered             boolean
isAdmin                boolean
tokenBalance           string
}

```

1.7 GET /passwordflow/refresh/{token}

Aktualizuje přístupový token, pro desktopové nebo mobilní aplikace

REQUEST

PATH PARAMETERS

NAME	TYPE	DESCRIPTION
*token	string	

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```

{
  access_token string
  scope         string
  token_type    string
  expires_in    integer
}

```


1.8 POST /auth/signup

Dokončení registrace na webové stránce, pouze pro neregistrované uživatele

REQUEST

REQUEST BODY - application/json

```
{
  firstName    string  Jméno
  lastName     string  Příjmení
  nickName     string  Přezdívka
  phoneNumber  string  Telefonní číslo
  picture      string  Url adresa obrázku
  email        string  Email
}
```

RESPONSE

STATUS CODE - 200: Success

1.9 POST /auth/verify-email

Posílá verifikační email

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  status      string
  type        string
  id          string
  connection  string
  connectionId string
  createdAt   string
  summary {
    failed    integer
    updated   integer
    inserted  integer
    total     integer
  }
  format      string
  location    string
  percentageDone integer
  timeLeftSeconds integer
  timeLeft {
    ticks      integer
    days       integer READ-ONLY
    hours      integer READ-ONLY
    milliseconds integer READ-ONLY
    minutes    integer READ-ONLY
  }
}
```

seconds	integer	READ-ONLY
totalDays	number	READ-ONLY
totalHours	number	READ-ONLY
totalMilliseconds	number	READ-ONLY
totalMinutes	number	READ-ONLY
totalSeconds	number	READ-ONLY

```
}  
}
```

2. BANS

2.1 GET /api/admin/bans

Vrací všechny blokace v systému, dotazuje se přes odata

REQUEST

QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
\$select	string	Jméno některé vlastnosti daného objektu
\$filter	string	Filtrování podle podmínky, více zde https://www.odata.org/getting-started/basic-tutorial/#filter
\$top	string	Maximální počet objektů v odpovědi
\$skip	string	Přeskočí určitý počet objektů https://www.odata.org/getting-started/basic-tutorial/#select
\$orderby	string	Seřazení objektů
\$count	boolean	Pokud je nastaven na true, vrací se, kolik objektů je v kolekci

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
[{  
  Array of object:  
    id          string  Id blokace  
    started     string  Kdy začala blokace platit  
    ended       string  Kdy se blokace zrušila  
    description string  Popis blokace  
    email       string  
    userId      string  Id blokováného uživatele  
}]
```

2.2 POST /api/admin/bans

Vytvoření nové blokace

REQUEST

REQUEST BODY - application/json

```
{  
  userId string  Id uživatele  
  reason string  Důvod udělení blokace  
}
```

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  id            string  Id blokace
  started       string  Kdy začala blokace platit
  ended         string  Kdy se blokace zrušila
  description   string  Popis blokace
  email         string
  userId        string  Id blokovaneho uzivatele
}
```

STATUS CODE - 400: Bad Request

RESPONSE MODEL - application/json

```
{
  error string  Chyba
}
```

2.3 GET /api/admin/bans/{id}

Vrací detail blokace

REQUEST

PATH PARAMETERS

NAME	TYPE	DESCRIPTION
*id	uuid	Id banu

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  id            string  Id blokace
  started       string  Kdy začala blokace platit
  ended         string  Kdy se blokace zrušila
  description   string  Popis blokace
  email         string
  userId        string  Id blokovaneho uzivatele
}
```

STATUS CODE - 404: Not Found

2.4 DELETE /api/admin/bans/{id}

Smazání záznamu blokace ze systému (blokace musí být nejdříve zrušená)

REQUEST

PATH PARAMETERS

NAME	TYPE	DESCRIPTION
*id	uuid	Id banu

RESPONSE

STATUS CODE - 200: Success

STATUS CODE - 400: Bad Request

RESPONSE MODEL - application/json

```
{
  error string Chyba
}
```

STATUS CODE - 404: Not Found

2.5 POST /api/admin/bans/{id}/cancel

Zrušení blokace

REQUEST

PATH PARAMETERS

NAME	TYPE	DESCRIPTION
*id	uuid	Id banu

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  id            string  Id blokace
  started       string  Kdy začala blokace platit
  ended         string  Kdy se blokace zrušila
  description    string  Popis blokace
  email         string
  userId        string  Id blokováného uživatele
}
```

STATUS CODE - 400: Bad Request

RESPONSE MODEL - application/json

```
{
  error string Chyba
}
```

STATUS CODE - 404: Not Found

3. CONFIG

3.1 GET /api/worker/config

Získá všechny konfigurační hodnoty nebo pouze jednu hodnotu, pokud je specifikován parametr 'key' (pro workera)

REQUEST

QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
key	string	Identifikátor proměnné

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  key          string  Identifikátor konfigurační proměnné
  type         enum    ALLOWED:0, 1
  value        string  Hodnota proměnné
  description  string  Popis proměnné
}
```

STATUS CODE - 400: Bad Request

RESPONSE MODEL - application/json

```
{
  error string  Chyba
}
```

3.2 GET /api/admin/config

Získá všechny konfigurační hodnoty nebo pouze jednu hodnotu, pokud je specifikován parametr 'key' (pro administrátora)

REQUEST

QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
key	string	Identifikátor proměnné

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
```

```
key          string  Identifikátor konfigurační proměnné
type         enum    ALLOWED:0, 1
value        string  Hodnota proměnné
description  string  Popis proměnné
}
```

STATUS CODE - 400: Bad Request

RESPONSE MODEL - application/json

```
{
  error string Chyba
}
```

3.3 PUT /api/admin/config

Aktualizuje systémovou konfigurační proměnnou

REQUEST

REQUEST BODY - application/json

```
{
  key   string  Klíč proměnné
  value string  Hodnota proměnné
}
```

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  key          string  Identifikátor konfigurační proměnné
  type         enum    ALLOWED:0, 1
  value        string  Hodnota proměnné
  description  string  Popis proměnné
}
```

STATUS CODE - 400: Bad Request

RESPONSE MODEL - application/json

```
{
  error string Chyba
}
```

4. PAYMENTS

4.1 GET /api/payments

Vrací všechny platby uživatele, dotazuje se přes protokol odata

REQUEST

QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
\$select	string	Jméno některé vlastnosti daného objektu
\$filter	string	Filtrování podle podmínky, více zde https://www.odata.org/getting-started/basic-tutorial/#filter
\$top	string	Maximální počet objektů v odpovědi
\$skip	string	Přeskočí určitý počet objektů https://www.odata.org/getting-started/basic-tutorial/#select
\$orderby	string	Seřazení objektů
\$count	boolean	Pokud je nastaven na true, vrací se, kolik objektů je v kolekci

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
[{  
  Array of object:  
    id      string  Id platby  
    userId  string  Id platícího uživatele  
    taskId  string  Id úlohy na kterou se platba vztahuje  
    time    string  Čas provedení platby  
    price   number  Cena  
}]
```

4.2 GET /api/admin/payments

Vrací všechny platby v systému, dotazuje se přes odata

REQUEST

QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
\$select	string	Jméno některé vlastnosti daného objektu
\$filter	string	Filtrování podle podmínky, více zde https://www.odata.org/getting-started/basic-tutorial/#filter
\$top	string	Maximální počet objektů v odpovědi
\$skip	string	Přeskočí určitý počet objektů https://www.odata.org/getting-started/basic-tutorial/#select
\$orderby	string	Seřazení objektů

NAME	TYPE	DESCRIPTION
\$count	boolean	Pokud je nastaven na true, vrací se, kolik objektů je v kolekci

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
[{  
  Array of object:  
    id      string  Id platby  
    userId  string  Id platícího uživatele  
    taskId  string  Id úlohy na kterou se platba vztahuje  
    time    string  Čas provedení platby  
    price   number  Cena  
}]
```

5. STATS

5.1 GET /api/stats

Vrací obecné statistiky pro přihlášeného uživatele

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
[{  
  Array of object:  
    allTaskCount      integer  Počet všech úloh (i smazaných), které uživatel vytvořil  
    finishedTaskCount integer  Celkový počet dokončených úloh (úspěšně a neúspěšně)  
    failedTasks        integer  Celkový počet dokončených neúspěšných úloh  
    successTasks       integer  Celkový počet dokončených úspěšných úloh  
    timeoutedTasks     integer  Celkový počet úloh, které se nestihly dokončit  
    crashedTasks       integer  Celkový počet zhavarovaných úloh  
    isVip              boolean  Určuje, zda jsou statistiky určeny z prioritních úloh  
}]
```

5.2 GET /api/stats/times

Vrací se kolekce statistik posledních 30 úloh za poslední den od přihlášeného uživatele. Statistiky udávají, kolik času daná úloha strávila ve workeru a jak dlouho se počítala

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
[{  
  Array of object:  
    isVip      boolean  Určuje, zda byla úloha prioritní  
    taskName   string   Jméno úlohy  
    timeInQueue {  
      ticks      integer  
      days       integer  READ-ONLY  
      hours      integer  READ-ONLY  
      milliseconds integer  READ-ONLY  
      minutes    integer  READ-ONLY  
      seconds    integer  READ-ONLY  
      totalDays  number   READ-ONLY  
      totalHours number   READ-ONLY  
      totalMilliseconds number READ-ONLY  
    }  
}]
```

```

        totalMinutes    number    READ-ONLY
        totalSeconds    number    READ-ONLY
    }
    cpuTime {
        ticks            integer
        days              integer    READ-ONLY
        hours             integer    READ-ONLY
        milliseconds     integer    READ-ONLY
        minutes           integer    READ-ONLY
        seconds           integer    READ-ONLY
        totalDays         number    READ-ONLY
        totalHours        number    READ-ONLY
        totalMilliseconds number    READ-ONLY
        totalMinutes      number    READ-ONLY
        totalSeconds      number    READ-ONLY
    }
}
}]

```

5.3 GET /api/stats/taskcountperdays

Vrací kolekci, kde položka je počet úloh spuštěných za den, jsou to úlohy za posledních 30 dní. Musí se specifikovat, zda se budou vracet vip nebo normalní úlohy

REQUEST

QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
vip	boolean	Vip

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```

[ {
  Array of object:
    count integer    Počet úloh
    date   string    Den
  } ]

```

5.4 GET /api/admin/stats

Vrací obecné seskupené statistiky od všech uživatelů nebo obecné statistiky zadaného uživatele přes parametr userId

REQUEST

QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
userId	string	Id uživatele

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
[{  
  Array of object:  
    allTaskCount      integer  Počet všech úloh (i smazaných), které uživatel vytvořil  
    finishedTaskCount integer  Celkový počet dokončených úloh (úspěšně a neúspěšně)  
    failedTasks        integer  Celkový počet dokončených neúspěšných úloh  
    successTasks       integer  Celkový počet dokončených úspěšných úloh  
    timeoutedTasks     integer  Celkový počet úloh, které se nestihly dokončit  
    crashedTasks       integer  Celkový počet zhavarovaných úloh  
    isVip              boolean  Určuje, zda jsou statistiky určeny z prioritních úloh  
}]
```

5.5 GET /api/admin/stats/times

Vrací se kolekce statistik posledních 30 úloh za poslední den od zadaného uživatele. Pokud se přes parametr `userId` nespecifikuje uživatel, vrací se kolekce statistik posledních 50 úloh celého systému za poslední den. Statistiky udávají, kolik času daná úloha strávila ve frontě a ve worker aplikaci

REQUEST

QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
userId	string	Id uživatele

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
[{  
  Array of object:  
    isVip      boolean  Určuje, zda byla úloha prioritní  
    taskName   string   Jméno úlohy  
    timeInQueue {  
      ticks      integer  
      days       integer  READ-ONLY  
      hours      integer  READ-ONLY  
      milliseconds integer  READ-ONLY  
      minutes    integer  READ-ONLY  
      seconds    integer  READ-ONLY  
      totalDays  number   READ-ONLY  
      totalHours number   READ-ONLY  
      totalMilliseconds number READ-ONLY  
      totalMinutes number   READ-ONLY  
      totalSeconds number   READ-ONLY  
    }  
    cpuTime {  
      ticks      integer  
      days       integer  READ-ONLY  
      hours      integer  READ-ONLY  
    }  
  }  
]
```

```

        milliseconds integer READ-ONLY
        minutes integer READ-ONLY
        seconds integer READ-ONLY
        totalDays number READ-ONLY
        totalHours number READ-ONLY
        totalMilliseconds number READ-ONLY
        totalMinutes number READ-ONLY
        totalSeconds number READ-ONLY
    }
}]

```

5.6 GET /api/admin/stats/taskcountperdays

Vrací se kolekce, kde položka je počet založených úloh za den. Vrací se založené úlohy za posledních 30 dní od zadaného uživatele, pokud není specifikován, vrátí se úlohy od všech uživatelů za posledních 30 dní. Musí se specifikovat, zda se budou vracet vip nebo normalní úlohy

REQUEST

QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
vip	boolean	Vip
userId	string	Id uživatele

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```

[ {
  Array of object:
    count integer Počet úloh
    date string Den
} ]

```

6. TASKS

6.1 GET /api/tasks/{id}/file

Vrací výsledný soubor dané úlohy

REQUEST

PATH PARAMETERS

NAME	TYPE	DESCRIPTION
*id	uuid	Id úlohy

RESPONSE

STATUS CODE - 200: Success

STATUS CODE - 400: Bad Request

6.2 GET /api/tasks

Zpřístupňuje dotazování přes odata nad celou kolekcí úloh přihlášeného uživatele

REQUEST

QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
\$select	string	Jméno některé vlastnosti daného objektu
\$filter	string	Filtrování podle podmínky, více zde https://www.odata.org/getting-started/basic-tutorial/#filter
\$top	string	Maximální počet objektů v odpovědi
\$skip	string	Přeskočí určitý počet objektů https://www.odata.org/getting-started/basic-tutorial/#select
\$orderby	string	Seřazení objektů
\$count	boolean	Pokud je nastaven na true, vrací se, kolik objektů je v kolekci

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
[{
```

Array of object:

id	string	Id úlohy
resultFile	string	Cesta k souboru definující úlohu
userId	string	Id uživatele
name	string	Jméno úlohy
description	string	Popis úlohy
creationDate	string	Čas vytvoření úlohy
actualStatus	enum	ALLOWED:0, 1, 2, 3, 4, 5, 6, 7, 8, 9

```
    isPriority    boolean    Určuje prioritu
  }]
```

6.3 POST /api/tasks

Vytvoření nové úlohy

REQUEST

FORM DATA PARAMETERS

NAME	TYPE	DESCRIPTION
Name	string	Jméno nové úlohy
Description	string	Popis úlohy
file	string(binary)	

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  id            string    Id úlohy
  resultFile    string    Cesta k souboru definující úlohu
  userId        string    Id uživatele
  name          string    Jméno úlohy
  description    string    Popis úlohy
  creationDate  string    Čas vytvoření úlohy
  actualStatus  enum      ALLOWED:0, 1, 2, 3, 4, 5, 6, 7, 8, 9
  isPriority     boolean   Určuje prioritu
}
```

STATUS CODE - 400: Bad Request

RESPONSE MODEL - application/json

```
{
  error string    Chyba
}
```

STATUS CODE - 401: Unauthorized

6.4 GET /api/tasks/{id}

Vrací detail úlohy společně s platbou a všemi událostmi

REQUEST

PATH PARAMETERS

NAME	TYPE	DESCRIPTION
*id	uuid	Id úlohy

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  id            string    Id úlohy
  resultFile    string    Cesta k souboru definující úlohu
  userId        string    Id uživatele
  name          string    Jméno úlohy
  description    string    Popis úlohy
  creationDate  string    Čas vytvoření úlohy
  actualStatus  enum      ALLOWED:0, 1, 2, 3, 4, 5, 6, 7, 8, 9
  isPriority     boolean   Určuje prioritu
  events [{
    Array of object:
      status     enum      ALLOWED:0, 1, 2, 3, 4, 5, 6, 7, 8, 9
      time       string    Čas vytvoření události
      description string    Popis události
    }]
  payment {
    time string  Čas potvrzení platby
    price number Uhrazená částka
  }
}
```

STATUS CODE - 404: Not Found

6.5 DELETE /api/tasks/{id}

Smazání úlohy, neposílá se žádná informace v těle dotazu

REQUEST

PATH PARAMETERS

NAME	TYPE	DESCRIPTION
*id	uuid	Id úlohy

RESPONSE

STATUS CODE - 200: Success

STATUS CODE - 404: Not Found

6.6 PATCH /api/tasks/{id}

Aktualizace informací úlohy

REQUEST

PATH PARAMETERS

NAME	TYPE	DESCRIPTION
*id	uuid	Id úlohy

REQUEST BODY - application/json-patch+json

```
[{  
  Array of object:  
    op    string  
    value {  
    }  
    path  string  
  }  
}]
```

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{  
  id            string  Id úlohy  
  resultFile    string  Cesta k souboru definující úlohu  
  userId        string  Id uživatele  
  name          string  Jméno úlohy  
  description    string  Popis úlohy  
  creationDate  string  Čas vytvoření úlohy  
  actualStatus  enum    ALLOWED:0, 1, 2, 3, 4, 5, 6, 7, 8, 9  
  isPriority     boolean Určuje prioritu  
}
```

STATUS CODE - 404: Not Found

6.7 POST /api/tasks/{id}/close

Příkaz o uzavření úlohy, neposílá se žádná informace v těle dotazu

REQUEST

PATH PARAMETERS

NAME	TYPE	DESCRIPTION
*id	uuid	Id úlohy

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{  
  status      enum    ALLOWED:0, 1, 2, 3, 4, 5, 6, 7, 8, 9  
  time        string  Čas vytvoření události  
  description  string  Popis události  
}
```

STATUS CODE - 400: Bad Request

RESPONSE MODEL - application/json

```
{  
  error string Chyba  
}
```

STATUS CODE - 401: Unauthorized

STATUS CODE - 404: Not Found

6.8 POST /api/tasks/{id}/cancel

Příkaz o zrušení úlohy a vrácení tokenu, neposílá se žádná informace v těle dotazu

REQUEST

PATH PARAMETERS

NAME	TYPE	DESCRIPTION
*id	uuid	Id úlohy

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  status      enum    ALLOWED:0, 1, 2, 3, 4, 5, 6, 7, 8, 9
  time        string   Čas vytvoření události
  description string   Popis události
}
```

STATUS CODE - 400: Bad Request

RESPONSE MODEL - application/json

```
{
  error string Chyba
}
```

STATUS CODE - 401: Unauthorized

STATUS CODE - 404: Not Found

6.9 POST /api/tasks/{id}/restart

Restartuje úlohu, pokud je ve stavu Nedokončeno, úloha ve stavu Zhavarováno se restartuje automaticky, zrušená úloha nemůže být restartována

REQUEST

PATH PARAMETERS

NAME	TYPE	DESCRIPTION
*id	uuid	Id úlohy

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  status      enum    ALLOWED:0, 1, 2, 3, 4, 5, 6, 7, 8, 9
  time        string   Čas vytvoření události
  description string   Popis události
}
```

STATUS CODE - 400: Bad Request

RESPONSE MODEL - application/json

```
{
  error string Chyba
}
```

STATUS CODE - 401: Unauthorized

STATUS CODE - 404: Not Found

6.10 GET /api/admin/tasks

Vrací všechny úlohy v systému, dotazuje se přes odata

REQUEST

QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
\$select	string	Jméno některé vlastnosti daného objektu
\$filter	string	Filtrování podle podmínky, více zde https://www.odata.org/getting-started/basic-tutorial/#filter
\$top	string	Maximální počet objektů v odpovědi
\$skip	string	Přeskočí určitý počet objektů https://www.odata.org/getting-started/basic-tutorial/#select
\$orderby	string	Seřazení objektů
\$count	boolean	Pokud je nastaven na true, vrací se, kolik objektů je v kolekci

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
[{
  Array of object:
    id          string   Id úlohy
    resultFile  string   Cesta k souboru definující úlohu
    userId      string   Id uživatele
    name        string   Jméno úlohy
    description  string   Popis úlohy
    createDate  string   Čas vytvoření úlohy
    actualStatus enum     ALLOWED:0, 1, 2, 3, 4, 5, 6, 7, 8, 9
    isPriority   boolean  Určuje prioritu
  }]
```

6.11 GET /api/admin/tasks/{id}

Zobrazí detail jakékoli úlohy v systému

REQUEST

PATH PARAMETERS

NAME	TYPE	DESCRIPTION
*id	uuid	Id úlohy

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  id                string    Id úlohy
  resultFile        string    Cesta k souboru definující úlohu
  userId            string    Id uživatele
  name              string    Jméno úlohy
  description        string    Popis úlohy
  creationDate       string    Čas vytvoření úlohy
  actualStatus       enum      ALLOWED:0, 1, 2, 3, 4, 5, 6, 7, 8, 9
  isPriority         boolean    Určuje prioritu
  events [{
    Array of object:
      status         enum      ALLOWED:0, 1, 2, 3, 4, 5, 6, 7, 8, 9
      time            string    Čas vytvoření události
      description     string    Popis události
    }]
  payment {
    time string    Čas potvrzení platby
    price number    Uhrazená částka
  }
}
```

STATUS CODE - 404: Not Found

6.12 DELETE /api/admin/tasks/{id}

Smazání úlohy

REQUEST

PATH PARAMETERS

NAME	TYPE	DESCRIPTION
*id	uuid	Id úlohy

RESPONSE

STATUS CODE - 200: Success

STATUS CODE - 404: Not Found

7. TOKEN

7.1 GET /api/admin/token

Vrací zůstatek tokenů daného uživatele

REQUEST

QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
userId	string	Id uživatele

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  currentAmount number Aktuální částka na účtě
  lastAdded      string Datum posledního přidání tokenu
}
```

7.2 PUT /api/admin/token

Aktualizace zůstatku tokenů daného uživatele

REQUEST

REQUEST BODY - application/json

```
{
  userId string Id uživatele
  amount number Částka
}
```

RESPONSE

STATUS CODE - 200: Success

7.3 GET /api/token

Vrací zůstatek tokenů přihlášeného uživatele

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{  
  currentAmount number Aktuální částka na účtě  
  lastAdded      string Datum posledního přidání tokenu  
}
```

8. USER

8.1 GET /api/user

Získá všechny aktuální informace o přihlášeném uživateli

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  appMetadata      undefined
  email            string
  emailVerified    boolean
  phoneNumber      string
  phoneVerified    boolean
  userMetadata     undefined
  userName         string
  nickName        string
  firstName       string
  fullName        string
  lastName        string
  picture         string
  blocked         boolean
  createdAt       string
  identities [{
    Array of object:
      accessToken    string
      accessTokenSecret string
      connection    string
      expiresIn     integer
      isSocial      boolean
      profileData {
      }
      provider      string
      refreshToken  string
      userId       string
    }]
  lastIpAddress    string
  lastLogin       string
  lastPasswordReset string
  locale          string
  loginsCount     string
  updatedAt       string
  userId          string
  providerAttributes {
  }
  multifactor     [string]
  original {
    appMetadata    undefined
```

```

email                string
emailVerified        boolean
phoneNumber          string
phoneVerified        boolean
userMetadata         undefined
userName             string
nickName             string
firstName            string
fullName             string
lastName            string
picture              string
blocked              boolean
createdAt            string
identities [{
  Array of object:
    accessToken        string
    accessTokenSecret  string
    connection         string
    expiresIn          integer
    isSocial            boolean
    profileData {
    }
    provider           string
    refreshToken       string
    userId             string
  }]
lastIpAddress         string
lastLogin             string
lastPasswordReset    string
locale               string
loginsCount           string
updatedAt            string
userId               string
providerAttributes {
}
multifactor          [string]
}
vip                  boolean
registered           boolean
isAdmin              boolean
tokenBalance         string
}

```

8.2 DELETE /api/user

Smazání účtu a všech jeho vytvořených dat

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Success

8.3 PATCH /api/user

Upravení uživatelského profilu

REQUEST

REQUEST BODY - application/json-patch+json

```
[{  
  Array of object:  
    op    string  
    value {  
    }  
    path  string  
}]
```

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{  
  appMetadata      undefined  
  email            string  
  emailVerified    boolean  
  phoneNumber      string  
  phoneVerified    boolean  
  userMetadata     undefined  
  userName         string  
  nickName         string  
  firstName        string  
  fullName         string  
  lastName         string  
  picture          string  
  blocked          boolean  
  createdAt        string  
  identities [{  
    Array of object:  
      accessToken      string  
      accessTokenSecret string  
      connection       string  
      expiresIn        integer  
      isSocial         boolean  
      profileData {  
      }  
      provider         string  
      refreshToken     string  
      userId           string  
    }]  
  lastIpAddress    string  
  lastLogin        string  
  lastPasswordReset string  
  locale           string  
  loginsCount      string  
  updatedAt        string  
  userId           string  
  providerAttributes {  
  }
```

```

multifactor          [string]
original {
  appMetadata        undefined
  email              string
  emailVerified       boolean
  phoneNumber         string
  phoneVerified       boolean
  userMetadata        undefined
  userName            string
  nickName            string
  firstName           string
  fullName            string
  lastName            string
  picture            string
  blocked             boolean
  createdAt           string
  identities [{
    Array of object:
    accessToken        string
    accessTokenSecret   string
    connection         string
    expiresIn           integer
    isSocial            boolean
    profileData {
    }
    provider            string
    refreshToken        string
    userId              string
  }]
  lastIpAddress       string
  lastLogin           string
  lastPasswordReset   string
  locale              string
  loginsCount         string
  updatedAt           string
  userId              string
  providerAttributes {
  }
  multifactor         [string]
}
vip                  boolean
registered            boolean
isAdmin              boolean
tokenBalance         string
}

```

8.4 GET /api/user/admin-contact

Zobrazí kontakt na jednoho z administrátorů

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  userId    string
  picture   string
  fullName  string
  email     string
}
```

9. WORKER

9.1 GET /api/worker

Přidělí se další úloha

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  id            string  Id úlohy
  userId        string  Id uživatele
  state          enum    ALLOWED:0, 1, 2, 3, 4, 5, 6, 7, 8, 9
  time          string  Čas vytvoření úlohy
  taskFilePath   string  Cesta k souboru definující úlohu
  resultFilePath string  Cesta k souboru obsahující výsledek
  isPriority      boolean Určuje prioritu
  name          string  Jméno úlohy
}
```

9.2 POST /api/worker

Zasílá se nová událost informující o stavu úlohy

REQUEST

REQUEST BODY - application/json

```
{
  id            string  Id úlohy
  userId        string  Id uživatele
  state          enum    ALLOWED:0, 1, 2, 3, 4, 5, 6, 7, 8, 9
  time          string  Čas vytvoření úlohy
  taskFilePath   string  Cesta k souboru definující úlohu
  resultFilePath string  Cesta k souboru obsahující výsledek
  isPriority      boolean Určuje prioritu
  name          string  Jméno úlohy
}
```

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```
{
  id    string  Id události
  status enum    ALLOWED:0, 1, 2, 3, 4, 5, 6, 7, 8, 9
  time  string  Čas vytvoření
  taskId string  Id úlohy
}
```

}
