API Documentation

AlmaShines
API
Integration
Guide

About integration & its benefits

Secured REST APIs

List of APIs

Contact Us

clients@almashines.com

www.almashines.io

V1.0

API Integration

Integrating existing systems of the Client with Almashines Portal is an important part of avoiding the hassles and effort of migration and maintenance. It also gives a greater control on the information, and increases flexibility to use it on multiple platforms. For this, AlmaShines uses secured **REST APIs** for a seamless integration.

What is REST API ?

REST API (Application Programming Interface) is a way of communicating with web application via simple http calls.

Authentication:

You are issued two tokens API_KEY and API_SECRET, which are used to authenticate your requests. Please send your API_KEY token in POST parameter “**apikey**” and API_SECRET token in POST parameter “**apisecret**” in each of your API calls.

How can i get my API_KEY and API_SECRET?

Login to the Admin Portal -> Goto Portal Settings -> API key Generation -> Generate a new Key

For example:

```
https://www.almashines.com/data/api/createUser?
POST Parameters in each request:
        apikey:  YOUR_API_KEY,
        apisecret: YOUR_API_SECRET
```

Objects Used in Various APIs

Address Object:

```
{
“address”:          Address Line String
“address_city_name”:City Name
“address_country_name”:Country Name
“address_pincode”:          Pincode
“address_type”: “current”  or  “office”  or “permanent”
}
```

Work object

```
{
“company”:        Company Name
“designation”: City Name
“from_year”:      Year when user joined this organization
“from_month”: Month when user joined this organization
“currently_working”:        True/False
“to_year”:          If not currently_working, Year when user left this organization
“to_month”:        If not currently_working, Month when user left this organization
“description”:      Experience and Achievements during this work
}
```

Education object

```
{
          “institute”:      Institute Name
          “degree”:         Degree/Program name
          “department”:  Course/Department name
          “from_year”:     Year when user joined this Institute/School/University
          “from_month”: Month when user joined this Institute/School/University
          “to_year”:        Year when user passed out from this Institute/School/University
          “to_month”:      Month when user passed out from this Institute/School/University
          }
```

List of APIs

1. CreateUser

Description: API used for creating a new user

URL: *https://www.almashines.com/data/api/createUser*

Additional POST Parameters:

userData[first_name]: First name of User
userData[last_name]: Last name of User
userData[email]: Email of User
userData[role]: role of user (eg:“alumni”/“student”/”ex-employee”/”volunteer”)
userData[yoj]: Year user had joined your organization
userData[yop]: Year user had left your organization
userData[primary_phone_number]: String, Phone number (format “91-7048547411”)
userData[secondary_emails]: Array (Other email addresses of user)
userData[secondary_phone_numbers]: Array (Other phone numbers of user)
userData[gender]: “male” or “female”
userData[date-of-birth]: Date of birth, format “YYYY-mm-dd”
userData[marital_status]: “single” or “married”
userData[current-city]: Name of city currently living in
userData[current-country]: Name of country currently living in
userData[hometown-city]: Name of city where user’s hometown is
userData[hometown-country]: Name of country where user’s hometown is
userData[profile_url_linkedin]: Linkedin profile URL of user
userData[addresses]: Array of Address objects
userData[work_experiences]: Array of WCork objects
userData[education]: Array of Education objects (For educational institute related communities, user’s education info of that institutes should not be sent in this education parameter)
userData[FIELD_NAME]: For all other fields and community related fields listed in "Admin Dashboard”->“Portal Settings”->“Fields Manager”, use Full name of the field here in place of FIELD_NAME. For example, If in the Fields Manager, there are fields with names “Class”, ”Campus”, “Program Name”, “Department”. Then these should be added using following Post Parameters: userData[Class], userData[Campus], userData[Program Name], userData[Department]
authenticated_entry: true/false, whether the user entry is authenticated or verified or not. (note: This parameter currently does not work and is under-development stage)
send_mail: true/false, If true, then the system sends an email to the user notifying him/her about his/her profile creation.

Response:

```
{
“success”: 1,
“uid”: Unique ID of user created (used in updateUser and loginUser APis),
“message”: “User has been successfully created.”
}
```

2. UpdateUser

Description: API used for updating an existing user
URL: *https://www.almashines.com/data/api/updateUser*
Additional POST Parameters:

uid: Unique ID of user created
userData[first_name]: First name of User
userData[last_name]: Last name of User
userData[role]: role of user (eg:“alumni”/“student”/”ex-employee”/”volunteer”)
userData[yoj]: Year user had joined your organization
userData[yop]: Year user had left your organization
userData[primary_phone_number]: String, Phone number (format “+91-7048547411”)
userData[secondary_emails]: Array (Other email addresses of user)
userData[secondary_phone_numbers]: Array (Other phone numbers of user)
userData[gender]: “male” or “female”
userData[date-of-birth]: Date of birth, format “YYYY-mm-dd”
userData[marital_status]: “single” or “married”
userData[current-city]: Name of city currently living in
userData[current-country]: Name of country currently living in
userData[hometown-city]: Name of city where user’s hometown is
userData[hometown-country]: Name of country where user’s hometown is
userData[profile_url_linkedin]: Linkedin profile URL of user
userData[addresses]: Array of Address objects
userData[work_experiences]: Array of Work objects
userData[education]: Array of Education objects (For educational institute related communities, user’s education info of that institutes should not be sent in this education parameter)

userData[FIELD_NAME]: For all other fields and community related fields listed in "Admin Dashboard”->“Portal Settings”->“Fields Manager”, use Full name of the field here in place of FIELD_NAME. For example, If in Fields Manager, there are fields with name “Class”, ”Campus”, “Program Name”, “Department”. Then these should be added using following Post Parameters: userData[Class], userData[Campus], userData[Program Name], userData[Department]

userData[authentication_status]: “authenticated” or “blocked” or “demised”

Response:

```
{
“success”: 1,
“uid”:  Unique ID of user created (used in updateUser and loginUser APIs),
“message”:  “User has been successfully created.”
}
```

3. LoginUser

Description: API used for logging a user on almashines powered portal

URL: *https://www.almashines.com/data/api/loginUser*

Additional POST Parameters:

data: Following json string encrypted with AES-256 algorithm with your API-SECRET as a salt
"{"uid": UID of user, "timestamp": Current Timestamp of request}"

Response:

User will be redirected to a secure url to enter into the portal

Alternate Approach:

We do recommend an alternative approach for this functionality. You can use our SSO integration APIs as per this documentation.

4. listUsers

Description: API used for retrieving the list of users of an almashines powered portal

URL: *https://www.almashines.com/data/api/listUsers*

Additional POST Parameters:

stream: (required) (Integer >=1) represents stream or page used in pagination
limit: (required) (Integer in range [1,200]) No. of entries to be fetched
email: (Optional) email address of user
enrollment_number: (Optional) Enrollment No. of user
dob: (Optional) (string in format of yyyy-mm-dd) Date of birth of the user
RecentUsers: (Optional) Set value of this parameter 1, if you want to fetch recent users

Response Sample:

```
{
"success": 1,
"data": [
          {
          "unique_profile_id": "1992233",
          "salutation": "Mr.",
          "first_name": "John",
          "last_name": "Doe",
          "name": "John Doe",
          "primary_email": "john@example.com”,
          "secondary_emails": [],
          "primary_phone_number": "",
          "secondary_phone_numbers": [],
          "gender": "male",
          "date-of-birth": "",
          "marital_status": "",
          "current-city": "Gandhinagar",
          "current-country": "India",
          "hometown-city": "",
          "hometown-country": "",
          "addresses": [],
          "profile_url_linkedin": “”,
          "work_experiences": [],
          "other_education": [],
          "role": “faculty”,
          "year-of-joining": 2017,
          "year-of-leaving": "",
          "enrollment_number": [],
          "aunthentication_status": "authenticated",
          "profile-created": “1501248968”, (UNIX TIMESTAMP)
          "last-logged-In": “1632226686”, (UNIX TIMESTAMP)
          ………….. (More Community Related Fields and Custom
              Fields as configured in Field Manager in Admin Dashboard will also be here,) For eg:
                    “Degree”:  “”,
                    “Program”: “”
          },
          ….  MORE USERS …
```

```
        ]
}
```

5. listRecentlyUpdatedUsers

Description: API used for retrieving the list of users who recently updated their profile data

URL: *https://www.almashines.com/data/api/listRecentlyUpdatedUsers*

Additional POST Parameters:

stream: (Integer >=1) represents stream or page used in pagination
limit: (Integer in range [1,200]) No. of entries to be fetched
updated_In_Last_X_Hours: *(optional)* (Integer in range [6, 720]) This parameter specifies the time frame for fetching profiles, specifically the number of hours within which they must have been updated. If no value is provided for this parameter, it defaults to 720 hours.

Response:

Same as the response of previous api **listUser**

6. verifyIcardQr

Description: API can find if the QR code of Digital ID Card is valid or not
URL: *https://www.almashines.com/data/api/verifyIcardQr*

Additional POST Parameters:

id: (required) (Integer) represents a unique id of the digital id card
name: (required) (String) represents a secret token of the digital id card
getFullDetails: (optional) (Boolean) if passed true, then API will return icard owner's complete profile details

How it works: Scan the QR code and you will find URL information in it (for e.g. www.yourdomain.com/icard/verify?id={int}&name={str}). Use the id and name extracted from it to call this API.

Response:

```
{
“success”:  1,
“data”:   {

          “verified”: true,

          “profile”: {

                   Same as the response of previous api listUser (details of one specific profile only)

}}}
```

Validate event registration with same API:

One additional parameter to the same API can find if that person has registered for a particular event or not.
Additional POST Parameter to last API:
getEventRegistrationDetails: (Integer) represents a unique id of the event
Response:

```
{
“success”:  1,
“data”:   {

          “verified”: true,

          “event_registration_details”: {

          “Is_registered”: true,

          “tickets_count”: 4

          }

        }

        }
```

7. createICard

Description: API generates an ICard for an user, if it hasn’t be generated yet

URL: *https://www.almashines.com/data/api/createICard*

Additional POST Parameters:

uid: (required) (Integer) represents a unique id of the digital id card

Response:

CASE-1: There is no icard in system for the user with given uid

```
{
“success”:  1,

“message”:  “Successfully created”
“data”:  {

              “icard_id”:  INTEGER

          }

          }
```

CASE-2: There is already an icard in system for the user with given uid

```
{
“success”:  0,

“message”:  “User has already an ICard”
“data”:  {

                 “icard_id”:  INTEGER

              }

          }
```

CASE-3: An unexpected error occurred

```
{
“success”:  0,

“error”:  ERROR_MESSAGE

          }
```

8. createJob

Description: API helps you create a job or internship in portal

URL: *https://www.almashines.com/data/api/createJob*

Additional POST Parameters:

creater_portal_id: (required, integer) "Unique portal id of an admin to display as job creator for this job

job_type: (required, string) "Full Time, Internship, Contractual / Freelance work, Volunteer"

locations: (required, array of strings) Locations of Job

company_name: (required, string) Company/Organisation of the job

designation_name: (required, string) Designation of the job

description: (required, string) Description of the job

company_desc: (required, string) Description of the Company

receive_on: (required, string) one of these values ("on_portal", "external_link")

external_link: (conditional, string) An external link to apply for the job

salary_type: (optional, integer) Salary frequency (0=>per year,1=>per month, 2=> per day, 3=> per hour)

salary_min: (optional, integer) Minimum salary

salary_max: (optional, integer) Maximum salary

salary_currency: (optional, string) 3 char currency code as per ISO 4217. Eg: "USD"

industries: (optional, array) ['Computer software', 'agriculture','finance'] (Must be present in our global list of industires)

is_fresher: (optional, integer) "0=>experienced, 1=>fresher"

experience_min: (optional, integer) Minimum Years of experience required for job

experience_max: (optional, integer) Maximum Years of experience allowed for job

location_type: (required, integer) "0=>On-site/Work from office, 1=>Remote/Work from Home, 2=>Hybrid"

skills: (optional, array) *['Web Developement', 'Mobile App Developement']*

Response:

```
{
"success": 1,

"job_id": INTEGER

}
```

9. updateJob

Description: API helps you update a job or internship in portal

URL: *https://www.almashines.com/data/api/updateJob*

Additional POST Parameters:

job_id: (required, integer) "Unique Job Id” generated in createJob API

creator_portal_id: (required, integer) "Unique portal id of an admin to display as job creator for this job”

job_type: (conditional, string) This parameter is not update, but it is required when you want to Update Internship related fields (duration, start_date)

locations: (optional, array) [“Ahmedabad”,”Surat”,”Rajkot”] Locations of Job

company_name: (optional, string) Company/Organisation of the job

designation_name: (optional, string) Designation of the job

description: (optional, string) Description of the job

company_desc: (optional, string) Description of the Company

receive_on: (optional, string) one of these values (“on_portal”, “external_link”)

external_link: (conditional, string) An external link to apply for the job

salary_type: (optional, integer) Salary frequency (0=>per year,1=>per month, 2=> per day, 3=> per hour)

salary_min: (optional, integer) Minimum salary

salary_max: (optional, integer) Maximum salary

salary_currency: (optional, string) 3 char currency code as per ISO 4217. Eg: “USD”

industries: (optional, array) [‘Computer software’, ‘agriculture’,’finance’] (Must be present in our global list of industires)

is_fresher: (optional, integer) “0=>experienced, 1=>fresher”

experience_min: (optional, integer) Minimum Years of experience required for job

experience_max: (optional, integer) Maximum Years of experience allowed for job

location_type: (required, integer) “0=>On-site/Work from office, 1=>Remote/Work from Home, 2=>Hybrid”

skills: (optional, array) *['Web Developement', 'Mobile App Developement']*

Response:

```
{
“success”: 1
}
```

10. deleteJob

Description: API helps you delete a job or internship in portal

URL: *https://www.almashines.com/data/api/deleteJob*

Additional POST Parameters:

job_id: (required, integer) "Unique Job Id" generated in createJob API
creater_portal_id: (required, integer) "Unique portal id of an admin to display as job creator for this job"

Response:

```
{"success": 1}
```

11. customObjectAction

Description: API helps you delete a job or internship in portal

URL: *https://www.almashines.com/data/api/customObjectAction*

Additional POST Parameters:

object_name: (required, integer) Unique Object Name
unique_profile_id: (required, integer) Unique portal id of the user, on whose profile, this action being taken
action: (required, string) (“create”, “update”, “delete”, “fetch”)
id: (int) (Only For action=”update”, “delete”) ID of Object, on which action is to be taken
data: (Associative KeyValue Array) (Only For action=”create”, “update”)

Example-1:

For action = “create”

```
POST /data/api/customObjectAction HTTP/1.1
Host: www.almashines.com
Content-Type: application/json
Content-Length: 343
{
  "apikey": API_KEY
  "apisecret": API_SECRET,
  "object_name": "awards",
  "unique_profile_id": 485,
  "action": "create",
  "data": {award: “Gold Medal”, “achieved_for”: “Best Performance”}
}
```

Response-1:

```
{ “success”: 1, “data”: {“id”: Object_ID} }
```

Example-2:

For action = “update”

```
POST /data/api/customObjectAction HTTP/1.1
Host: www.almashines.com
Content-Type: application/json
Content-Length: 343
{
  "apikey": API_KEY
  "apisecret": API_SECRET,
  "object_name": "volunteering_details",
  "unique_profile_id": 485,
  "action": "update",
  "id": 5610,
  "data": {department: “ABC”,  “activity”: “Test Activity”,  date:”2020-04-16”}
}
```

Response-2: { “success”: 1}

Example-3:

```
For action = “delete”
POST /data/api/customObjectAction HTTP/1.1
Host: www.almashines.com
Content-Type: application/json
Content-Length: 343
{
  "apikey":  API_KEY
  "apisecret": API_SECRET,
  "object_name": "volunteering_details",
  "unique_profile_id": 485,
  "action": "delete",
  "id": 5610
}


Response-3:   { “success”:  1}
```

Example-4:

```
For action = “fetch”
POST /data/api/customObjectAction HTTP/1.1
Host: www.almashines.com
Content-Type: application/json
Content-Length: 343
{
  "apikey":  API_KEY
  "apisecret": API_SECRET,
  "object_name": "awards",
  "unique_profile_id": 485,
  "action": "fetch"
}


Response-4:

{
“success”:  1,
"data":[
         {"id":"5610", award: “Gold Medal”, “achieved_for”: “Best Performance”},
         {"id":"5611", award: “Silver Medal”, “achieved_for”: “Best Performance”}
         ]
}
```

12. getFormDetails

Description: API helps you to retrieve information about a specific form in the community. The API is used to fetch form metadata, status, and the fields associated with the form.

URL: *https://www.almashines.com/data/api/getFormDetails*

Additional POST Parameters:

form_id : (`required`, integer) "The unique identifier of the form whose details are to be fetched.”

Response:

```
{
 "success": 1,
 "data": {
  "form_id": "18",
  "title": "Payment Form",
  "description": "<p>\r\n\tGive money</p>\r\n",
  "created_time": "1531852200", (timestamp in seconds)
  "status": "Active",
  "metadata": {
   "statuses": [
    {
     "key": "1",
     "text": "open"
    },
    {
     "key": "2",
     "text": "pending"
    },
    {
     "key": "3",
     "text": "close"
    }
   ]
  },
  "fields": [
   {
    "id": "96",
    "name": "amount",
    "title": "Amount",
    "description": "",
  "type": "Single-Line Text",
  "is_required": "1",
  "data": [],
  "order": 0
},
{
  "id": "99",
  "name": "email",
  "title": "Email",
  "description": "",
  "type": "Single-Line Text",
  "is_required": "1",
  "data": [],
  "order": 3
},
{
  "id": "97",
  "name": "for",
  "title": "Payment is for",
  "description": "",
  "type": "Single-Line Text",
  "is_required": "1",
  "data": [],
  "order": 1
},
{
  "id": "98",
  "name": "name",
  "title": "Name",
  "description": "",
  "type": "Single-Line Text",
  "is_required": "1",
  "data": [],
  "order": 2
},
{
  "id": "100",
  "name": "phone",
        "title": "Phone",
        "description": "",
        "type": "Phone Number",
        "is_required": "1",
        "data": [],
        "order": 4
      },
      {
        "id": "3093",
        "name": "Role_yv6qq",
        "title": "Role",
        "description": "",
        "type": "Checkbox",
        "is_required": "0",
        "data": [
          "Admin",
          "Faculty",
          "Student",
          "Alumni"
        ],
        "order": 5
      },
      {
        "id": "3094",
        "name": "Verification ID_c1nqx",
        "title": "Verification ID",
        "description": "College ID, Employment ID, Govt. ID",
        "type": "File Upload",
        "is_required": "0",
        "data": [],
        "order": 6
      }
    ]
  }
}
```

13. listFormResponses

Description: API helps you to fetch responses associated with a specific form in a community. It provides various filtering options, including status filters, date filters, and attribute-based filters
URL: *https://www.almashines.com/data/api/listFormResponses*

Additional POST Parameters:

form_id :(required, integer) "The unique identifier of the form whose responses are to be fetched."
stream: (optional, integer) "Stream of data, default is 1"
limit: (optional, integer) "Limit of responses to fetch, default is 100, max is 100"
responseIDs: (optional, array) "Array of response IDs" i.e. ['1234','5678']
statusFilter: (optional, array) "Array of status filters" i.e ['Open','Close']
attributes[paymentDone]: (optional, array) "Array of attributes for payment status filters"
attributes[paymentId]: (optional, array) "Array of attributes for payment ID filters" i.e. ['1234','5678']
attributes: (optional, array of key value pair objects) "Array of attributes for multiple field filters" i.e.
[{"phone_t23ps":"0123456789"},{"email_rlre3":"manohar.sharma1.693@gmail.com"}]
orderByStatusChanged: (optional, boolean) "Fetch data in order by status changed time in descending order"
dateFilters: *(optional, array)*

An array to filter data by dates, with two keys: creation and status_update. Each key contains a `value` array that specifies a date range using from (start timestamp) and to (end timestamp), both in seconds.

Example:
dateFilters[creation][value][from] = '1722950032'
dateFilters[creation][value][to] = '1723010370'
dateFilters[status_update][value][from] = '1722950032'
dateFilters[status_update][value][to] = '1723010370'

Response:

```
{
 "success": 1,
 "data": [
  {
   "response_id": "6324",
   "Time": "Aug 09, 2024 06:12 PM IST",
   "response": {
    "amount": {
     "title": "Amount",
     "value": "9999999"
    },
    "email": {
     "title": "Email",
     "value": "michael@example.com"
    },
      "for": {
        "title": "Payment is for",
        "value": "Donation"
      },
      "name": {
        "title": "Name",
        "value": "Mattie C. Harrell"
      },
      "phone": {
        "title": "Phone",
        "value": "1-978-556-5059"
      },
      "Role_yv6qq": {
        "title": "Role",
        "value": [
          "Alumni"
        ]
      }
    },
    "isuser": "0",
    "userid": "0",
    "time_unformatted": "1723207375", (timestamp in seconds)
    "status_changed_time": "1723207375", (timestamp in seconds)
    "si": "0",
    "status": "Unknown",
    "payment_done": false,
    "user": {
      "fullname": "Guest User"
    }
  },
  {
    "response_id": "6258",
    "Time": "Aug 07, 2024 01:27 PM IST",
    "response": {
      "amount": {
        "title": "Amount",
        "value": "200"
      },
    },
    "email": {
      "title": "Email",
      "value": "xyz.693@gmail.com"
    },
    "for": {
      "title": "Payment is for",
      "value": "xyz"
    },
    "name": {
      "title": "Name",
      "value": "John Doe"
    },
    "phone": {
      "title": "Phone",
      "value": "91-9898989898"
    }
  },
  "isuser": "1",
  "userid": "1",
  "time_unformatted": "1723017461", (timestamp in seconds)
  "status_changed_time": "1723017461", (timestamp in seconds)
  "si": "0",
  "status": "Unknown",
  "user": {
    "uid": "1",
    "first_name": "John",
    "last_name": "Doe",
    "fullname": "John Doe",
    "profile_pic": {
      "small": "assets/images/profiles/50x50/1.jpg?v=1720165550",
      "medium": "assets/images/profiles/170x170/1.jpg?v=1720165550",
      "large": "assets/images/profiles/480x480/1.jpg?v=1720165550"
    },
    "profile_url": "http://localhost/portal/dev/skitahd/profile/1",
    "role": "Admin",
    "start": "2010",
    "end": "2024",
    "institute": "2nd bulk",
        "degree": "MSc.",

        "department": "Bio-Technologya",

        "specialization": "Account",

        "signature": "Admin",

        "uui": "61",

        "enrollment_number": "qwertyuiop",

        "joined": "1423918504", (timestamp in seconds)

        "cid": "3",

        "identity_number": null,

        "email": "xyz.693@gmail.com",

        "phone_code": "91",

        "phone_number": "9898989898"

      },

      "payment_done": false

    }

  ]

}
```

14. createFormResponse

Description: API helps you to submit responses to a specific active form in the community. It supports both registered user and guest submissions.

URL: *https://www.almashines.com/data/api/createFormResponse*

Additional POST Parameters:

form_id :(required, integer) "The unique identifier of the form to which the response is being submitted."

formdata:(required, array) "Form response details in array representing the form fields name as key and their response values. If date and time fields exist in form then provide value as a timestamp in seconds." i.e {

```
  "form_id": 18,
  "uid": 1811167,
  "formdata": {
    "amount": 100023,
    "email": "example@gmail.com",
    "for": "Event",
    "name": "John Dsouza",
    "phone": "91-xxxxxxxx",
    "Role_yv6qq": [
      "Student",
      "Alumni"
    ],
    "Verification ID_c1nqx": [
      {"url": "https://example.com/demo.jpg"},
      {"url": "https://example.com/demo.jpg"}
    ]
  }
}
```

uid:(optional, integer) "The unique identifier of the registered user submitting the response. Not required if the guest user is submitting the response."

Response:

```
{
   "success": 1,
   "data": {
       "response_id": 6323
   }
}
```

15. updateFormResponse

Description: API helps you to update an existing form response with new data. The response is updated for active forms in the community.

URL: *https://www.almashines.com/data/api/updateFormResponse*

Additional POST Parameters:

response_id: (required, integer) "Unique identifier of the response to be updated.”

form_id:(required, integer) “Unique identifier of the form associated with the response.”

formdata:(required, array) "Form response details in array representing the form fields name as key and their response values."

```
i.e {
  "response_id": 6324,
  "form_id": 18,
  "formdata": {
    "amount": 9999999,
    “Verification ID_c1nqx": [
      {"url": "https://example.com/demo.jpg"}
    ]
  }
}
```

Response:

```
{
    "success": 1,
    "data": {
        "response_id": 6324
    }
}
```

16. deleteFormResponse

Description: API helps you to delete an existing form response identified by its unique response ID.

URL: *https://www.almashines.com/data/api/deleteFormResponse*

Additional POST Parameters:

response_id: (required, integer) "Unique identifier of the response to be deleted.”

Response:

```
{

   "success": 1,

   "data": true

}
```

17. listJobs

Description: API helps you to retrieve a list of jobs available.

URL: *https://www.almashines.com/data/api/listJobs*

Additional POST Parameters:

stream: (optional, integer) "Specifies the stream of response list.”

limit:(optional, integer) “Limits the number of jobs returned in the response.”

filters:(optional, array of objects) filter by cites, company or designation. i.e

```
{
        "keyword": "Manager", // Search by company name or title of the job
        "open_only" : true/false, //Filter to show only open jobs(i.e.,those with deadlines that have not yet passed)
        "is_fresher": 0/1,
        "job_type": ["Full Time", "Internship","Contractual / Freelance work","Volunteer"],
        "location_type": ["On-site / Work from office", "Remote / Work from home", "Hybrid"],
        "salary_type": ["per year", "per month", "per day", "per hour"],
        "salary_min": 100000,
        "salary_max": 200000,
        "salary_currency":”INR”,
        "experience_min": 0,
        "experience_max": 10,
        "industries":["Computer Software"],
        "skills":["Web Developement", "Mobile App Developement"],
        "locations":["Ahmedabad","New York"], //List of cities or "work from home"
}
```

fetch_type: (optional, integer) Allowed values 0 and 1, if 0 then jobs list fetched and if 1 then internships list fetched, by default both jobs and internships are fetched.

Response:

```
{
  "success": 1,
  "jobs": [
    {
      "id": "740",
      "designation": "Web Application UI Developer",
      "description": "Tech",
      "company": "A3 Logics India Ltd",
      "created": "1655295398", (timestamp in seconds)
      "Deadline": "1656527400", (timestamp in seconds)
      "view_count": "0",
      "company_desc": "description",
      "receive_type": "on_portal",
      "job_type": "Job",
"locations":[{"id":"680869","name":"Ass\u00e8mini\/Assemini","region":"Sardinia","country":"Italy","country_code":"it"},{"id":"986943","name":"Ahmedabad","region":"Gujarat","country":"India","country_code":"in"}],
      "count_applications": "0",
      "posted_by": {
        "name": "Mihir Soniiii",
        "uid": "704"
      }
    }
  ]
}
```

18. JobApplications

Description: API helps you to retrieve a list of job applications for a specific job.

URL: *https://www.almashines.com/data/api/JobApplications*

Additional POST Parameters:

job_id: (required, integer) "Unique Job Id" generated in createJob API

iterate: (optional, JSON Object) By default latest 20 applications will be returned. Use this object to Iterate the applications. Below are the details of this object.

```
{
    "application_id": INTEGER, ID of an application of this job
    "iterator": "before" or "after" (if iterator value is "before", then the 20 applications received before application_id will be
     returned, if iterator value is "after", then the 20 applications received after application_id will be returned)
}
```

Response:

```
{
  "success": 1,
  "data": [
    {
      "applicant_id": "25251",
      "uid": "1811166",
      "status": "Shortlisted", (Update: Rejected state is now Not interested)
      "resume": "http://localhost/portal/dev/data/api/resume_download?token=64bf75e590ee288236e0a0de2172b77655d3928
                 1bead574f35b8b64969d9aac895b6b133c5422f75636ad998be37e99b",
      "applied_time": "1731300731" (timestamp in seconds)
    }
  ]
}
```