# Contents

1 API Documentation ........................................... 3
   1.1 Introduction ............................................. 3
   1.2 Tokens .................................................. 3
   1.3 Requests ............................................... 3
   1.4 Responses .............................................. 4
   1.5 User Management ........................................ 4
   1.6 Student Sync ............................................ 8
   1.7 Majors/Minors ........................................... 9
   1.8 Contacts ............................................... 10
   1.9 Jobs .................................................... 12
   1.10 Attendees .............................................. 13
   1.11 Events .................................................. 14
   1.12 Career Interests ...................................... 14

2 CSV Upload .................................................... 17
   2.1 File Requirements ...................................... 17
   2.2 Staff .................................................... 17
   2.3 Students ............................................... 18
   2.4 Contacts ............................................... 21
   2.5 Appointments .......................................... 21
   2.6 Appointment Types ..................................... 22
   2.7 Events .................................................. 23
   2.8 Notes ................................................... 24
   2.9 Labels ................................................... 24
   2.10 Campuses .............................................. 25
   2.11 Majors ................................................. 25
   2.12 Minors ................................................. 25
   2.13 Buildings .............................................. 26
   2.14 Rooms .................................................. 26
   2.15 Attendees .............................................. 26
   2.16 FDS .................................................... 26

3 Frequently Asked Questions .................................. 29
   3.1 How safe, reliable and secure is your application? .... 29
   3.2 What SSO systems do you support out of the box? .... 30
   3.3 How do we get student data into the system? ........... 30
   3.4 Can we import data into the system? ..................... 30
   3.5 Do I need to dedicate any of my infrastructure for Handshake? 30
3.6 Does Handshake use SSL? .................................................. 30
3.7 Does Handshake use SFTP? ............................................. 31
3.8 Is a header row required in the CSV file? ......................... 31
3.9 Do you have a preferred format for Booleans? ................. 31
3.10 What happens if the email_address field is passed with a blank? ................. 31
3.11 Are column headers case sensitive? .............................. 31
3.12 Where can I find your terms of service and privacy policy? ....... 31

4 Data Transfer FAQ ......................................................... 33
4.1 Why do you use Amazon S3? ........................................ 33
4.2 What setup do I have to do on my end? ............................ 33
4.3 What if I can't install python to use the aws cli? ............... 33
4.4 What sort of encryption is used with the data? .................. 33
4.5 How should I format my [student/staff/alumni] data? ......... 33
4.6 How do I transfer the data to my secure bucket? ............... 34
4.7 Can I whitelist s3 through my firewall? ........................... 34

5 Single Sign On ............................................................. 35
5.1 Integration Guide ......................................................... 35
5.2 Supported Integrations .................................................. 36

6 Email Delivery ............................................................. 39
6.1 Email Delivery Guidelines ............................................. 39
6.2 Whitelisting ............................................................... 39

7 References ................................................................. 41
7.1 Please remember all values are case sensitive .................... 41
7.2 Encodings ............................................................... 41
7.3 Booleans ............................................................... 41
7.4 Dates ................................................................. 41
7.5 School Years .......................................................... 41
7.6 Gender ................................................................. 42
7.7 Work Authorizations .................................................. 42
7.8 Education Levels ....................................................... 42
7.9 Administrative Roles .................................................. 42
7.10 Salary Type Names .................................................... 43
7.11 Job Type Names ....................................................... 43
7.12 Employment Type Names .......................................... 43
7.13 Ethnicity ............................................................. 44
7.14 Industries ............................................................ 44
7.15 Job Functions ........................................................ 45
7.16 Time Zone Options ................................................... 46
7.17 Major Groups ........................................................ 48

8 Application Documentation ............................................ 53
8.1 Applications ............................................................ 53
8.2 Appointments .......................................................... 54
8.3 Attendees .............................................................. 56
8.4 Authentication ........................................................ 57
8.5 Bulk Actions ........................................................... 58
8.6 Career Fairs ............................................................ 59
8.7 Checkin Kiosk .......................................................... 61
8.8 Contacts .............................................................. 63
8.9 Consortia .............................................................. 65
8.10 Conversations ........................................................ 65
8.11 Employers ................................................. 66
8.12 Documents ................................................ 68
8.13 Events ...................................................... 68
8.14 Labels ....................................................... 69
8.15 Mass Emails ............................................... 70
8.16 Notes ........................................................ 71
8.17 Interview Schedules ..................................... 72
8.18 Invoicing .................................................. 76
8.19 Job Postings ............................................... 77
8.20 Mentorships ................................................. 78
8.21 Notifications ............................................... 79
8.22 Payments ................................................... 79
8.23 Posts ........................................................ 80
8.24 Qualifications .............................................. 80
8.25 Document and Profile Review ......................... 81
8.26 RSS Feeds .................................................. 82
8.27 Student Permissions ..................................... 82
8.28 User Links .................................................. 83
8.29 Users and Students ...................................... 83
8.30 Users vs. Contacts ....................................... 84
Contents:
1.1 Introduction

The Handshake API allows users to make RESTful calls using JSON for sending and receiving data.

The API root URL is: https://app.joinhandshake.com/api/v1/

1.2 Tokens

API Calls must use an API token to authenticate with the service. API Tokens can be managed in the school administration page. To create an API token, please contact your Handshake account manager.

Please safeguard the API token like you would for password. Never include or check-in the API token into any publicly hosted repositories (such as GitHub), nor post it to any publicly accessible pages.

If you believe that an API token has been compromised, immediately delete it from the administration page and it will no longer be active.

When making API calls, the API token needs to be passed via an HTTP header. Set the authorization header to have your API token. An example curl request would be:

curl https://app.joinhandshake.com/api/v1/users/create -H 'Authorization: Token token="token_goes_here"'

1.3 Requests

POST, PUT and DELETE requests take JSON as the request payload. A normal request would look something like the below hash:

```json
{
    user: {
        attr1: "value",
        attr2: "value2"
    }
}
```
1.4 Responses

The Handshake API has a consistent response format. API responses are always returned as JSON hash. Each response has a ‘success’ attribute on that hash which will be set to either true or false. This is an easy way to know if the call succeeded.

In addition, there is at least one other attribute which contains the bulk of the data. This will either be a single object (ex. after creating a user), a list (ex. a search) or errors. An example response for each is below.

**Single Object**

```json
{
    success: true,
    user: {
        attr: "value"
    }
}
```

**List**

The individual objects within the ‘objects’ list will typically contain the same data as a single object response would:

```json
{
    success: true,
    users: [
        { ... }, { ... }
    ]
}
```

**Error**

The ‘errors’ attribute is a hash with attributes as keys and values as arrays of errors on that attribute:

```json
{
    success: false,
    errors: {
        "attribute": ["error 1 on attribute", "error 2 on attribute"],
        "attribute2": ["error 1 on attribute2"]
    }
}
```

1.5 User Management

Handshake allows users to manage users at their school via the API. This can be useful when integrating with other systems at the university which hold student data in order to keep Handshake up to date.

This can also be used to add system labels to students. System labels are labels that can only be applied during a sync, and can’t be edited or removed on the UI. System labels are like private labels, and can only be seen by other staff members at the institution. System labels should be passed in as an array of label names. If a label exists with the same name, it will be converted to a system label. If the system labels key is passed in but no label names are passed, it will remove all labels from the student. Similarly, if a student has a system label applied, but it is not present in the current sync, it will be removed from the student. If the system labels key is not passed, no existing labels will be removed from the student.
1.5.1 [GET] /users

Allows administrators to search for students at their school.

**Params**

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>query</td>
<td>A simple string query to search with</td>
</tr>
</tbody>
</table>

**Sample Response**

```json
{
  success: true,
  users: [
    {
      email_address: "sgringwe@mtu.edu",
      username: "sgringwe",
      first_name: "Scott",
      ...
    },
    {...}, {...}
  ]
}
```

1.5.2 [POST] /users

Allows administrators to add a student.

**Params**
### Key | Value
--- | ---
*email_address | (String) Student’s email address
*username | (String) Student’s username
*user_type | (String) Defaults to “Students”, one of “Students”, “Career Services”, “Mentors”
auth_identifier | (String) This is the identifier that is required if you use Single Sign On.
recomended_authentication | (String) One of “sso” or “standard”. Allows you to suggest what type of authentication the user should use when logging in.
card_id | (String) A card id that can be used for card swipe checkins.
first_name | (String) Student’s first name
last_name | (String) Student’s last name
school_year_name | (String) The name of student’s school year. See references for possible values.
prefereed_name | (String) The student’s preferred name
middle_name | (String) The student’s middle name
etnicity | (String) The ethnicity of the user. See the reference section for options.
gender | (String) The gender of the user. One of “Male”, “Female”, “Other”, or blank (Not specified)
bio | (String) A student bio
skill_names | (String Array) An array of skills to list on the students profile
external_link_urls | (String Array) An array of external links to list on the students profile
time_zone | (String) The time zone that this user is in. See time zones section for more details.
disabled | (Boolean) Pass true if this student should not be able to login
override_disabled_field | (Boolean) This field tells Handshake to ignore this user in future syncs and is used to transition a student to an alumni.
work_study_eligible | (Boolean) Pass true if this student is eligible for work study jobs
mentor_information_attributes | (Hash) A nested hash containing mentor-specific attributes. See below table for possible attributes.
campus_name | The name of the campus the student is at. Must be one of the campuses set up in your settings.
mobile_number | The user’s mobile phone number
system_label_names | (String Array) An array of label names to apply to the user
profile_review_status | This can be used to manage a students review status. Set this to “approved” if this student will not need a profile review. (Not relevant if profile review is not turned on for your school). All options: [‘unsubmitted’, ‘pending’, ‘approved’].
document_review_status | This can be used to manage a students document review status. Set this to “automatically_approved” if this student will not need documents approved. (Not relevant if document review is not turned on for your school). All options: [‘no_pending_documents’ ‘pending_documents’ ‘automatically_approved’]
primary_education_attributes | (Hash) A nested hash containing the primary education attributes. See below table for possible attribute values.

### Training configuration fields
These are used only for trainings and demo setups.

| Key | Value |
--- | ---
role_names | A semi colon separated list of roles to give to this user.
password | Only used in demo environment for setting up trainings. Must match confirmation.
password_confirmation | Only used in demo environment for setting up trainings. Must match confirmation.

### Primary education params
These are nested inside of ‘primary_education_attributes’ above. These values will be assigned to the student’s primary education, which is the education determined as the record to respect when determining job and interview schedule qualifications.
Key | Value
--- | ---
education_level_name | (String) Undergraduate, Graduate, Postgraduate. This shows up on their main education on their profile.
cumulative_gpa | (Decimal) The student’s cumulative GPA
department_gpa | (Decimal) Decimal of student’s departmental GPA
major_names | (String Array) An array of major names for this student. These must be majors configured in the school’s majors list.
minor_names | (String Array) An array of minor names for this student. These must be minors configured in the school’s minors list.
college_name | (String) The college the student belongs to. Must be one of the colleges configured in the school’s college list.
start_date | (Date) The date the student started at the school in any standard date format. See references for date formats.
end_date | (Date) The date the student finished at the school (can be blank if currently_attending is set). See references for date formats.
currently_attending | (Boolean) Should be set to true if education_end_date is blank. This signifies they are currently attending this school.

**Mentor information params** These are nested inside of ‘mentor_information_attributes’ above

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>student_contact_preference</td>
<td>(String) Whether or not this mentor can be contacted by students. Either ‘allowed’ or ‘not_allowed’</td>
</tr>
<tr>
<td>advice</td>
<td>(String) Generic advice that this mentor has to offer</td>
</tr>
<tr>
<td>hobbies</td>
<td>(String) Relevant hobbies that this mentor listed</td>
</tr>
<tr>
<td>expertise_names</td>
<td>(String Array) An array of expertise that this mentor has. Will create if not already listed on school administrator page.</td>
</tr>
<tr>
<td>maximum_mentees</td>
<td>(Integer) The maximum number of ongoing mentorships that this mentor is willing to do. Defaults to 50.</td>
</tr>
<tr>
<td>maximum_student_contacts_per_month</td>
<td>(Integer) The maximum number of messages that this mentor is willing to receive.</td>
</tr>
<tr>
<td>industry_name</td>
<td>(String) The industry that this mentor is in. See references for possible values</td>
</tr>
</tbody>
</table>

* required

**Sample Response**

```
{
  success: true,
  user: {
    email_address: "sgringwe@mtu.edu",
    username: "sgringwe",
    first_name: "Scott",
    ...}
}
```

### 1.5.3 [PUT] /users/update

Allows administrators to update a student’s details.

Updating sensitive fields (username, email_address, and auth_identifier) require setting top-level param ‘change_sensitive_fields’ to true. The request would fail otherwise.

**Top Level Params**
Key | Value
---|---
change_sensitive_fields | (Boolean) Pass true to force update sensitive user fields.

See [POST] /users for user params.

Sample Response
```
{
    success: true,
    user: {
        email_address: "sgringwe@mtu.edu",
        username: "sgringwe",
        first_name: "Scott",
        ...
    }
}
```

### 1.5.4 [DELETE] /users/destroy

Allows administrators to remove a student from Handshake.

**Params**

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>*email_address</td>
<td>Student’s email address</td>
</tr>
<tr>
<td>*username</td>
<td>Student’s username</td>
</tr>
</tbody>
</table>

*One of email_address or username must be passed in order to find the user to remove*

Sample Response
```
{
    success: true,
    user: {
        email_address: "sgringwe@mtu.edu",
        username: "sgringwe",
        first_name: "Scott",
        ...
    }
}
```

### 1.6 Student Sync

#### 1.6.1 [POST] /users/start_sync

Tells the Handshake API that you are beginning a student data sync and moves the school in to “sync status”.

Sample Response
```
{
    success: true
}
```
1.6.2 [POST] /users/create_or_update

Takes in normal user params (see [POST] /users). If user does not yet exists, creates them. If user already exists, updates with given fields.

Updating sensitive fields require setting top-level param ‘change_sensitive_fields’ to true. See [PUT] /users/update.

Sample Response

See [POST] /users.

1.6.3 [POST] /users/sync_details

Gives details about the current status of the sync including how many have been updated, how many have been created and how many users are not yet accounted for.

Sample Response The following is an example of a response near the beginning of the sync process.

```json
{
  success: true,
  unaccounted_count: 11283,
  updated_count: 4239,
  created_count: 4
}
```

1.6.4 [POST] /users/end_sync

Finishes the sync process. Disables any students who were not accounted for during the sync and moves the school out of “sync status”.

Sample Response

```json
{
  success: true
}
```

1.7 Majors/Minors

The following is the same for minors. This part of the API allows career services centers to add, remove and receive a list of majors in the system for their school.

1.7.1 [GET] /majors

Allows administrators to list majors for their school by name

Params

None

Sample Response

```json
{
  success: true,
  majors: ['Major name', 'Major 2 name']
}
```
1.7.2 [POST] /majors

Allows administrators to add a major to their school. Returns false if major is already at the school.

**Params**

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of major</td>
</tr>
<tr>
<td>major_group_names</td>
<td>Array of major group names to allocate this major into</td>
</tr>
</tbody>
</table>

**Sample Response**

```json
{
  success: true,
  major: 'Major name that was added'
}
```

1.7.3 [DELETE] /majors/destroy

Allows administrators to remove a major from their school. Returns false if major is not at the school.

**Params**

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of major</td>
</tr>
</tbody>
</table>

**Sample Response**

```json
{
  success: true,
  major: 'Major name that was removed'
}
```

1.8 Contacts

Allows managing contacts at your institution.

1.8.1 [GET] /contacts

Allows administrators to list contacts.

**Params**

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>*first_name</td>
<td></td>
</tr>
<tr>
<td>*last_name</td>
<td></td>
</tr>
<tr>
<td>*email_address</td>
<td></td>
</tr>
<tr>
<td>**employer_id</td>
<td>The id of the employer that you want to list the contact for</td>
</tr>
<tr>
<td>**employer_name</td>
<td>The name of the employer that the contact represents</td>
</tr>
<tr>
<td>title</td>
<td>The job title of this contact, for example 'University Relations'</td>
</tr>
<tr>
<td>location_name</td>
<td></td>
</tr>
<tr>
<td>phone</td>
<td></td>
</tr>
<tr>
<td>cell_phone</td>
<td></td>
</tr>
<tr>
<td>fax</td>
<td></td>
</tr>
<tr>
<td>description</td>
<td></td>
</tr>
<tr>
<td>assigned_to_id</td>
<td>The id of the user in Handshake that manages this contact</td>
</tr>
</tbody>
</table>
* Required ** Either employer_id or employer_name may be provided, but employer_id is more accurate

Sample Response

```json
{
  success: true,
  contacts: [
    {
      first_name: 'Bill',
      last_name: 'Hertz',
      email_address: 'careers@acmecorp.com',
      ...
    },
    { ... },
  ]
}
```

1.8.2 [POST] /contacts

Add a contact to an employer

**Params**

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>*email_address</td>
<td>The email address of the contact.</td>
</tr>
<tr>
<td>employer_id</td>
<td>The id of the employer to add the contact to.</td>
</tr>
<tr>
<td>employer_name</td>
<td>The name of the employer to add the contact to.</td>
</tr>
<tr>
<td>first_name</td>
<td>The first name of the contact.</td>
</tr>
<tr>
<td>last_name</td>
<td>The last name of the contact.</td>
</tr>
<tr>
<td>title</td>
<td>The title of the contact.</td>
</tr>
<tr>
<td>address</td>
<td>The address of the contact.</td>
</tr>
<tr>
<td>location_id</td>
<td>The id of the work location of the contact.</td>
</tr>
<tr>
<td>phone</td>
<td>The phone number of the contact</td>
</tr>
<tr>
<td>cell_phone</td>
<td>The cell phone number of the contact</td>
</tr>
<tr>
<td>fax</td>
<td>The fax number of the contact</td>
</tr>
</tbody>
</table>

* Required

If employer_id or employer_name is specified, the contact will be associated with the corresponding employer. Associating contacts using employer_name will only work if the name is an exact match with the employer profile’s current name. If the name is not an exact match or the employer has no Handshake profile, the contact will display as belonging to that employer by name only; no employer profile will be linked to the contact. That contact may later become associated with an employer via the contact invite process, if the contact creates a user account and joins or creates an employer profile.

If the contact being created already has a user account on Handshake (looked up via the provided email address), and that user account is associated with an employer, then the contact will be associated with the same employer as the user. This will override anything that is passed for employer_id and/or employer_name.

Sample Response

```json
{
  success: true,
  contact: {
    employer_id: 1,
    email_address: 'bill@acmecorp.com',
    ...
  }
}
```
### 1.8.3 [DELETE] /contacts/destroy

Allows administrators to remove a contact from an employer. Returns false if contact is not at the school.

**Params**

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>employer_id</td>
<td>The id of the employer to add the contact to.</td>
</tr>
<tr>
<td>email_address</td>
<td>The email address of the contact.</td>
</tr>
</tbody>
</table>

**Sample Response**

```
{
  success: true,
  employer: {
    name: 'Acme Corp.',
    email_domain: 'careers@acmecorp.com'
  }
}
```

### 1.9 Jobs

Allows managing jobs at your school

#### 1.9.1 [GET] /jobs

Allows administrators to list jobs at your school

**Params**

None

**Sample Response**

```
{
  success: true,
  jobs: [
    {
      title: 'Engineering Intern'
    }
  ]
}
```

#### 1.9.2 [POST] /jobs

Allows administrators to create jobs at your school

**Params**
**Key** | **Value**  
--- | ---  
*title* | The job's title  
*employer_id* | System ID of the employer associated with this job  
*job_type_name* | The type of job. Must be one of the system job types  
*application_medium* | The method a student should use to apply. One of ['handsake', 'external_link', 'offline', 'handshake_and_external']  
*physical_application_instructions* | Instructions on how a student should submit a physical application. This is required if the application medium is 'offline'  
description | Description of the job  
job_function_names | An array of job function names which must be one of the system job functions.  
locations_attributes | An array of hashes containing a “name” field. Each “name” field should represent a location for the job.  
salary_type | The salary type. Must be one of the system salary types  
contact_email | The email of the contact to be associated with the job. Must match with an existing contact  
expiration_date | The date the posting should expire. yyyy-mm-dd  
posting_status | The status of the posting, if being posted to a school. Possible values: expired, approved, pending, declined.

* Required fields

**Sample Response**

```json
{
  success: true,
  job: {
    title: 'Engineering Intern'
  }
}
```

**1.10 Attendees**

Allows for creation and indexing of attendee records.

**1.10.1 [GET] /attendees**

Allows pulling for attendees of an attendable event. The results can be paginated and are ordered by most recently updated first.

**Params**

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>identifier</strong></td>
<td>The import identifier of the attendable.</td>
</tr>
<tr>
<td><strong>identifiable_id</strong></td>
<td>The id of the attendable.</td>
</tr>
<tr>
<td><em>identifiable_type</em></td>
<td>The type of the event, either ‘CareerFair’ or ‘Event’.</td>
</tr>
<tr>
<td>page</td>
<td>The page of results that you want, 0-based.</td>
</tr>
</tbody>
</table>

* Required fields ** Either identifier or identifiable_id must be provided.

**Sample Response**

```json
{
  success: true,
  attendees: [
    |
  ]
}
```
### 1.11 Events

Allows managing events at your institution.

**1.11.1 [POST] /events**

Add an event

**Params**

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>*start_date</td>
<td>(String) The date and time when the event starts. Ex: ‘2017-03-10 9:00 AM’</td>
</tr>
<tr>
<td>*end_date</td>
<td>(String) The date and time when the event ends. Ex: ‘2017-03-10 10:00 AM’</td>
</tr>
<tr>
<td>*name</td>
<td>(String) The name of the event.</td>
</tr>
<tr>
<td>*event_type_name</td>
<td>(String) The type of event. Possible types: ‘Workshop’, ‘Info Session’, ‘Group Appointment’, ‘Other’</td>
</tr>
<tr>
<td>description</td>
<td>(String) The description of the event.</td>
</tr>
<tr>
<td>student_registration_start</td>
<td>(String) The date students can start registering. Ex: ‘2017-03-01 7:00 AM’</td>
</tr>
<tr>
<td>student_registration_end</td>
<td>(String) The date students can no longer register. Ex: ‘2017-03-09 8:00 PM’</td>
</tr>
<tr>
<td>invite_only</td>
<td>(Boolean) If the event requires an invite.</td>
</tr>
<tr>
<td>attendee_limit</td>
<td>(Integer) The number of attendees allowed to register.</td>
</tr>
<tr>
<td>external_link</td>
<td>(String) If provided, students will be redirected to this link when they register.</td>
</tr>
<tr>
<td>welcome_student_email</td>
<td>(String) This content will be emailed to students when they join this event.</td>
</tr>
</tbody>
</table>

* Required fields

**Sample Success Response**

```json
{
  success: true,
  event: {
    "id": 1,
    "name": "New Event"
  }
}
```

### 1.12 Career Interests

Allows getting information about career clusters your school has configured, and which students have indicated their interest in those clusters.
1.12.1 [GET] /career_interests

Get the list of career clusters that are configured for your school.

**Params** None

**Sample Response**

```json
{
  success: true,
  career_clusters: [
    {
      id: 1,
      name: "Software Development"
    },
    {
      id: 2,
      name: "Business"
    }
  ]
}
```

1.12.2 [GET] /career_interests/{id}

Get a list of students who are interested in a given career cluster. The ID of a career cluster can be determined using the above endpoint.

**Params** None

**Sample Response**

```json
{
  success: true,
  students: [
    {
      id: 123,
      email_address: "jane@handshake.edu",
      first_name: "Jane",
      last_name: "Doe"
    },
    {
      id: 456,
      email_address: "john@handshake.edu",
      first_name: "John",
      last_name: "Doe"
    }
  ]
}
```
CHAPTER 2

CSV Upload

The easiest way to get data into Handshake is using the CSV upload. Below you will find tips, requirements, and an explanation of the different files.

2.1 File Requirements

- All headers are case-sensitive; **Must be downcased**
- All preset values are case-sensitive; Casing varies
- File **must be** exported in a true .CSV format
- File **may not** be another file type with a .CSV extension added to it
- File **may not** include CRLF terminators, any line breaks will result in immediate failure
- File **may not** have trailing commas (or additional columns), this will result in immediate failure
- File must be exported in Unicode (UTF-8)

Note: Not exporting to UTF-8 can cause bad encoding issues, which can lead to poorly translated records or failure to sync/import correctly. For information on how to export or convert your file to UTF-8, please see: [http://help.meetedgar.com/article/107-help-how-do-i-export-my-spreadsheet-to-a-utf-8-encoded-csv](http://help.meetedgar.com/article/107-help-how-do-i-export-my-spreadsheet-to-a-utf-8-encoded-csv)

2.2 Staff

2.2.1 Required Fields

<table>
<thead>
<tr>
<th>CSV Header Value</th>
<th>Value Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>email_address</td>
<td>Staff member’s email address. In general this should be a .edu address</td>
</tr>
<tr>
<td>username</td>
<td>Staff member’s username. This MUST be unique and should not be something that changes</td>
</tr>
<tr>
<td>user_type</td>
<td>Should be “Career Services”</td>
</tr>
<tr>
<td>auth_identifier</td>
<td>This is the identifier that is required if you use Single Sign On. <strong>Must be unique. This cannot be applied to both a staff and student’s account</strong></td>
</tr>
</tbody>
</table>
2.2.2 Recommended Fields

<table>
<thead>
<tr>
<th>Header</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>first_name</td>
<td>The staff member’s first name <strong>Required for new records</strong></td>
</tr>
<tr>
<td>last_name</td>
<td>The staff member’s last name</td>
</tr>
<tr>
<td>preferred_name</td>
<td>The staff member’s preferred name <strong>Do not include the last name as this will cause duplication on the user’s profile</strong></td>
</tr>
<tr>
<td>middle_name</td>
<td>The staff member’s middle name</td>
</tr>
<tr>
<td>gender</td>
<td>The gender of the user. One of “Male”, “Female”, “Other”, or blank (Not specified)</td>
</tr>
<tr>
<td>bio</td>
<td>The staff member’s bio</td>
</tr>
<tr>
<td>mobile_number</td>
<td>The staff member’s bio</td>
</tr>
<tr>
<td>role_names</td>
<td>Applications;Applications and Outcomes;Articles;Career Fair;Career Fairs;Career Plans;Employer Approvals;Events;Experiences;External Feeds;Interview Schedules;Jobs;Launch Check-in Kiosk;Manage All Appointments;Manage Labels;Manage Own Appointments;Manage Staff;Manage Students;Mass Emails;Mentorships;Outcomes;Pins;Posts;Reports;Request Access to Schools;Rooms;Student Reviews;Surveys;Upload Attachments;View as Student;View Shared Notes</td>
</tr>
</tbody>
</table>

For an example file of the required fields [Click Here]

2.3 Students

For an example file of the suggested fields [Click Here]

Format: TIMESTAMP_users.csv OR users.csv

Where TIMESTAMP is a single integer representing the time with smaller units further to the right. Ex: 9:00:15 on 7/14/2014 => 201471490015_users.csv

Timestamps are optional, but can be used to make sure only the latest users list will be synced if the network connection is unreliable and causes two files originally uploaded at different times to show up on our system at the same time.

2.3.1 Required Fields

<table>
<thead>
<tr>
<th>CSV Header</th>
<th>Value Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>email_address</td>
<td>Student’s email address. In general this should be a .edu address</td>
</tr>
<tr>
<td>username</td>
<td>Student’s username. This MUST be unique and should not be something that changes</td>
</tr>
<tr>
<td>user_type</td>
<td>Defaults to “Students”, one of “Students”, “Career Services”, “Mentors”. user_type is case sensitive so it must be “Students”.</td>
</tr>
<tr>
<td>auth_identifier</td>
<td>This is the identifier that is required if you use Single Sign On.</td>
</tr>
</tbody>
</table>
## 2.3.2 Recommended Fields

<table>
<thead>
<tr>
<th>Header</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>first_name</td>
<td>Student’s first name <strong>Required for new records</strong></td>
</tr>
<tr>
<td>last_name</td>
<td>Student’s last name</td>
</tr>
<tr>
<td>school_year_name</td>
<td>The name of student’s school year. For a list of acceptable values see the references section. Can only have one.</td>
</tr>
<tr>
<td>primary_education:education_level_name</td>
<td>This shows up on their main education on their profile. For a list of acceptable values see the references section. Can only have one.</td>
</tr>
<tr>
<td>primary_education:cumulative_gpa</td>
<td>(Decimal) The student’s cumulative GPA</td>
</tr>
<tr>
<td>primary_education:department_gpa</td>
<td>(Decimal) Decimal of student’s departmental GPA</td>
</tr>
<tr>
<td>primary_education:majors</td>
<td>(String Array) An array of major names for this student. Semicolon separated list. <strong>Quotation is only necessary in the event of special characters being used</strong></td>
</tr>
<tr>
<td>primary_education:minor</td>
<td>(String Array) An array of minor names for this student. Semicolon separated list. <strong>Quotation is only necessary in the event of special characters being used</strong></td>
</tr>
<tr>
<td>primary_education:college</td>
<td>(String) The college the student belongs to. Must be one of the colleges configured in the school’s college list. <strong>Can only have one college</strong></td>
</tr>
<tr>
<td>primary_education:start_date</td>
<td>(Date) The date the student started at the school in any standard date format. See references for date formats.</td>
</tr>
<tr>
<td>primary_education:end_date</td>
<td>(Date) The date the student graduated or plans to graduate school (can be blank if currently_attending is set). Must be after the education start date. See references for date formats.</td>
</tr>
<tr>
<td>primary_education:currently_attending</td>
<td>(Boolean) Should be set to true if education_end_date is blank. This signifies they are currently attending this school.</td>
</tr>
<tr>
<td>card_id</td>
<td>(String) Used for checking in students using a card swipe. This string must be contained in a card swipe output. Handshake can regex the direct output to match this value.</td>
</tr>
<tr>
<td>ethnicity</td>
<td>The ethnicity of the user. Can only have one. See the reference section for options.</td>
</tr>
<tr>
<td>gender</td>
<td>The gender of the user. One of “Male”, “Female”, “Other”, or blank (Not specified)</td>
</tr>
<tr>
<td>opt_cpt_eligible</td>
<td>(Boolean) Should be set to true if the student is opt/cpt eligible.</td>
</tr>
</tbody>
</table>

## 2.3.3 Optional Fields

Note: these fields generally add value to career services but are not required for normal operation.
### CSV Header Value

<table>
<thead>
<tr>
<th>Key</th>
<th>Value Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>preferred_name</td>
<td>The student's preferred name</td>
</tr>
<tr>
<td>middle_name</td>
<td>The student’s middle name</td>
</tr>
<tr>
<td>recommended_action</td>
<td>One of “sso” or “standard”. Allows you to suggest what type of authentication the user should use when logging in. (not currently active)</td>
</tr>
<tr>
<td>bio</td>
<td>A student’s bio. Shown on the student profile. Visible to everyone who can see the profile.</td>
</tr>
<tr>
<td>skill_names</td>
<td>Semi-colon separated list of skills. This generally should not be used in a sync.</td>
</tr>
<tr>
<td>external_link_urls</td>
<td>Semi-colon separated list of external links for the profile</td>
</tr>
<tr>
<td>disabled</td>
<td>Pass true if this student should not be able to login and access Handshake.</td>
</tr>
<tr>
<td>campus_name</td>
<td>The name of the campus the student is at. Must be one of the campuses set up in your settings. <strong>May only have one campus</strong></td>
</tr>
<tr>
<td>mobile_number</td>
<td>The user’s mobile phone number. The format should follow the following format:</td>
</tr>
<tr>
<td></td>
<td>(999)999-9999 Ext:9999</td>
</tr>
<tr>
<td>system_label_names</td>
<td>Semi-colon separated list of label names to apply to the user</td>
</tr>
<tr>
<td>assigned_to_email_address</td>
<td>Email Address of staff member the student is assigned to</td>
</tr>
<tr>
<td>profile_review_status</td>
<td>This can be used to manage a students review status. Set this to “approved” if this student will not need a profile review. (Not relevant if profile review is not turned on for your school). All options: [’unsubmitted’, ‘pending’, ‘approved’].</td>
</tr>
<tr>
<td>document_review_status</td>
<td>This can be used to manage a students document review status. Set this to “automatically_approved” if this student will not need documents approved. (Not relevant if document review is not turned on for your school). All options: [’no_pending_documents’, ‘pending_documents’, ‘automatically_approved’].</td>
</tr>
</tbody>
</table>

### 2.3.4 Training configuration fields

These are used only for trainings and demo setups.

<table>
<thead>
<tr>
<th>Key</th>
<th>Value Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>role_names</td>
<td>A semi colon separated list of roles to give to this user.</td>
</tr>
<tr>
<td>password</td>
<td>Only used in demo environment for setting up trainings. Must match confirmation.</td>
</tr>
<tr>
<td>password_confirmation</td>
<td>Only used in demo environment for setting up trainings. Must match confirmation.</td>
</tr>
</tbody>
</table>

### 2.3.5 Mentor information params

These are nested inside of ‘mentor_information_attributes’ above

<table>
<thead>
<tr>
<th>Key</th>
<th>Value Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>student_contact_preference</td>
<td>(String) Whether or not this mentor can be contacted by students. Either ‘allowed’ or ‘not_allowed’</td>
</tr>
<tr>
<td>advice</td>
<td>(String) Generic advice that this mentor has to offer</td>
</tr>
<tr>
<td>hobbies</td>
<td>(String) Relevant hobbies that this mentor listed</td>
</tr>
<tr>
<td>expertise_names</td>
<td>(String Array) An array of expertise that this mentor has. Will create if not already listed on school administrator page.</td>
</tr>
<tr>
<td>maximum_mentees</td>
<td>(Integer) The maximum number of ongoing mentorships that this mentor is willing to do.</td>
</tr>
<tr>
<td>maximum_student_contacts_per_month</td>
<td>(Integer) The maximum number of messages that this mentor is willing to receive.</td>
</tr>
<tr>
<td>industry_name</td>
<td>(String) The industry that this mentor is in. See references for possible values</td>
</tr>
</tbody>
</table>
2.3.6 Handling Students who Graduate

- Run a final sync before graduation that updates their school year status to Alumni

2.3.7 System Labels

System labels are labels that can only be applied during a sync, and can’t be edited or removed on the UI. System labels are like private labels, and can only be seen by other staff members at the institution. System labels should be passed in as a semi-colon separated list of label names. If a label exists with the same name, it will be converted to a system label. If the system labels key is passed in but no label names are passed, it will remove all labels from the student. Similarly, if a student has a system label applied, but it is not present in the current sync, it will be removed from the student. If the system labels key is not passed, no existing labels will be removed from the student.

2.4 Contacts

Contacts in Handshake are used to keep track of employers, alumni, and other individuals who may not have a username and password for Handshake. The most common use for importing contact is to bring over employer relationships. Contacts can be labeled, sorted, tried to a Handshake employer, and more.

**Params**

<table>
<thead>
<tr>
<th>Header</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>*first_name</td>
<td>The first name of the contact (String)</td>
</tr>
<tr>
<td>*last_name</td>
<td>The last name of the contact (String)</td>
</tr>
<tr>
<td>*email_address</td>
<td>The email of the contact (String)</td>
</tr>
<tr>
<td>title</td>
<td>The title of the contact (String)</td>
</tr>
<tr>
<td>description</td>
<td>A description of the contact (Text)</td>
</tr>
<tr>
<td>employer_id</td>
<td>The Handshake id of the employer that you want to list the contacts for (int)</td>
</tr>
<tr>
<td>employer_name</td>
<td>The name of the employer that you want to list the contacts for (String)</td>
</tr>
<tr>
<td>location_attributes:name</td>
<td>The name of the location of the contact. NOTE: creates only.</td>
</tr>
<tr>
<td>phone</td>
<td>The contact’s phone number</td>
</tr>
<tr>
<td>cell_phone</td>
<td>The contact’s cell number</td>
</tr>
<tr>
<td>fax</td>
<td>The contact’s fax machine number</td>
</tr>
</tbody>
</table>

* Required

For an example file of the suggested fields [Click Here]

2.5 Appointments

You can import historical appointment records from appointments with students.

**Params**
### 2.6 Appointment Types

You can import appointment types to be used within Handshake.

**Params**

<table>
<thead>
<tr>
<th>Header</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>appointment_medium_name</td>
<td>The name of the appointment medium. Case sensitive, must be one of the configurable appointment mediums on your school.</td>
</tr>
<tr>
<td>appointment_type_name</td>
<td>The name of the appointment type. Case sensitive, must be one of the configurable appointment types on your school.</td>
</tr>
<tr>
<td>staff_member_email</td>
<td>The email of the staff member involved. Must be a staff member in the system.</td>
</tr>
<tr>
<td>student_email</td>
<td>The email of the student involved. Must be a student in the system.</td>
</tr>
<tr>
<td>start_date</td>
<td>The start date and time</td>
</tr>
<tr>
<td>end_date</td>
<td>The end date and time</td>
</tr>
<tr>
<td>description</td>
<td>A description of the appointment (Text)</td>
</tr>
<tr>
<td>status</td>
<td>[cancelled, requested, approved, rejected, no_show, started, completed] (String)</td>
</tr>
<tr>
<td>walkin</td>
<td>Was this appointment a walk-in? (Boolean)</td>
</tr>
<tr>
<td>import_identifier</td>
<td>This identifier must be completely unique, used if you are importing notes or labels on this appointment.</td>
</tr>
</tbody>
</table>

* Required
### Header

<table>
<thead>
<tr>
<th>Header</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>*name</td>
<td>The name of the appointment type</td>
</tr>
<tr>
<td>*length</td>
<td>The length of the appointment type in minutes (Integer)</td>
</tr>
<tr>
<td>id</td>
<td>The ID of the appointment you wish to update. Note this is optional and only required if you’re updating appointments and know the Handshake ID. Normally you can get this though Handshake Insights.</td>
</tr>
<tr>
<td>description</td>
<td>A description of the appointment type</td>
</tr>
<tr>
<td>pre_survey_id</td>
<td>The ID of a Handshake survey that the student will fill out as part of their appointment request</td>
</tr>
<tr>
<td>post_survey_id</td>
<td>The ID of a Handshake survey that will be sent to the student following their appointment</td>
</tr>
<tr>
<td>advisor_survey_id</td>
<td>The ID of a Handshake survey that the staff member may fill out once the appointment has started</td>
</tr>
<tr>
<td>pre_message</td>
<td>A message that will be sent to the student prior to their appointment</td>
</tr>
<tr>
<td>post_message</td>
<td>A message that will be sent to the student following their appointment</td>
</tr>
<tr>
<td>drop_in_enabled</td>
<td>Whether or not you would like students to be able to select this appointment type when checking into Drop In appointments (Boolean)</td>
</tr>
<tr>
<td>appointment_category_names</td>
<td>Names of appointment categories that this appointment type should be used for</td>
</tr>
<tr>
<td>student_screen_attributes:department_gpa_required</td>
<td>Whether or not a minimum department GPA is required to schedule this appointment type (Boolean)</td>
</tr>
<tr>
<td>student_screen_attributes:department_gpa</td>
<td>The minimum department GPA that a student must have to schedule this appointment type (Decimal)</td>
</tr>
<tr>
<td>student_screen_attributes:cumulative_gpa_required</td>
<td>Whether or not a minimum cumulative GPA is required to schedule this appointment type (Boolean)</td>
</tr>
<tr>
<td>student_screen_attributes:cumulative_gpa</td>
<td>The minimum cumulative GPA that a student must have to schedule this appointment type (Decimal)</td>
</tr>
<tr>
<td>student_screen_attributes:major_names</td>
<td>Names of majors that a student must be a part of to schedule this appointment type</td>
</tr>
<tr>
<td>student_screen_attributes:major_group_names</td>
<td>Names of major groups that a student must be a part of to schedule this appointment type</td>
</tr>
<tr>
<td>student_screen_attributes:school_year_names</td>
<td>Names of school years that a student must be a part of to schedule this appointment type</td>
</tr>
<tr>
<td>student_screen_attributes:institution_label_names</td>
<td>Names of labels that a student must have to schedule this appointment type</td>
</tr>
<tr>
<td>student_screen_attributes:college_names</td>
<td>Names of colleges that a student must be a part of to schedule this appointment type</td>
</tr>
</tbody>
</table>

* Required

## 2.7 Events

You can import historical events

### Params
2.8 Notes

File name: notes.csv

Schools may import notes onto various items in Handshake. The items can be a contact, user, job, appointment, or event.

<table>
<thead>
<tr>
<th>Header</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>*identifiable_type</td>
<td>One of [User, Contact, Job, Appointment, Event]. Case sensitive.</td>
</tr>
<tr>
<td>**identifiable_id</td>
<td>The Handshake ID of the identifiable (found in URL) <strong>Do not include unless you are linking an appointment or event created in Handshake</strong></td>
</tr>
<tr>
<td><strong>identifier</strong></td>
<td>If the identifiable_type is a User or contact, this is email. Otherwise it is the import_identifier</td>
</tr>
<tr>
<td>*user_type</td>
<td>If the identifiable_type is a User, the user_type must be specified.</td>
</tr>
<tr>
<td>content</td>
<td>The note contents</td>
</tr>
<tr>
<td>privacy_preference</td>
<td>If this is a personal note or shared with staff. [personal, institution]</td>
</tr>
<tr>
<td>reminder_date</td>
<td>If there should be a reminder associated with the note. See reference section for date formats.</td>
</tr>
<tr>
<td>written_at</td>
<td>The date the note was written at.</td>
</tr>
<tr>
<td>created_by_email</td>
<td>E-mail address of the author of the note. This must correspond to a user in Handshake.</td>
</tr>
</tbody>
</table>

* Required fields ** Either identifier or identifiable_id must be provided.

2.9 Labels

File name: labels.csv

Schools may import labels onto various items in Handshake. The items can be a contact, user, job, appointment, or event. This will simply apply labels. If a label already exists it will not apply a duplicate. It will not remove labels.
**Header** | **Value**  
--- | ---  
*identifiable_type | One of [User, Contact, Job, Appointment, Event]. Case sensitive.  
**identifiable_id | The Handshake ID of the identifiable (found in URL) Do not include unless you are linking an appointment or event created in Handshake  
**identifier | If the identifiable_type is a User or contact, this is email. Otherwise it is the import_identifier  
*user_type | If the identifiable_type is a User, the user_type must be specified.  
label_type | Either ‘normal’ or ‘public’. Defaults to ‘normal’.  
name | The label name to apply.  

* Required fields ** Either identifier or identifiable_id must be provided.

## 2.10 Campuses

File name: campuses.csv

Schools may import campuses into Handshake.

| **Header** | **Value**  
--- | ---  
*name | The name of the campus. This must be unique across your school.  
description | A description of the campus.  
location_name | The address of the campus.  

* Required fields

For an example file of the suggested fields [Click Here]

## 2.11 Majors

File name: majors.csv

Schools may import majors into Handshake. The columns DO matter - name should be column 1, major group names should be column 2.

| **Header** | **Value**  
--- | ---  
*name | The name of the major. This must be unique across your school.  
major_group_names | A semi-colon separated list of major group names that the major belongs to. Leave this blank to leave the major groups as-is.  

* Required fields

For an example file of the suggested fields [Click Here]

## 2.12 Minors

File name: minors.csv

Each row should contain the name.

For an example file of the suggested fields [Click Here]
2.13 Buildings

File name: buildings.csv

<table>
<thead>
<tr>
<th>Header</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>The name of the building</td>
</tr>
<tr>
<td>location_attributes:location_name</td>
<td>The location the building is in. This should be a geo-codeable address</td>
</tr>
</tbody>
</table>

2.14 Rooms

File name: rooms.csv

<table>
<thead>
<tr>
<th>Header</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>The name of the room</td>
</tr>
<tr>
<td>building_name</td>
<td>The name of the building. Must be a building already existing at the school.</td>
</tr>
<tr>
<td>capacity</td>
<td>The room’s capacity (integer)</td>
</tr>
<tr>
<td>available_start</td>
<td>When the room becomes available (datetime)</td>
</tr>
<tr>
<td>available_end</td>
<td>When the room is no longer available (datetime)</td>
</tr>
</tbody>
</table>

2.15 Attendees

File name: attendees.csv

<table>
<thead>
<tr>
<th>Header</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>student_email_address</td>
<td>The email address of the student to be checked in</td>
</tr>
<tr>
<td>registered</td>
<td>Boolean - Mark this student as pre registered?</td>
</tr>
<tr>
<td>checked_in</td>
<td>Boolean - Mark this student as checked in at the event?</td>
</tr>
<tr>
<td>*identifiable_type</td>
<td>Must be one of: Event or CareerFair (no space between words). Case sensitive.</td>
</tr>
<tr>
<td>**identifiable_id</td>
<td>The id of the identifiable.</td>
</tr>
<tr>
<td>**identifier</td>
<td>If the identifiable_type is a User or contact, this is email. Otherwise it is the import_identifier</td>
</tr>
</tbody>
</table>

2.16 FDS

File name: fds.csv

<table>
<thead>
<tr>
<th>required</th>
<th>header name</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRUE</td>
<td>student_email</td>
<td>the email address of the student that submitted that response</td>
</tr>
<tr>
<td>TRUE</td>
<td>first_destination_survey_name</td>
<td>the survey this response is associated with</td>
</tr>
<tr>
<td>TRUE</td>
<td>first_destination_response_type_name</td>
<td>must be one of ['Working', 'Volunteering', 'Continuing Education', 'Military', 'Still Looking', 'Not Seeking']</td>
</tr>
<tr>
<td>TRUE</td>
<td>email_address</td>
<td>length must be &lt;= 255 characters</td>
</tr>
<tr>
<td>TRUE</td>
<td>education_level_name</td>
<td>must be one of ['Certificate', 'Advanced Certificate', 'Associates', 'Bachelor', 'Masters', 'Doctorate', 'Postdoctoral Studies']</td>
</tr>
<tr>
<td>TRUE</td>
<td>import_identifier</td>
<td>must be a unique identifier for each response (e.g. naming convention)</td>
</tr>
<tr>
<td>TRUE</td>
<td>first_destination_survey_id</td>
<td>this is the Handshake ID, found within the URL when viewing survey</td>
</tr>
<tr>
<td>conditionally</td>
<td>employment_category_name</td>
<td>required if the type is 'Working', can be one of ['Organization', 'Entrepreneur', 'Freelancer', 'Fellowship', 'Temporary/Contract Work Assignment', 'Faculty Tenure', 'Faculty Non-Tenure']</td>
</tr>
<tr>
<td>conditionally</td>
<td>pay_schedule_name</td>
<td>required if salary is present, must be one of ['Hourly Wage', 'Annual', 'Monthly Stipend', 'Salary', 'Hourly Wage', 'Annual', 'Monthly Stipend']</td>
</tr>
<tr>
<td>conditionally</td>
<td>employer_name</td>
<td>required if the type is 'Working' or 'Volunteering'</td>
</tr>
</tbody>
</table>
### Table 2.1 – continued from previous page

<table>
<thead>
<tr>
<th>required</th>
<th>header name</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>conditionally</td>
<td>continuing_education_school_name</td>
<td>required if type is ‘Continuing Education’</td>
</tr>
<tr>
<td>conditionally</td>
<td>military_branch_name</td>
<td>required if type is ‘Military’, one of [‘Air Force’, ‘Army’, ‘Coast Guard’, ‘Navy’, ‘Marine Corps’]</td>
</tr>
<tr>
<td>conditionally</td>
<td>not_seeking_option_name</td>
<td>required if type is ‘Not Seeking’, can be one of [‘Taking time off’, ‘Taking a gap year’, ‘Traveling’]</td>
</tr>
<tr>
<td>conditionally</td>
<td>seeking_option_name</td>
<td>required if type is ‘Still Looking’, must be one of [‘Continuing Education’, ‘Employment’]</td>
</tr>
<tr>
<td>FALSE</td>
<td>primary_major_name</td>
<td>name of major which student should be reported under</td>
</tr>
<tr>
<td>FALSE</td>
<td>graduation_date</td>
<td>yyyy-mm-dd</td>
</tr>
<tr>
<td>FALSE</td>
<td>employment_type_name</td>
<td>must be one of [‘Full-Time’, ‘Part-Time’]</td>
</tr>
<tr>
<td>FALSE</td>
<td>location_name</td>
<td>the location of the student’s job</td>
</tr>
<tr>
<td>FALSE</td>
<td>status</td>
<td>one of ‘unsubmitted’, ‘in_progress’, ‘submitted’ (default)</td>
</tr>
<tr>
<td>FALSE</td>
<td>bonus_amount</td>
<td>must be a number</td>
</tr>
<tr>
<td>FALSE</td>
<td>salary</td>
<td>0 &lt; salary &lt;= 10000000 to two decimal places</td>
</tr>
<tr>
<td>FALSE</td>
<td>found_through_handshake</td>
<td>boolean</td>
</tr>
<tr>
<td>FALSE</td>
<td>job_position_name</td>
<td>student’s job title</td>
</tr>
<tr>
<td>FALSE</td>
<td>job_function_name</td>
<td>must be in large array of job functions</td>
</tr>
<tr>
<td>FALSE</td>
<td>start_date</td>
<td>must be after accept_date</td>
</tr>
<tr>
<td>FALSE</td>
<td>offer_date</td>
<td>must be after accept_date and before start_date and before ‘Today’</td>
</tr>
<tr>
<td>FALSE</td>
<td>accept_date</td>
<td>must be after offer_date and before start_date and before ‘Today’</td>
</tr>
<tr>
<td>FALSE</td>
<td>application_id</td>
<td>should be present if found_through_handshake is true</td>
</tr>
<tr>
<td>FALSE</td>
<td>fellowship_name</td>
<td>length must be &lt;= 255 characters</td>
</tr>
<tr>
<td>FALSE</td>
<td>military_rank_name</td>
<td>must be one of [‘Enlisted’, ‘Warrnt Officer’, ‘Officer’]</td>
</tr>
<tr>
<td>FALSE</td>
<td>continuing_education_major_name</td>
<td>length must be &lt;= 255 characters</td>
</tr>
<tr>
<td>FALSE</td>
<td>other_compensation_amount</td>
<td>0 &lt; salary &lt;= 10000000 to two decimal places</td>
</tr>
<tr>
<td>FALSE</td>
<td>employer_industry_name</td>
<td>must be in large array of employer industries</td>
</tr>
<tr>
<td>FALSE</td>
<td>specialization</td>
<td>length must be &lt;= 255 characters</td>
</tr>
<tr>
<td>FALSE</td>
<td>is_fellowship</td>
<td>boolean; default false</td>
</tr>
<tr>
<td>FALSE</td>
<td>other_employment_category_name</td>
<td>length must be &lt;= 255 characters</td>
</tr>
<tr>
<td>FALSE</td>
<td>employed_during_education</td>
<td>boolean but can be blank</td>
</tr>
<tr>
<td>FALSE</td>
<td>knowledge_source</td>
<td>length must be &lt;= 255 characters; default is ‘Survey Response’</td>
</tr>
<tr>
<td>FALSE</td>
<td>knowledge_response</td>
<td>boolean; default is false; used for NACE reporting - TRUE if this coming from anything other than the student directly</td>
</tr>
<tr>
<td>FALSE</td>
<td>authorized_to_work_in_us</td>
<td>boolean</td>
</tr>
<tr>
<td>FALSE</td>
<td>major_names</td>
<td>semicolon separated list of secondary majors</td>
</tr>
</tbody>
</table>

* Required fields ** Either identifier or identifiable_id must be provided.
3.1 How safe, reliable and secure is your application?

We have had a strong focus on reliability and security since the beginning. We’ve worked with industry experts for the last two years to ensure we are building the application to a very high standard of excellence in security and reliability.

A quick overview of a few of our features:

• 99.9% Uptime Policy
• Automatic horizontal scaling and load balancing
• Distributed across multiple regions for high availability
• Granular level access control
• Globally distributed content delivery network
• Specific disaster response team designated to handle any incidents
• Backed up daily with redundant follower databases
• Backup restoration is regularly tested
• DDoS mitigation
• Caching across the application
• Over 1000 automated tests verify features are working and system is continuously secure
• SSL Security on all client to app and inter-app communications
• All production deployments pass through a rigorous change management process and testing procedure
• We do monthly exhaustive penetration tests against our application
• Production access is strictly controlled, logged and regulated
• Our infrastructure provider is compliant or has been audited under the following:
  – ISO 27001
  – SOC 1 and SOC 2/SSAE 16/ISAE 3402 (Previously SAS 70 Type II)
  – PCI Level 1
  – FISMA Moderate
  – Sarbanes-Oxley (SOX)
• The data centers we use are ISO 27001 and FISMA certified
3.2 What SSO systems do you support out of the box?

We support SAML (and Shibboleth), LDAP and CAS. We are also more than happy to work with your to support your specific SSO solution if we do not already offer it. More information: Single Sign On

3.3 How do we get student data into the system?

There are two primary ways to get student data into the system, depending on your needs.

1. API

   If you want to have a stronger degree of control over our data and when it is updated in the system, we have a RESTful API that you may use to update, delete, create and synchronize student/staff/alumni accounts. API Documentation

2. Simple CSV dump

   With the Simple CSV dump, we ask for an export of data from your student information system (ie Banner, DataTel, etc). We then have you push the data to our secured and sandboxed S3 environment. Pushing data to S3 is simple and over a secure connection with the AWS CLI or our small python script that uses the REST S3 API.

   All of your data that you send us on S3 is not readable by anybody except for a single account that we tightly monitor and control. Even if your own credentials are compromised, an attacker will not be able to access the data.

   Student sync data is encrypted in transit and at reset with AES 256 bit encryption

   For more information, see this document: Data Transfer FAQ

3.4 Can we import data into the system?

We can work with you to help you export your data into a CSV format, and we can consume data from a CSV format into our system.

3.5 Do I need to dedicate any of my infrastructure for Handshake?

Handshake is provided as SaaS, so you do not need to dedicate any infrastructure to the application, besides what you need to send us up-to-date student data.

3.6 Does Handshake use SSL?

Yes, the production application forces SSL / TLS 1.2 to ensure that all interactions with the Handshake application are as secure as possible. The demo version of the application does not force SSL, and only test data should be used on the demo version.
3.7 Does Handshake use SFTP?

Handshake leverages a secure amazon web services s3 bucket to receive data on an automated basis. We also offer a secure web service for manually uploading data to handshake, our project management software.

We don’t currently support SFTP but let us know if you have trouble uploading to AWS and we’re happy to work though a solution with you.

3.8 Is a header row required in the CSV file?

Yes we do need a header for the file. But every header is not required to be in the file (e.g. if you don’t want to include a non-required field). Order also doesn’t matter.

3.9 Do you have a preffered format for Booleans?

True and False are the preferred format.

3.10 What happens if the email_address field is passed with a blank?

If the email is blank an error will be thrown and that row will be ignored.

3.11 Are column headers case sensitive?

Yes, column headers are case sensitive and must match what we have listed in the csv documentation.

3.12 Where can I find your terms of service and privacy policy?

Terms of Service
Privacy Policy
4.1 Why do you use Amazon S3?

AWS S3 is an industry standard for data storage and transfer. In order to provide the best security of your sensitive data, we use a rigorous access control policy and encryption in transit and at rest. Data can only be uploaded not read from AWS S3 and will only sit there for a short time before it is ingested into Handshake by an automated process.

4.2 What setup do I have do on my end?

You do not have to setup an S3 bucket or interact with Amazon in any way other than using one of the many tools available to upload to Handshake’s S3 bucket and folder for your team. We will provide you with the bucket, access id, secret key, and required prefix for uploading your file.

4.3 What if I can’t install python to use the aws cli?

The AWS cli is just interacting with the AWS S3 rest api, if you like, you can use any tool that can talk to a RESTful service such as curl. This isn’t recommended because it opens up more opportunities for errors. See here for an example: http://tmont.com/blargh/2014/1/uploading-to-s3-in-bash

4.4 What sort of encryption is used with the data?

We use server side 256 bit AES. “Amazon S3 Server Side Encryption employs strong multi-factor encryption. Amazon S3 encrypts each object with a unique key. As an additional safeguard, it encrypts the key itself with a master key that it regularly rotates. Amazon S3 Server Side Encryption uses one of the strongest block ciphers available, 256-bit Advanced Encryption Standard (AES-256), to encrypt your data.” All data sent in transit is transferred over TLS.

4.5 How should I format my [student/staff/alumni] data?

An up-to-date format for CSV files is included here: CSV Upload If you have any questions about formatting, or we don’t have a field included that you require, let us know and we are more than happy to work with you to ensure we meet all of your needs.
4.6 How do I transfer the data to my secure bucket?

The easiest way to transfer data is with the AWS CLI (http://aws.amazon.com/cli/)

Install the AWS CLI

You should have received an ‘Access Key ID’ and a ‘Secret Access Key’ from the Handshake team.

Type “aws configure” and follow the prompts to enter your Key ID and Secret Key mentioned above. Unless otherwise instructed by the Handshake team, you should use “us-east-1” as your region.

You may choose any default output format you wish.

For more information see this article.

Once you have the CLI set up, uploading your data is as simple as issuing the following command:

```bash
aws s3 cp [/path/your_local_file] s3://handshake-importer-uploads/[your folder]/[yyyymmdd]_users.csv
```

Handshake has one aws bucket ‘handshake-importer-uploads’ and every school has a unique folder which comes after the bucket which they only have write access to upload files.

example upload:

```bash
aws s3 cp 20140410_users.csv s3://handshake-importer-uploads/importer-production-hudson_university/20140410_users.csv
```

example response:

```bash
upload: to s3://handshake-importer-uploads/importer-production-hudson_university/20140410_users.csv
```

The AWS S3 API will respond with the document ID if the file was successfully transferred, otherwise it will respond with an error. You can also check the command exit code to determine it was successful.

If you wish to do a test run, simply send a file and our data team will verify that it is received and ready for production.

4.7 Can I whitelist s3 though my firewall?

There are a lot of IPs for s3 but they are updated here on a regular basis: https://ip-ranges.amazonaws.com/ip-ranges.json (we only use the us-east-1 region)

If you do want to whitelist by domain:

- https://s3.amazonaws.com/*
If you are attempting to do single sign on with Handshake, please check your Teamwork account for the form link to submit the Single Sign on Details we need.

5.1 Integration Guide

5.1.1 Getting Started

Fill out our survey here and start the conversation with one of our integration engineers: https://joinhandshake.wufoo.com/forms/handshake-single-sign-on-information/

5.1.2 FAQs

Do we support SAML?
  • Yes
Do we support InCommon, and if so, is it required?
  • No it is not required, but preferred
Do we support LDAP?
  • Yes
Do we support CAS?
  • Yes
Who are the users?
  • Students, Career Services, and sometimes faculty
Is the scope is limited to a certain group of users, how does Handshake determine authorization? By provisioning, attribute check, or group membership?
  • We limit access based on provisioning, we do not use group membership or any other attributes other then a uid to look the user up
What are the required attributes?
  • The attribute value sent must match the user’s username or auth_identifier
How are users provisioned for Handshake?
• Users can be provisioned in three ways:
  1. Imported vs CSV upload
  2. Via the API
  3. Have accounts created and approved in the Handshake UI via user signup process

What is our entityID?
• https://app.joinhandshake.com/sp

Where can I get a copy of Handshake’s Metadata?
• If you are on InCommon, see the metadata extract: http://md.incommon.org/InCommon/InCommon-metadata.xml
• If you are not on Incommon, you can download here: https://drive.google.com/file/d/0B-83VSAMe8ddNUhiTjVTUx4V1U/view

How can I setup SSO with Handshake?
• Start by filling out the form and someone will be in contact with you

5.2 Supported Integrations

5.2.1 CAS

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Login Path</td>
<td>The path a user should be sent to for authentication</td>
<td><a href="https://cashost:8443/cas-server-webapp/login">https://cashost:8443/cas-server-webapp/login</a></td>
</tr>
</tbody>
</table>

If you need to whitelist our ‘service’ or ‘system’, please whitelist:
• *app.joinhandshake.com/*
• *yourschoolname.joinhandshake.com/*

If you can not whitelist our domains and require a static IP address, please reach out and we can get this setup for you but this configuration is very rare.

5.2.2 SAML

If you are an Incommon Member you can use our entityID https://app.joinhandshake.com/sp

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Login Path</td>
<td>The path that a user should be sent to from our application (the Service Provider) for authentication</td>
<td><a href="https://idp.testshib.org/idp/profile/SAML2/Redirect/SSO">https://idp.testshib.org/idp/profile/SAML2/Redirect/SSO</a></td>
</tr>
<tr>
<td>name_id_format</td>
<td>See this article: <a href="http://stackoverflow.com/questions/11693297">http://stackoverflow.com/questions/11693297</a></td>
<td>urn:oasis:names:tc:SAML:2.0:nameid-format:transient</td>
</tr>
<tr>
<td>identifier</td>
<td>Which field in the user’s record should we use to look them up in the database? This field’s value should correspond with the username of the Handshake user id</td>
<td>0e82a794ce98b4c0e084ff1c9514270c0db941</td>
</tr>
</tbody>
</table>
5.2.3 LDAP

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>The LDAP server host</td>
<td><a href="https://ldap.myserver.com">https://ldap.myserver.com</a></td>
</tr>
<tr>
<td>Port</td>
<td>The port that LDAP is using for incoming requests</td>
<td>636</td>
</tr>
<tr>
<td>Service Username</td>
<td>The user our system should be using to perform LDAP searches</td>
<td>CN=Handshake,CN=Users,dc=test,DC=edu</td>
</tr>
<tr>
<td>Service Password</td>
<td>The password for authenticating the above user</td>
<td>dontuseawakpassword</td>
</tr>
<tr>
<td>Base DN</td>
<td>The DN we should search in for the user that is logging in</td>
<td>OU=Students,dc=test,dc=edu</td>
</tr>
<tr>
<td>Filter</td>
<td>Which field should correspond with the login name they use</td>
<td>(uid={{login}})</td>
</tr>
<tr>
<td>Firewall?</td>
<td>Is your system behind a firewall that will need us to provide a static IP for whitelisting?</td>
<td>yes/no</td>
</tr>
</tbody>
</table>

You may need to whitelist Handshake’s IP addresses to access your ldap server please allow the following IPs:

- 54.88.136.216
- 54.84.188.199
Email Delivery

Handshake maintains highly reputable emailing domains to ensure that emails are delivered. However, email delivery can sometimes fail due to the various systems in between Handshake and students which are designed to prevent email abuse. In order to maintain high deliverability, Handshake monitors and acts on spam scores, blacklisting, IP throttling and more.

6.1 Email Delivery Guidelines

6.1.1 Google

We recommend whitelisting all IP addresses in your Google Apps Console for Gmail as this will ensure no messages are flagged as spam. This can be accomplished by following this: https://support.google.com/a/answer/60751?hl=en

There are also some general guidelines for bulk emailing recommended by Google if you are mass emailing using Handshake: https://support.google.com/mail/answer/81126

6.1.2 Microsoft

We recommend whitelisting all IP addresses in your Exchange or Office 365 admin portals. This can be accomplished by following this: https://technet.microsoft.com/en-us/library/jj200718(v=exchg.150).aspx

6.2 Whitelisting

Handshake sends a large amount of email notifications and depending on your school’s use of the mass email feature, you should consider whitelisting handshake for more reliable email delivery. Whitelisting can better assure that emails sent to students and staff are received.

6.2.1 Whitelist by Domain

Domain: *.joinhandshake.com

6.2.2 Whitelist by IP

We send notifications from a dedicated set of IP addresses, you can whitelist these.
Regular Notifications are sent from: (notifications.joinhandshake.com)
• 192.237.158.52
• 192.237.159.133

Mass emails are sent from (mail.joinhandshake.com)
• 192.237.159.131
• 192.237.159.132

6.2.3 Whitelist by Email Address

We send from:
• handshake@notifications.joinhandshake.com
• handshake@mail.joinhandshake.com
• handshake@m.joinhandshake.com

It is possible to configure a custom reply-to email address, by default it is set to
• handshake@mail.joinhandshake.com
• handshake@notifications.joinhandshake.com
• handshake@m.joinhandshake.com
7.1 Please remember all values are case sensitive

7.2 Encodings

Handshake requires your file uploads to be UTF-8 (http://www.utf8-chartable.de/)
Example command to check your file’s encoding: `file ./input_file.csv`
Example command to convert your file to UTF-8: `iconv -f MACROMAN -t UTF-8 ./input_file.csv > ./output_file.csv`

7.3 Booleans

If specifying a boolean value (true or false) please use all lower case or all upper case letters Good: TRUE true FALSE false Bad: True False

7.4 Dates

When using dates or datetimes in the CSV or API, please provide them in one of the following formats:
- yyyy-mm-dd
- yyyy-mm-ddThh:mm:ss (assumes UTC time if no offset is present)
- yyyy-mm-ddThh:mm:ss offset (where timezone offset is either + or -, and hh:mm)
Example with offset: 2016-11-01T08:15:00-08:00

7.5 School Years

The following standardized list of school years are supported in Handshake:

- Freshman
- Sophomore
- Junior
- Senior
- Masters
- Doctorate
7.6 Gender

The following standardized list of genders supported in Handshake:

- Male
- Female
- Other

7.7 Work Authorizations

The following standardized list of work authorizations supported in Handshake:

- U.S. Citizen
- Student (F-1) Visa
- J-1 Visa (Exchange Program)
- Permanent U.S. Resident
- Employment (H-1) Visa
- TN Visa
- L1 Visa
- Work Card
- H4 Visa

7.8 Education Levels

The following standardized list of education levels are supported in Handshake:

- High School
- Associates
- Certificate
- Advanced Certificate
- Bachelors
- Masters
- Doctorate
- Postdoctoral Studies

7.9 Administrative Roles

The following roles are supported in Handshake:

- Experiences
- Articles
- Manage Own Appointments
- Manage All Appointments
- External Feeds
- Jobs
- Employer Approvals
- Manage Students
7.10 Salary Type Names

The following standardized list of salary types is supported in Handshake:

- Paid
- Unpaid
- Commission Only
- Commission Plus Salary

7.11 Job Type Names

The following standardized list of job types is supported in Handshake:

- Job
- Internship
- Cooperative Education
- Experiential Learning
- On Campus Student Employment
- Fellowship
- Graduate School

7.12 Employment Type Names

The following standardized list of employment types is supported in Handshake:

- Full-Time
- Part-Time
- Seasonal
7.13 Ethnicity

The following standardized list of ethnicities is supported in Handshake:

<table>
<thead>
<tr>
<th>Ethnicity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Native American/Alaskan Native</td>
</tr>
<tr>
<td>Black or African American</td>
</tr>
<tr>
<td>Asian/Asian American</td>
</tr>
<tr>
<td>Native Hawaiian/Pacific Islander</td>
</tr>
<tr>
<td>Latino(a)</td>
</tr>
<tr>
<td>White/Caucasian</td>
</tr>
<tr>
<td>Middle Eastern</td>
</tr>
</tbody>
</table>

7.14 Industries

The following standardized list of industries are supported in Handshake:

<table>
<thead>
<tr>
<th>Industry</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accounting</td>
</tr>
<tr>
<td>Advertising, PR &amp; Marketing</td>
</tr>
<tr>
<td>Aerospace</td>
</tr>
<tr>
<td>Animal &amp; Wildlife</td>
</tr>
<tr>
<td>Architecture and Planning</td>
</tr>
<tr>
<td>Automotive</td>
</tr>
<tr>
<td>Biotech &amp; Life Sciences</td>
</tr>
<tr>
<td>Civil Engineering</td>
</tr>
<tr>
<td>Commercial Banking &amp; Credit</td>
</tr>
<tr>
<td>Computer Networking</td>
</tr>
<tr>
<td>Construction</td>
</tr>
<tr>
<td>CPG - Consumer Packaged Goods</td>
</tr>
<tr>
<td>Defense</td>
</tr>
<tr>
<td>Design</td>
</tr>
<tr>
<td>Electronic &amp; Computer Hardware</td>
</tr>
<tr>
<td>Environmental Services</td>
</tr>
<tr>
<td>Farming, Ranching and Fishing</td>
</tr>
<tr>
<td>Fashion</td>
</tr>
<tr>
<td>Food &amp; Beverage</td>
</tr>
<tr>
<td>Forestry</td>
</tr>
<tr>
<td>Government - Local, State &amp; Federal</td>
</tr>
<tr>
<td>Healthcare</td>
</tr>
<tr>
<td>Higher Education</td>
</tr>
<tr>
<td>Hotels &amp; Accommodation</td>
</tr>
<tr>
<td>Human Resources</td>
</tr>
<tr>
<td>Insurance</td>
</tr>
<tr>
<td>Interior Design</td>
</tr>
<tr>
<td>International Affairs</td>
</tr>
<tr>
<td>Internet &amp; Software</td>
</tr>
<tr>
<td>Investment Banking</td>
</tr>
<tr>
<td>Investment / Portfolio Management</td>
</tr>
<tr>
<td>Journalism, Media &amp; Publishing</td>
</tr>
<tr>
<td>K-12 Education</td>
</tr>
<tr>
<td>Legal &amp; Law Enforcement</td>
</tr>
<tr>
<td>Management Consulting</td>
</tr>
<tr>
<td>Manufacturing - Other</td>
</tr>
<tr>
<td>Medical Devices</td>
</tr>
<tr>
<td>Movies, TV, Music</td>
</tr>
<tr>
<td>Natural Resources</td>
</tr>
<tr>
<td>NGO</td>
</tr>
</tbody>
</table>
Non-Profit - Other
Oil & Gas
Other Agriculture
Other Education
Other Industries
Performing and Fine Arts
Pharmaceuticals
Politics
Real Estate
Religious Work
Research
Restaurants & Food Service
Retail Stores
Scientific and Technical Consulting
Social Assistance
Sports & Leisure
Telecommunications
Tourism
Transportation & Logistics
Utilities and Renewable Energy
Veterinary
Wholesale Trade

7.15 Job Functions

The following standardized list of job functions are supported in Handshake:

Accounting
Actuary
Administration
Advertising, Media & PR
Architecture & Planning
Business Development
Community & Social Services
Construction / Contracting
Consulting
Counseling
Customer/Technical Support
Data & Analytics
Design / Art
Education / Teaching / Training
Engineering - Civil / Mechanical / Other
Engineering - Web / Software
Entrepreneurship
Environmental / Sustainability Mgmt
Financing
Fundraising & Event Management
General Management
Healthcare Services
Hotel / Restaurant / Hospitality
Human Resources
Information Technology
Legal
Logistics & Supply Chain
Marketing - Brand Management
Marketing - General
7.16 Time Zone Options

The supported options for time zones in Handshake are:

"American Samoa"
"International Date Line West"
"Midway Island"
"Hawaii"
"Alaska"
"Pacific Time (US & Canada)"
"Tijuana"
"Arizona"
"Chihuahua"
"Mazatlan"
"Mountain Time (US & Canada)"
"Central America"
"Central Time (US & Canada)"
"Guadalajara"
"Mexico City"
"Monterrey"
"Saskatchewan"
"Bogota"
"Eastern Time (US & Canada)"
"Indiana (East)"
"Lima"
"Quito"
"Caracas"
"Atlantic Time (Canada)"
"Georgetown"
"La Paz"
"Santiago"
"Newfoundland"
"Brasilia"
"Buenos Aires"
"Greenland"
"Montevideo"
"Mid-Atlantic"
"Azores"
"Cape Verde Is."
"Casablanca"
"Dublin"
"Edinburgh"
"Lisbon"
7.16. Time Zone Options
7.17 Major Groups

The following list of major groups is supported in Handshake. The categories are listed at the top level, with the major groups themselves underneath.

Arts and Design:
Architecture
Art History
Design and Applied Arts
Drama and Theatre Arts
Fine and Studio Arts
Graphic Design
Industrial Design
Interior Design
Museum Studies
Music and Music Education
Photography
Product Design/Packaging
Textiles and Clothing

Business and Entrepreneurship:

Accounting
Actuarial/Risk Analysis
Business Administration and Management
Business Analytics
Consulting
Economics
Entrepreneurship
Finance and Financial Management
Food Industry Management
Human Resources
Marketing
Operations Management
Parks, Recreation, and Leisure Studies
Real Estate
Retail and Hospitality Administration
Sales
Sport Business and Marketing
Supply Chain Management

Communications:

Advertising
Communication and Media Studies
Digital Communication
Documentary/Film
Journalism
Public Relations
Radio, Television, Media

Computer Science, Information Systems, and Technology:

Computer Programming
Computer Science
Cyber Security
Data Mining
Information Systems Management
Library Sciences
Software Design
User Experience/Social Computing

Education:
<table>
<thead>
<tr>
<th>Education Areas</th>
</tr>
</thead>
<tbody>
<tr>
<td>Early Childhood Education</td>
</tr>
<tr>
<td>Education Administration</td>
</tr>
<tr>
<td>Elementary Education</td>
</tr>
<tr>
<td>Health and Physical Education</td>
</tr>
<tr>
<td>Language Arts Education</td>
</tr>
<tr>
<td>Mathematics Education</td>
</tr>
<tr>
<td>Secondary Education</td>
</tr>
<tr>
<td>Special Education</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Engineering Areas</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aerospace Engineering</td>
</tr>
<tr>
<td>Agriculture and Biological</td>
</tr>
<tr>
<td>Biomedical Engineering</td>
</tr>
<tr>
<td>Chemical Engineering</td>
</tr>
<tr>
<td>Civil/Environmental Engineering</td>
</tr>
<tr>
<td>Computer Engineering</td>
</tr>
<tr>
<td>Construction Engineering &amp;</td>
</tr>
<tr>
<td>Management</td>
</tr>
<tr>
<td>Electrical Engineering</td>
</tr>
<tr>
<td>General Engineering</td>
</tr>
<tr>
<td>Industrial Engineering</td>
</tr>
<tr>
<td>Materials Science &amp; Engineering</td>
</tr>
<tr>
<td>Mechanical Engineering</td>
</tr>
<tr>
<td>Nautical/Naval Engineering</td>
</tr>
<tr>
<td>Network Engineering</td>
</tr>
<tr>
<td>Nuclear Engineering</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Health Professions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Athletic Training</td>
</tr>
<tr>
<td>Communication Disorders Sciences</td>
</tr>
<tr>
<td>Dentistry</td>
</tr>
<tr>
<td>Health/Exercise Science</td>
</tr>
<tr>
<td>Health/Hospital Administration</td>
</tr>
<tr>
<td>Kinesiology</td>
</tr>
<tr>
<td>Medicine</td>
</tr>
<tr>
<td>Movement Science</td>
</tr>
<tr>
<td>Nursing</td>
</tr>
<tr>
<td>Nutrition</td>
</tr>
<tr>
<td>Pharmacy</td>
</tr>
<tr>
<td>Physical/Occupational Therapy</td>
</tr>
<tr>
<td>Public Health</td>
</tr>
<tr>
<td>Speech Pathology</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Social Sciences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anthropology</td>
</tr>
<tr>
<td>Cognition &amp; Neuroscience/Biopsy</td>
</tr>
<tr>
<td>Counseling</td>
</tr>
<tr>
<td>Family and Consumer Science</td>
</tr>
<tr>
<td>Human and Child Development</td>
</tr>
<tr>
<td>Psychology</td>
</tr>
<tr>
<td>Social Work/Human Services</td>
</tr>
<tr>
<td>Sociology</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Civics and Government</th>
</tr>
</thead>
<tbody>
<tr>
<td>Criminal Justice/Criminology</td>
</tr>
<tr>
<td>Emergency Management/Homeland</td>
</tr>
<tr>
<td>Forensics</td>
</tr>
<tr>
<td>--------------------------------</td>
</tr>
<tr>
<td>International Studies/Comparative Politics</td>
</tr>
<tr>
<td>Law</td>
</tr>
<tr>
<td>Political Science and Government</td>
</tr>
<tr>
<td>Public Administration</td>
</tr>
<tr>
<td>Public Policy</td>
</tr>
<tr>
<td>Urban Planning</td>
</tr>
</tbody>
</table>

**Humanities and Languages:**

<table>
<thead>
<tr>
<th>Classical Studies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparative Literature</td>
</tr>
<tr>
<td>Creative Writing</td>
</tr>
<tr>
<td>Cultural and Ethnic Studies</td>
</tr>
<tr>
<td>English</td>
</tr>
<tr>
<td>Foreign Languages and Literature</td>
</tr>
<tr>
<td>Gender Studies</td>
</tr>
<tr>
<td>History</td>
</tr>
<tr>
<td>Linguistics</td>
</tr>
<tr>
<td>Philosophy/Ethics</td>
</tr>
<tr>
<td>Religious Studies/Divinity/Theology</td>
</tr>
</tbody>
</table>

**Life Science:**

<table>
<thead>
<tr>
<th>Animal Science</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anthropology/Zoology</td>
</tr>
<tr>
<td>Biology</td>
</tr>
<tr>
<td>Ecology</td>
</tr>
<tr>
<td>Epidemiology</td>
</tr>
<tr>
<td>Genetics</td>
</tr>
<tr>
<td>Immunology</td>
</tr>
<tr>
<td>Marine Biology</td>
</tr>
<tr>
<td>Microbiology</td>
</tr>
<tr>
<td>Physiological Science</td>
</tr>
</tbody>
</table>

**Math and Physical Sciences:**

<table>
<thead>
<tr>
<th>Chemistry</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physics</td>
</tr>
<tr>
<td>Mathematics</td>
</tr>
<tr>
<td>Statistics</td>
</tr>
</tbody>
</table>

**Natural Resources, Sustainability and Environmental Science:**

<table>
<thead>
<tr>
<th>Agriculture</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cartography</td>
</tr>
<tr>
<td>Conservation</td>
</tr>
<tr>
<td>Earth Sciences</td>
</tr>
<tr>
<td>Fisheries and Wildlife</td>
</tr>
<tr>
<td>Forestry</td>
</tr>
<tr>
<td>Geology/Mining</td>
</tr>
<tr>
<td>Natural Resource Management</td>
</tr>
<tr>
<td>Oceanography</td>
</tr>
<tr>
<td>Plant Sciences/Horticulture</td>
</tr>
</tbody>
</table>
Application Documentation

High level documentation for the various features in Handshake

8.1 Applications

Applications are how students to apply to jobs and interview schedules. Applications can have multiple statuses throughout their lifecycle, have documents attached as part of the application package, and are associated with the student’s profile. Jobs and interview schedules may be sometimes referred to as “applicables”.

8.1.1 Settings

Administrators can limit the maximum number of open applications per student. In addition, you can ensure that only approved documents are used to apply to applicables by using document and profile review.

8.1.2 Requirements

Jobs and interviews often have requirements which applicants must meet in order to apply. If these requirements are not met, the student will not be able to apply. The reasons for not being able to apply will be shown to the student.

Qualifications

Students must meet basic requirements such as school years, minimum GPA, majors and more. For more details on possible qualifications, see the qualifications documentation.

Document Requirements

Employers can require that certain types of documents be included in the application packages. Examples of required documents include Resumes, Sample Work, Cover Letter, Transcripts or ‘Other’. ‘Other’ can be customized on a per-applicable basis.

8.1.3 Applying to Interviews

Applying to interview schedules has two different routes, depending on how the interview schedule is configured. The most common application mode is ‘Students Apply through Associated Jobs’. The other option is to apply directly.
Students Apply through Associated Jobs

When this mode is picked, the student must apply to one of the associated jobs. While applying to that job, they will be given the choice if they want to also apply for the interview it is associated with. This option is on by default.

When the ‘Students Apply through Associated Jobs’ application method is chosen, the student is allowed to apply as long as they qualify for one of the associated jobs. Document requirements are also determined based on the job they apply through. The qualifications and document requirements for the interview schedule itself are not considered.

Students Apply Directly

With this mode, students apply directly to the interview. This is convenient for interviews that don’t have associated jobs.

8.1.4 Withdrawing Applications

As long as the application is still pending, it can be withdrawn by the student. When the student withdraws their application they will be prompted to also withdraw from associated interview schedules, if they’d like.

8.1.5 Managing Applications

As an administrator you are allowed to manage and view your student’s applications. On the management page you can filter and apply bulk actions.

8.1.6 Interview Schedule Application Priorities

Interview schedules have an advanced applicant review page. On this page you can filter and choose which applicants are selected to interview. Once applicants are selected, they can be reordered using a drag and drop interface to sort between primary and alternate (preselect only).

Primaries and Alternates

With preselect interviews, there are both primary and alternate applicants. Primaries can take slots before alternates, and the exact dates are configured on the interview details form.

8.2 Appointments

Appointments can be scheduled, tracked, and reported on in Handshake. We provide an interface for students to request appointments within constraints specified by advisors. Advisors can override constraints and manage appointments with students. Appointments can be restricted to a subset of students using student permissions.

Note: Appointments have a one-click report download option. Filter down to the relevant items and click the download button to quickly generate a report.
8.2.1 Settings

Categories and Types

As an administrator you can configure the categories and types of appointments. Categories are the first level of categorization of appointments. If you have no categories, the option will be removed in the interface. Otherwise students will be prompted to choose a category before choosing a type.

Each appointment is associated with an appointment type. Appointment types can belong to many categories.

Note: Common examples of appointment types are ‘Resume Review’, ‘Mock Interview’ or ‘Career Exploration’

Advisors each are required to specify the type of appointments that they are able to schedule appointments for.

Media

Appointment media specify how the appointment takes place. Examples may be ‘Face to Face’ or ‘Phone Call’.

8.2.2 Scheduling an Appointment

Both students and staff can schedule an appointment. Appointment availability is determined using appointment blocks and enforce when appointments are allowed be booked. Staff have the ability to override and schedule appointments for any start and end time.

Student Appointment Requests

When requesting an appointment, students will be walked through steps to filter and find the appointment that they are interested in. Students may filter by appointment category, appointment type and staff advisor. When filtering by advisor, a full week of appointment options will be shown to the student rather than the default of one day of options.

8.2.3 Scheduled vs. Drop-in Appointments

There are core types of appointments in Handshake: Scheduled and Drop-in appointments. Scheduled appointments are scheduled and requested before the student arrives. Scheduled appointments have reminders sent out ahead of the appointment to the student and shows on their calendar for reference. They can also be cancelled, if needed. Drop-in appointments cannot be scheduled ahead of time and as a result do not have reminders. Drop-in appointments are recorded as a full appointment record, similarly to scheduled appointments.

8.2.4 Appointment Blocks

Appointment blocks specify an advisor’s availability. Students will only be allowed to book appointments during availability.

Options

When setting up appointment blocks, options include:

Day of Week What day of week is this block for? Start and End Time For the day of week picked, what is the start and end time for this block? Drop-in Is this a drop-in appointment? Drop-in appointments cannot be scheduled
ahead of time. **Repeating** Is this a repeating (weekly) block? **Start and End Date** If it is repeating, specify when the repeating starts and ends. The block will not be repeated outside of those dates.

### 8.2.5 Notifications

Throughout the appointment scheduling process and leading up to the appointment, notifications are sent to the student. Notifications are sent:

- When the appointment is successfully scheduled.
- A 24 hour warning
- A one hour warning
- If the appointment is recorded as a ‘no show’.

**Note:** When the appointment is first scheduled and the student is notified, a calendar attachment will be included.

### 8.3 Attendees

Attendees are how students RSVP and attend events in Handshake. Attendee records can have a survey response for customizable survey questions, are attached to the student profile and can also require payment. Career fairs and events are attendable in Handshake.

#### 8.3.1 Statuses of Attendees

There are multiple ways an attendee status may be changed.

**Registered**

To register for an event, click the ‘Register’ button when viewing an attendable. By registering, the attendee record is now ‘registered’.

By registering for the attendable, the student will receive updates about it.

**Checking In**

Students can checkin to attendables using the *checkin kiosk*. Once checked in, the attendee record will be denoted as such.

**Paid**

If an attendable requires payment, attendees have a payment status. Payment will be prompted for during the registration process and students will not be allowed to register without paying. Payment is done using our integration with Stripe.
8.3.2 Viewing Attendee List

All attendables have a user interface for reviewing the attendee list. On the attendee review page, you can review all aspects of the attendee. This includes:

- basic student details
- student profile
- registration, checked in, and payment status
- survey question responses

Note: Attendees have a one-click report download option. Filter down to the relevant items and click the download button to quickly generate a report.

8.3.3 Inviting Attendees

For invite only attendables, students can be invited directly. The simplest way to do this is by attaching the event as a card to a mass email to take advantage of the auto invite feature.

In addition to mass emails, you may use the “Invite Attendees” page which provides a simple breakdown of the student demographic being invited.

8.4 Authentication

Handshake provides multiple convenient ways to authenticate and login to the application. Employers, students, alumni, mentors and administrators all may have different options available depending on the School’s Single Sign On systems.

8.4.1 Login Page

Handshake allows all users to login from any subdomain. This means that whether the student finds handshake at “app.joinhandshake.com” or “myschool.joinhandshake.com”, they’ll be able to login.

If “myschool.joinhandshake.com” is accessed, the student will intelligently be provided with relevant options based on their school’s SSO systems. Handshake supports SAML, CAS and LDAP integrations.

Configuring Your Login Options

Each school is allowed to present two login options to their users. The first slot is for Single Sign On users and the second slot is for Handshake authentication. To set up your login options, contact the Handshake team.

SSO vs. Handshake Authentication

While Handshake integrates with common Single Sign On technologies, it also provides its own authentication method which is available to any user.

Handshake Authentication is most often used by employers, mentors and alumni without SSO access. SSO Authentication is most often used by active students and administrators.
8.4.2 Automatically Fixing Misconfigured SSO Users

In some cases, students may have a misconfigured SSO setup. This most often happens in cases where students register for Handshake before their school joins. Handshake has an automated and seamless process for helping students connect their Handshake account with their SSO credentials with the following process:

1. User logs into SSO portal successfully and is redirected back to Handshake.
2. User is given a link to start the process.
3. Upon clicking the link, they are sent an email with a special link to click and a passcode.
4. User clicks on the link in their inbox and enters the passcode when prompted.

With those four steps, any student can connect their Handshake account to their school’s SSO.

8.4.3 Password Resets

Any user can reset their Handshake password at any time using the link on the login page.

8.4.4 Multiple User Types

Handshake allows users to have multiple account types, each connected with the same email address. For these users, you can easily switch between the different types using the top right dropdown switcher.

8.4.5 Logging Out

Users can logout using the top right dropdown. If you are signed in using SSO, we will also sign you out of your SSO portal.

8.5 Bulk Actions

Bulk actions are available on most manage and search pages. They provide a powerful way to interact with multiple records at a time.

8.5.1 Choosing Items to Act On

There are three ways to choose what records you want to act on.

Checkboxes

If you are working with a small (less than a page) amount of records, you may check the records of interest in order to apply the bulk action to those specific records.

Filtering

To act on records that meet a certain set of filtering criteria, simply filter down to the records of interest and perform the bulk action. The action will be performed on each page of results.
**All Records**

To act on all records, do not filter nor check any checkboxes and all records will be included. This will apply to each page of results, so be careful!

**8.5.2 Confirmation**

Because bulk actions are so powerful, we require confirmation before performing the action. This is to help ensure mistakes are not made. If the number of records being acted on is particularly high we will also ask you to type the number of records into a confirmation input, just to be sure.

**8.6 Career Fairs**

Career fairs are a dedicated feature in Handshake in addition to normal events. Career fairs can be managed fully with booths, registrations, payments, attendees and social media sharing.

**8.6.1 Settings**

Most settings for career fairs are on an individual career fair basis. Optionally, one may configure their payment preferences (for accepting credit card payments).

**8.6.2 Career Fair Options**

Career fairs are highly configurable and can fit a large array of functionalities. Both students and employers have a unique description for the career fair. Some commonly used configuration options include:

- **Student limit**: How many students are allowed to register?

**Visibility**

- **Status**: Drafting career fairs are not viewable by any users other than administrators.
- **Public**: Public career fairs can be searched for and found by students at other schools.
- **Hidden**: For employers only, career fairs that are hidden require a special invitation in order to view.
- **Invite only**: For students only, career fairs that are invite only require an invitation to attend. Students can be invited using mass emails or the attendee invitation page.

**Dates and Times**

Career fair start and end, student registration start and end, and employer registration start and end can all be configured.

Registration periods define when students and employers are allowed to attend or register for the career fair.
Payment Options

- **Allowed Methods**: You may choose to allow credit card, manual or both payment methods.
- **Student cost**: Students can be charged a flat rate for attending. For more details, see the attendees page.
- **Employer refund policy**: A refund policy that all employer registrations must accept.
- **Payment instructions**: Instructions shown to employers when they are asked to pay.
- **Invoice after approving**: As a convenience, administrators will be prompted to send an invoice after approving a registration.

Additional Questions

In addition to the normal registration questions, you may specify a custom set of questions that employers are asked when registering. Students may also be asked custom questions when registering.

8.6.3 Booth Options

Booth options are what employers must pick from when registering for the fair. Booth options can be configured for price, availability, and description. ‘Hidden’ booth options are not shown to employers; only administrators.

8.6.4 Extra Items

Extra items are purchasable amenities, aside from the booth, that employers may be interested in. Examples include electrical outlet access, internet access, lunches or advertisement space.

8.6.5 Discounts

Discount options can be added to career fairs. Only administrators are allowed to add discounts to registrations. Apply discounts before invoicing in order to ensure invoices are sent with accurate details.

8.6.6 Sharing With Schools

In addition to being public to all schools, career fairs can be shared with specific schools. When sharing your career fair with other schools, the students at those schools will be able to find, view and register for the fair. Administrators at those schools will also be able to view, but will not have access to configure or edit the fair.

**Note**: In addition to sharing with schools, you may also share with any consortia you are a part of.

8.6.7 Registrations

Employers register to career fairs through ‘Registration’ records. Registrations have multiple details associated with them including the approval and payment status, and are managed by administrators.

**Note**: Registrations have a one-click report download option. Filter down to the relevant items and click the download button to quickly generate a report.
Approval Status

In order to make registering for career fairs a seamless experience for employers, they are not required to obtain ‘employer approval’ to post registrations before registering. However, all registrations for a career fair must still be approved by administrators.

Payment Status

Registrations have a payment status which is automatically updated according to payments made and amounts due. For details on invoicing registrations, see the invoicing documentation. As an administrator you will be able to view the invoice and payment history for the registration.

Looking For

When registering, employers can specify what types of students they are looking for. This is specified through:

- majors and major groups
- job types
- employment types
- work authorization options

They can also specify the job titles they are hiring for.

Representatives

Employers are asked to list the number of representatives attending the career fair and to add details about any known representatives in the form of contact records.

Employers Editing Registrations

As long as the career fair registration period is still open, employers may edit a subset of fields on their registration. This includes ‘Looking For’ information, job titles and company description.

8.6.8 Booth Assignments

As booth options are configured, booths will automatically be built and made available for assignment. Registrations can be assigned to booths, and the employer will be informed of their booth number by viewing the registration on Handshake.

8.7 Checkin Kiosk

The Checkin Kiosk allows for students and attendees to checkin to events, career fairs, interviews, and appointments. Checkins are recorded in the system and can be reported on. Using the Checkin Kiosk requires the ‘Launch Kiosk’ permission.
8.7.1 Logging Out

When starting a checkin kiosk you will be prompted to either log out or stay logged in. Logging out ensures that the checkin kiosk can not be used to access the logged in user’s details since the user who launches the checkin kiosk will be logged out immediately. We always recommend logging out when launching.

8.7.2 Drop-in Appointments

In addition to normally scheduled appointments, drop-in appointments can and must be claimed and checked into using the checkin kiosk. For more information on drop-in appointments, see the appointments page.

8.7.3 Waiting Room

The Waiting Room feature can be used to display upcoming appointments, the staff member for that appointment, when the appointment is, and whether or not the student is checked in. This can be displayed in an advising waiting room for students to see.

8.7.4 Card Swipe Configuration

Checkin Kiosks can be configured to read in card swipe data. Setting up card swipes is considered advanced configuration and may require contacting the Handshake team.

Salting the Card Data: We allow card data to be salted before hashing and doing the lookup.

Card Regex: In order to know which input is a card swipe and which is a normal username or email lookup, specify a regex for matching card swipes. If the input at a checkin kiosk matches this regex we will assume it is a card and only send the card data.

Card Substring Regex: An optional second regex used to determine what data to send to the server for matching. If specified, only the first string match of the swiped card data will be sent.

8.7.5 Security

To securely look students up with possibly sensitive card ID’s - Handshake and the University first agree on a salt to append to the card_id and when the card is swiped. Handshake’s application then appends this salt, hash the results, and then looks the student up based on that value. This supposes that the university is also salting and hashing the card_id’s they are sending to Handshake in there student sync. Throughout the whole process the card id is never stored and security transmitted.

<table>
<thead>
<tr>
<th>Step</th>
<th>Process</th>
<th>Where it Happens</th>
<th>Security</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>With the kiosk page open, a user swipes their card and Handshake’s client side code applies a regex to the input validate that is indeed a card output</td>
<td>End User’s Browser</td>
<td>Sent via HTTPS</td>
</tr>
<tr>
<td>2</td>
<td>Handshake applies a second regex to the output of first to get the card_id from the string</td>
<td>Handshake Server</td>
<td>Card ID not stored</td>
</tr>
<tr>
<td>3</td>
<td>Handshake’s server then appends the pre-defined salt and SHA-512 hashes the result (optional)</td>
<td>Handshake Server</td>
<td>Card ID not stored</td>
</tr>
<tr>
<td>4</td>
<td>Handshake then looks up the value returned on all the user’s card_ids. The card_id is not logged only the hash.</td>
<td>Handshake Server</td>
<td>Card ID not stored</td>
</tr>
</tbody>
</table>
8.7.6 Nametag Printing

Checking into an attendable allows for nametag printing and can be configured on a per-attendable basis. To use nametag printing for an attendable, enable name tag printing and specify the XML layout you’d like to use on the edit page. Label fields named ‘Name’, ‘FirstName’, ‘LastName’, ‘Major’, ‘GradDate’, and ‘MajorGradDate’ are supported. When the student checks in a nametag will automatically be printed.

**Note:** Nametag printing integrates with Dymo printers only at this time. For more information about Dymo printers please visit http://dymo.com.

8.8 Contacts

Handshake implements a Contact Relationship Management feature through ‘Contacts’. Contacts are records of individuals for note taking, labelling, associating with jobs, events, career fairs, registrations, interviews and more. Note that contacts are separate from users. In order to login and use the Handshake interface, a user account must be created. For more information on the difference between contacts and users please see contacts vs users.

8.8.1 Managing your Contacts

You can manage your contacts on the ‘Contacts’ page. Contacts are private to only your institution and are not viewable by other institutions or employers. The notes and labels on contacts are also private. The sort of details that you can record and track with contacts includes email address, name, employer, contact information, and description.

Assigning Contacts

Contacts may be assigned to a member of your institution. Assigning contacts can be useful for splitting up responsibility. Contacts can then be filtered by the assigned to user.

Contact Sources

In addition to manually created contacts, contact records will automatically be created for you during special events. These events include approvals of job postings, employer approval requests, interviews, career fair registrations and more. When these contacts are automatically created the ‘Contact Source’ will also be recorded. This allows you to easily filter contacts by their source.

8.8.2 Bulk Inviting Contacts

In order to easily invite employers to use Handshake, contacts can be bulk invited through a bulk action on the management page. Bulk inviting contacts is especially convenient in that it allows you to:

- Automatically add the invited user to the equivalent contact’s assigned employer
- Create pending employer approval requests for the contact’s employer account once they join.
- Automatically approve any existing user’s employer in the Handshake system that you invite.
Behavior of Invitation Email Links

Depending on the scenario, the behavior of a contact invite has different behavior. Below are the potential scenarios for a given contact.

Contact’s company does not exist yet
1. Visit register page
2. Fill in details
3. Confirm your email address
4. Connect with schools and get an intro to Handshake, how it works
5. Don’t find company on search page -> create a new company
6. If you registered with a unique email domain then that email domain is set on the employer

Contact’s company does exist, their email domain matches their company’s
1. Visit register page
2. Fill in details
3. Confirm your email address
4. Connect with schools and get an intro to Handshake, how it works
5. Find company and click on ‘Join’
   1. If the company is distributed, your request must be approved
   2. If company is not distributed (only one with email domain) you are auto approved since domain matches

Contact’s company does exist, their email domain does not match or company does not have email domain or their email address is a generic one
1. Visit register page
2. Fill in details
3. Confirm your email address
4. Connect with schools and get an intro to Handshake, how it works
5. Find company and click on ‘Join’
6. Request is sent to company to be approved

Statistics

After the invitations are sent out, we will record statistics on the success and reach of the invitations. These statistics include how many users have registered using the invitation, how many existing employers have been approved, how many new employers have been approved, the number of contacts who already had a user account on Handshake, and more.

Because each email is personalized to the recipient based on their current status in Handshake, the invitation process can take up to an hour to complete and process fully for larger invitations.
8.8.3 Using Contacts

Contacts can be used when setting points of contact for jobs, interviews, events, career fairs, career fair registrations and more. By using contacts instead of users, you do not need to require points of contact to create a user account in order to be listed on the entity.

8.9 Consortia

Consortia are groups of schools on Handshake. They allow for easy sharing of data and cross-school communications.

8.9.1 Sharing Within a Consortium

Handshake allows users to share events, career fairs and interview schedules with other schools. In addition, if a school is a member of any consortia, they are given the option to share with their whole consortium. This will share with each school with one click instead of many.

Sharing Job Postings

Job postings can always be posted to individual members of a consortium in the same way that jobs can be posted to many schools. In addition, consortium members can be configured to have automatic distribution or automatic approval on a per-member basis.

**Automatic Distribution**: Members with this option will automatically have any postings they approve shared with the rest of their consortium.

**Automatic Approval of Distributions**: Members with this option on will automatically approve any postings shared to them from their consortium.

8.10 Conversations

Messaging and conversations can easily take place in Handshake. Through the messaging interface conversations can be initiated between two users. There are permissions about who can message who, many of which are configurable.

8.10.1 Viewing Conversations

When viewing your messages you can browse each user you are having a conversation with and details about that conversation. When viewing the conversation, you will see the messages and details about that user.

8.10.2 Messaging Permissions

There are different permissions about who is allowed to message whom.

**Messages from Employers**

Employers are only allowed to message students if they have been given the ‘Message Students’ role. They can message any career services staff user.
Messages from Students

Students are allowed to message career services staff and mentors at their school. They may only message employers if that option is turned on in the School Settings.

Messages from Staff

Career Services Staff may message students and mentors at their school as well as any employers.

8.11 Employers

An employer in Handshake represents a single company with a collection of users from that company. Each recruiter in Handshake belongs to an employer, and must join an employer before using the Handshake interface. Employers request access to schools as a whole, are given permissions, and build a single brand for students.

8.11.1 Employer Model

Handshake’s employer model is based around a single employer with multiple users (people) at that company joining the employer. This model is made possible by the fact that Handshake is a single platform of which employers only need to create one account to recruit at multiple schools.

When deciding what companies are unique or the same, email domain is taken into account. In some cases a parent company needs to be broken into multiple employer accounts on Handshake. This often happens with franchises which span multiple geographic locations and are highly separated, even though they share a common email domain. These sorts of companies are referred to as ‘Distributed Employers’ and are allowed to share a single email domain. The Handshake team monitors for distributed employers and merges distributed employers that are not meant to be.

Within a single employer exists multiple divisions. Divisions allow a separation of content within a single employer. Divisions can be specified for jobs and career fair registrations.

8.11.2 Employer Approvals

Before an employer’s users can post jobs, request interviews, search students and request events the employer must first gain approval. Employer approvals have a status, roles, and individual preferences around automatic approvals. The employer approval process is the first step in an employer’s goal of recruiting on a campus.

Employers can be assigned to a staff member once an approval request is submitted. The ‘Assigned To’ field can then be filtered on.

Roles

Post Jobs: Determines if the employer is allowed to post jobs.

Post Work Study Jobs: Determines if the employer is allowed to post, specifically, work study jobs. They must also have the base ‘Post Jobs’ role.

Post Events: Determines if the employer is allowed to post events such as info sessions.

Post Interview Schedules: Determines if the employer is allowed to request interview schedules.

Search Students: Determines if the employer is allowed to search students. Only students who have logged in and marked their profile as public will be searchable.
Message Students: Determines if the employer is allowed to message students directly.

Automatic Approval Options

In addition to the roles, employers can be approved for automatic job approval and automatic interview schedule approval.

Approve and View Next

To easily approve employers one by one, use the ‘Approve and View Next’ feature. This feature allows you to review an employer and set it’s status, roles, ‘Assigned To’ and other fields. When finished, choosing ‘Approve and View Next’ will take you to the next pending employer approval automatically.

8.11.3 Helping Employers be Successful

As an administrator, you have the ability to help your employers be successful on Handshake.

Inviting Employers

Your contacts can be invited to Handshake using a special link to streamline the process. See the contacts page for more information on bulk inviting.

Creating Employer Users

One of the easiest ways to add a new user account to an employer is to create the user account yourself. By viewing the staff management page for an employer and creating a new user for that employer, they will automatically be associated with that employer and receive an invitation email. From that email they can click a link to set their password and login.

Adding Existing Users by Email

In some cases an employer user may already have a user account created but not be able to join the employer they want to. In this scenario, you can add the existing user to the employer account from the staff management page.

Leaving an Employer

If an employer user accidentally joins the wrong employer, they can leave that employer to join another. Their data will be properly retained with either the employer they are leaving or their personal account, depending on the data (for example jobs and interviews will stay with the employer while profile information will stay with the user). The employer can leave their institution from their Account Settings page.

8.11.4 Duplicate Employers

In some cases, duplicate employers are created in Handshake. We closely monitor for these situations and run an automated merger process if the accounts should be merged. This process involves a sufficient warning to interested parties and the ability to back out of the merge. Once the merge goes through, all data will be preserved and merged to the new account.
8.12 Documents

Documents allow students to upload resumes, cover letters, sample work, transcripts and more. These documents are used in the application process. Documents can be required to be approved by a staff member. For more information on document or profile approval, see Resume and Profile Review.

8.12.1 Approval Process

There are two types of approvals that occur when a document is uploaded.

PDF Conversion

When a non-PDF format document is uploaded (such as docx or excel files), Handshake will automatically convert that document to be a PDF. Although the conversion generally goes smoothly, not all documents can be converted as exactly expected. For this reason we require students to approve or ‘sign off’ on the conversion to avoid improperly styled PDF documents being sent to employers.

The student will be able to easily review and approve the document after it is converted. Converting documents typically takes only a few seconds.

School Approval

As determined in the School Settings page, documents and profiles can be required to be approved by a Career Services Staff member. For more information on this please see the Resume and Profile Review page.

Changing Documents

Students may change the file of a document in only certain scenarios. If the document has been approved during document review, for example, it is not allowed.

8.12.2 Limitations

Handshake has requirements on the type and size of files students are able to upload. Documents must be no larger than 20 megabytes and must be a common office, text or PDF file.

8.13 Events

You can host and manage your on campus events in Handshake. Events in Handshake include Information Sessions, Workshops, Group Appointments and ‘Other’ events. Events have attendee management with registrations and check-ins, payments, comments section and more. Note that Career Fairs and Interview Schedules are considered special types of events in Handshake and have their own dedicated feature sets.

8.13.1 Attendee Management

Just like with Career Fairs, Events can have attendees. Attendees are records of users invited, registered, or checked in to an event. For more information about attendees, please view the Attendees page.
8.13.2 Sharing with Schools

Events can be shared with specific schools. When sharing your event with other schools, the students at those schools will be able to find, view and register for the event. Administrators at those schools will also be able to view, but will not have access to configure or edit the event.

Note: In addition to sharing with schools, you may also share with any consortia you are a part of.

8.14 Labels

Labels are a highly configurable way to tag entities in any way unique to your institution. Often times labels can be used to meet edge case workflows and resolve scenarios where Handshake does not have a specific field desired. There are three types of labels: ‘public’, ‘normal’ and ‘system’. Just about anything can be labelled including jobs, students, events, interviews, employers, contacts, registrations and outcomes.

8.14.1 Managing labels

Labels can be managed in School Settings. From the management page you can view each label, the number of items using that label, what type of label it is, and what sort of items are allowed to use that label.

8.14.2 How to Label

There are two ways to label items in Handshake: Individual labelling and bulk labelling.

Individual Items

To label an individual item, simply open that item and label using the sidebar. Existing labels will be shown, as well.

Bulk Labelling

Bulk labelling is an efficient way to label multiple items at once. The bulk labelling option is available as a bulk action. For more information on bulk actions, please see the bulk actions page.

8.14.3 Label Types

There are three types of labels in Handshake: ‘normal’, ‘public’ and ‘system’. Although each label type is different from the other, label options (which are configured in the settings page) cannot be created with the same name even if they are different types of labels.

Normal Labels

Normal labels are private to only you and your staff. These labels are not viewable by students, mentors, or staff of any other institution in Handshake.
Public Labels

Public labels are still only viewable by users at your institution, but are public to students. This includes labels on students themselves being public. Public labels are useful for sharing and categorizing items for students to see, such as a group of jobs or employers.

System Labels

System labels are private like normal labels, but are not manageable on the Handshake interface. Instead they are automatically synced in through the data received by the registrar office. When user syncs happen, any labels specified will be added to the student as a system label. Note that if that label already existed as a normal label it will be converted to a system label.

Applying or Removing System Labels

System labels are meant to be added and removed only during student data syncs. This ensures that the system labels are accurate and consistent. For this reason, they cannot be added or removed through the user interface.

8.14.4 Searching with Labels

Once items are labelled, they can become useful through the search interface.

8.15 Mass Emails

Mass Emailing allows staff to send targeted emails to large numbers of recipients. It gives full control over who receives the mass email, the content in the email, and has additional functionality such as attaching cards.

8.15.1 Setting the Recipients

There are generally two ways to initiate a mass email to a set of recipients.

Searching for Students

The simplest way to start a mass email is by directly visiting the ‘New Mass Email’ page. Using this method will present you with a search page to narrow down and find the students you wish to email. This will also provide with a graph breakdown of the recipient demographic.

Preset Recipients

Mass Emails may also be created using a preset set of recipients. Examples of these situations include mass emailing everyone who has not yet responded to a survey, the attendees of an event, or a set of applicants. With this option no graph breakdown will be shown.

In some scenarios, there will be a mix of contacts and users when using preset recipients. In this situation only users will be able to login to links and cards attached to the mass email. For more information on why, see the users vs. contacts page
8.15.2 Writing the Email

Writing the email is done using a simple, What You See is What You Get interface. It allows for rich text such as colors, bolding, different sizes, links, images, and more. For the more advanced users, an HTML editor is also allowed. Although the mass email editor is highly configurable, some styling options are not available for security reasons.

Attaching Cards

A common pattern in mass emails is to email about something in Handshake such as jobs, events, career fairs or surveys. To make this use case easier, consider using cards. Mass Email Cards are pre-styled information included at the base of the mass email about the item desired.

In addition to being a convenient way to add information about something in Handshake to your mass email, certain cards will also automatically invite the recipient to the card. Mass emails that include cards for career fairs, events or surveys have this behavior.

8.15.3 Finishing the Email

To ensure no simple mistakes are made, a review and confirmation process is included.

Review Page

Once your mass email is ready to go, a review page is shown before moving on. On the review page you can see a high level overview of the mass email and confirm that all of the details are correct. Once reviewed, finalize the mass email to move forward.

Sending Test Emails

Once a mass email is finalized, it cannot be edited by you or other staff. From this page you can review the content of the mass email and send yourself a test email for review. As long as the email is correct, the mass email can then be sent.

8.15.4 Variables in Mass Email

In order to personalize mass emails, you can use variables in your email body. The variables available should be wrapped with ‘%’ tags when being used, for example ‘%recipient.first_name%’, without the quotes.

The variables available are ‘recipient.username’ ‘recipient.name’ ‘recipient.first_name’ ‘recipient.calculated_first_name’ ‘recipient.last_name’ ‘recipient.email_address’ ‘recipient.institution_name’

Note: The ‘calculated_first_name’ variable uses the preferred first name if available, and otherwise uses their first name.

8.16 Notes

Handshake has a system wide option for taking notes on various items. Notes have different privacy levels, none of which allow students to view the notes of staff. Almost anything on Handshake is noteable including jobs, students, employers, events, interviews, contacts and more.
8.16.1 Writing a Note

Notes use the WYSIWYG editor for writing and editing notes. Attachments can be added to notes. In addition to the note content itself, there are various options that go with the note.

Privacy Options

Notes can be either private to yourself or shared with your staff. Notes shared with the staff are viewable by anyone with the permissions to view shared notes.

Mentions

Notes can have multiple ‘mentions’. Mentions allow you to notify other users about the note through a common syntax. To mention another user, type ‘+’ or ‘@’ followed by the user’s email address. An autocomplete option will be shown.

Reminders

Each note can have a reminder attached. For notes with reminders attached, the creator of the note as well as anyone mentioned in the note will be notified on the specified date.

8.16.2 Editing Notes

Any note written by you can be edited using the same options as when writing a new note.

8.16.3 Activities

For your information, important activities are included in the notes list. These activities provide valuable information about the noteable item and its history.

8.16.4 Notes from Email

If you would like to send student notes from your email to Handshake you can do so by forwarding the messages to the email address provided in your User Settings.

8.17 Interview Schedules

Interview Scheduling is a core feature in Handshake and a special type of Event. Interview scheduling in Handshake supports Room Conflict Management, setting qualifications, slot management, timelines, automated slot swapping, and applicant management. Interview Schedules can be restricted to a subset of students using student permissions.

Note: Registrations have a one-click report download option. Filter down to the relevant items and click the download button to quickly generate a report.
8.17.1 Interview Schedule Preferences

There are many configuration and templating options available for interview scheduling. Configurations allow you to customize interview scheduling for your school and templating makes the interview schedule administration easier and automated.

Interview Seasons

Use interview seasons for establishing when interview schedules can be requested and for what dates interviews can take place. These seasons are enforced for employers requesting interviews at your school, but can be overridden by career services staff.

Interview Schedule Templates

Interview Schedule templates allow slots to be built in one click using pre-determined slots. Templates can be hidden from employers to limit the options available to them. As long as templates are available to the interview schedule requester, a template must be chosen.

Interview Schedule Timelines

While templates are used to easily build out slots for an interview schedule request, timelines are used for establishing the dates of the interview schedule. The dates included in the timeline are application periods, signup periods, and reminder dates.

Because the relevant dates for an interview schedule depend on the type of schedule, timelines also have an Interview Schedule Type associated with them. Specifically, the signup period is only relevant to Preselect interview schedules.

As long as a timeline is available to the interview schedule requester, a timeline must be chosen. In addition, employers will not be allowed to manually choose their Interview Schedule Type or dates if timelines are available to be chosen.

Room Conflict Management

Room Conflict Management can be set up to enforce limitations on the number of interviews that can take place per day. To use room conflict management, you must first choose which rooms are available for interviews. Once you have established which rooms are available, the Handshake System will automatically ensure that there are never more interviews than available rooms for a given day. Career Services Staff can override room conflict management limits.

Additional Configuration

In addition to the above, there are options for the following.

Room Costs Specify a room cost to charge employers for their rooms. This cost is based on the price for a room for one day. Maximum Rooms per Request To limit the number of rooms an employer can request per schedule, use this option.

8.17.2 Building an Interview Schedule

Building an interview schedule requires entering basic details such as the employer and description, timeline details such as interview dates and timelines, and an application method such as qualifications or associated jobs.
Application Methods

Interviews can be configured to receive applicants through two methods. The default and recommended option is to apply through jobs. When applicants apply through jobs, they will be asked which associated job they want to apply to. The other option is to apply directly without having to associate any jobs. For this option, you will specify the qualifications for the interview schedule.

For more information about managing applicants, see the Applications Page.

Associating Jobs

By associating jobs to an interview schedule, you are allowing students to apply to the interview schedule through those jobs. The applicants will be shown on the interview schedule’s applicants page like normal.

When a job is associated with an interview, all applicants for the associated job will also be applied to the interview as long as the interview schedule is 1) approved and 2) not yet finished with the application period. If those two conditions are not met, the applicants will be drawn in as soon as they are met.

Qualifications

If you choose to have students apply directly, you will be prompted to specify the qualifications for the interview schedule. For more information on setting qualifications see the Qualifications page.

Approval Process

Once an interview schedule is requested, it will take one of two routes.

1. If the interview schedule is being requested by/for an employer that has automatic interview schedule approval turned on, it will automatically be approved.
2. If the interview schedule is being requested by/for an employer that does not have automatic approval, it will be listed as pending for your approval.

Interview Dates

Each interview schedule must have at least one interview schedule date specified. Interview schedule dates specify the dates on which interviews will take place. If room conflict management is enabled, the remaining room counts will be shown and automatically updated as the counts change.

8.17.3 Managing Interview Slots

Interview slots represent the time, student, room, and interviewer of an interview. Interview schedules have multiple interview schedule dates, each of which has multiple interview slots. Interview slots can be edited individually or in bulk.

Associating a Job

Interview Slots with an associated Job are reserved for applicants who applied to the interview through that job. Other applicants will not be able to take that slot. If there is no associated slot, it is open to all applicants.
Adding Breaks

To add a break to an interview slot, mark the slot as “Unavailable”.

**Note:** You may also enter the reason for the break in place of the ‘Interviewer Name’ field.

Building Slots Automatically

Slots can be built automatically using the template specified for the interview schedule. When automatically building slots, you can also specify a room to use for each room requested.

**Note:** As an administrator you can override the template used at any time.

8.17.4 Interview Swap Requests

Interview slot swap requests are an automated, controlled system for letting students request slot swaps with other students. The swaps only take place if both students agree to the swap, and happen as one single transaction to ensure no student will lose their slot during the swap. Interview slot swaps may only occur during the sign up period; once signup end is reached, slot swaps are cancelled and unavailable.

As an administrator, you may view existing slot swap requests and their status.

8.17.5 Timeline Status

Interview schedules have a series of dates, as specified by the interview schedule timeline. These dates are automatic triggers for system events as the interview schedule moves through each ‘Timeline Status’. When a schedule moves into the next timeline status, relevant email notifications and reminders are sent to the appropriate parties.

8.17.6 Interview Schedule Types

There are four types of interviews in Handshake.

- **Room Only:** Interview slots in a room will be reserved. Handshake will not be used for signing up students.
- **Open:** Specify an application period in which students that pass the specified qualifications are allowed to take slots.
- **Preselect Continuous:** Specify an application period in which students that pass both the specified qualifications as well as are manually approved are allowed to take slots.
- **Preselect:** Specify both an application period and a signup period. Primary and alternate student choices are selected after the application period. Primaries are allowed to sign up before alternates.

**Note:** Preselect interview schedules are the only schedules that have a signup period. The other types of interview schedules have students apply and sign up in the same time period.
8.17.7 Sharing with Schools

Interview schedules can be shared with specific schools. When sharing your interview schedule with other schools, the students at those schools will be able to find, view and register for the schedule. Administrators at those schools will also be able to view, but will not have access to configure or edit the interview schedule.

*Note:* In addition to sharing with schools, you may also share with any consortia you are a part of.

8.17.8 Print Packets

Interview schedule details can be printed by room, employer or employer and room. To print these packets you will need to include the date of the interview, how you would like the interview packet grouped and the items you would like included in the interview packet.

8.18 Invoicing

As a part of the payment processing features in Handshake, Invoicing is included. Employers can be invoiced for career fair registrations and interview schedules. Invoicing sends emails to the relevant parties with the invoice details.

8.18.1 Sending an Invoice

**Recipients**

You are able to choose the recipients of an invoice. The potential recipients include the registrants and the associated contacts.

**Options**

**Invoice Number:** Each invoice has an associated number. This is a readonly field for your reference.

**Amount:** The amount will be automatically filled to the correct amount, but this can be changed for unique scenarios.

**Subject:** This will be the subject of the email received.

**Custom Message:** If you have any extra details to send, specify them here. A default message will be filled for career fair registrations.

**Invoice Email Contents**

Once the invoice is sent, an email with details about the invoice will arrive shortly. Details in the invoice include the invoice number, subject, custom message, contact information, previously paid charges, previous invoices, itemized list of the invoice amounts, and directions on how to pay. At this time no PDF attachment is included, but is planned.

8.18.2 Registration Invoicing

For career fair registrations we take into account the booth option, extra items and discounts applied for the invoice total. For more information on career fair registrations, please see the *Career Fairs Page.*
8.18.3 Interview Schedule Invoicing

For interview schedule invoicing, we take into account the number of daily room usages for the invoice amount. If automatic approval is turned on, an invoice will be sent and payment will be required as a part of the request. For more information about interview scheduling please see the Interview Schedules Page.

8.19 Job Postings

Handshake has a full featured job creation, posting, sharing and applicant management interface. With Handshake’s unique employer model, jobs have many job postings which allow jobs to easily be posted to multiple schools. Each job posting has an approval status. Jobs can have only one unexpired posting per school. Jobs can be restricted to a subset of students using student permissions.

Note: Postings have a one-click report download option. Filter down to the relevant items and click the download button to quickly generate a report.

8.19.1 Posting a Job

Posting jobs is broken up into a few separate sections. There are basic details such as title, job type and employment type, details such as location, contacts, pay rate, description fields, qualifications and postings management.

Engaging Applicants

In order to engage applicants, it is recommended to include high levels of detail in job postings. Make sure that your job is reaching a desired amount of students through the qualifications breakdown, and make sure that your employer profile is fully built out.

Setting Qualifications

Jobs allow for qualifications to determine what students are allowed to apply. See the Qualifications Page for more information. When qualifications are specified for a job, they are applied to every school that the job is applied to. The one exception to this rule is majors; only the majors relevant to the school will be considered.

Work Study Permissions

Jobs may be marked as work study jobs. A special role is required for an employer to be post work study jobs. Work study jobs will only be shown to students who are work study eligible if work study eligibility is enforced. The option to enforce work study eligibility can be changed in the School Settings.

Renewing a Job Posting

Coming soon.

8.19.2 Job Contacts

Jobs may have points of contact be listed. The points of contacts are used for sending new applicant packages and applicant summaries when jobs expire. The contacts may also have details shown to students, if desired.
8.19.3 Applying for Jobs

For details on applying to jobs, see the Applications Page.

8.19.4 Approve and View Next

To easily approve postings one by one, use the ‘Approve and View Next’ feature. Similar to Employer Approvals, this feature allows you to review a posting and set it’s status, expiration date, ‘Assigned To’ and other fields. When finished, choosing ‘Approve and View Next’ will take you to the next pending posting automatically.

Job Search Relevancy

In order to provide relevant jobs to students, the most relevant jobs are pushed to the top of the search results and less relevant jobs are pushed to the end of the results. Relevancy is based on numerous factors including whether or not they qualify, the student’s major, what employers the student follows, popularity and more.

Note: Although often the jobs that the student qualifies for are pushed to the top, sometimes jobs that the student does not qualify will be more relevant then ones they do qualify for. Students may opt to filter out jobs they do not qualify for using the search interface.

8.20 Mentorships

Handshake can facilitate an array of mentoring programs. Mentorships provide a connection between students and mentors, and can be managed by administrators. Mentorships can be restricted to a subset of students using student permissions.

8.20.1 Configuration Preferences

Mentoring will only show to students if there is at least one active mentor at your school.

Mentors Mange Mentorships

If this option is turned on, mentors at your school can respond to mentorship requests and view pending and declined mentorships. With this option turned off, mentors will only be able to view their ongoing and completed mentorships.

Mentor Survey

If you’d like your mentors to complete any additional questions outside of the default Handshake fields, you may specify a survey with those custom fields.

8.20.2 Mentor User Types

The mentor user type is separate from student, employer and staff accounts and is generally limited in what they have access to. Mentor registration can be done like normal on the registration page. If the mentor registers on your school’s domain, a request will automatically be submitted.
Mentors have a one-click report download option. Filter down to the relevant items and click the download button to quickly generate a report.

Mentor Options

There are additional fields available to mentor profiles.

**Student Contact Preference**: Determines how students are allowed to communicate with this mentor.

**Maximum student contacts per Month**: Determines how many students may contact this mentor, per month.

**Maximum mentees**: The maximum number of ongoing mentorships this mentor is willing to accommodate.

**Industry**: Specifies what industry this mentor is in, for better matching of students.

**Advice and Hobbies**: Quick blurbs for the mentors to provide, to help students find good matches.

8.20.3 Setting up a Mentorship

Setting up a mentorship involves three fields. The first two are the mentor and student involved. The last is the status. The status may be one of ‘requested’, ‘declined’, ‘ongoing’ and ‘completed’. Ongoing mentorships are in progress and active.

Communications

Mentors and students can communicate through Handshake’s messaging feature once a mentorship is established.

8.21 Notifications

In order to keep up to date on changes in Handshake important to you, Handshake supports a configurable notification system. Users can receive notifications for all important activities in Handshake and can choose how they receive those notifications.

8.21.1 Editing Notification Preferences

To edit your notification preferences, visit the ‘Notification Preferences’ page under ‘Account Settings’. Each type of notification has its own preference. By default, a set of default notifications are turned on for each new user.

8.21.2 Following

In addition to being able to track desired items, following plays a role in deciding what you are notified about. To denote you want to receive notifications about a specific item, follow that item.

8.22 Payments

Payments for all types of events can be processed and managed through Handshake. Payments can be either required upfront or paid after approval and an invoice being sent. See the invoicing page for more information on sending invoices. Handshake uses Stripe as its payment processor.
8.22.1 Receiving Payments

There are two ways to receive payments.

Credit Card Payments

The simplest and most automated way to receive payments is through Credit Card payments. For some scenarios, this is the only option available, including scenarios where payment is required immediately.

Manual Payments

In some scenarios you may want to receive a manual payment and record that payment in Handshake. This option is currently only available for career fair registrations and interview schedules.

8.22.2 Integration with Stripe

Handshake uses Stripe for all Credit Card processing. Stripe is a highly trusted payment processor used by some of the world’s most trusted companies. Before accepting Credit Card payments on Handshake, you will have to set up and configure Stripe. For help with this process, simply reach out to the Handshake team.

Issuing Refunds

If you need to issue a refund you may use the Stripe dashboard.

8.23 Posts

Posts are a simple, quick way to share text and web site links with students, employers or both. Posts appear on the student’s dashboard.

8.23.1 Writing a Post

When writing a post you will have the ability to target the post and include a link.

Targeting Posts

Posts can be targeted to either your students, employers or both. For employers, they may choose to target either students or schools.

Including Links

To include a link in your post simply paste or type the url and a preview link will automatically be attached.

8.24 Qualifications

Qualifications are used by various items to limit what students are allowed to view and apply to different content. Qualifications are used for jobs, interviews, articles, pins, career fairs and events.
8.24.1 Setting Qualifications

There are numerous types of criteria that can be specified in qualifications.

**School Years**

Specific school years can be chosen to only allow students of a certain school year to be qualified. Leave blank to allow all school years.

**GPA**

To qualify only certain GPA’s, require either a minimum department GPA or cumulative GPA.

**Majors and Major Groups**

Handshake uses a concept of Major Groups to simplify qualifying by major. Major Groups are a generalized collection of majors, specified by each school. Choosing major groups allows you to easily pick a common set of majors without requiring knowledge of the different majors at each school and how they differ.

In addition to choosing major groups, individual majors may also be picked. Any student who matches individual majors or a major within a selected major group will qualify.

**Colleges and Labels**

Colleges and Labels are qualification options only available to school staff. Choose specific colleges or labels to require them, otherwise leave blank to impose no restrictions.

8.24.2 Automatically Updated Counts

In order to easily see how many students qualify for the qualifications being set, you are provided with a real-time student count. Use this to fine tune your qualifications to the desired visibility.

8.25 Document and Profile Review

Career services staff may turn on the document review or profile review option. These options require that student documents are reviewed and approved before being used in applications. These options can be turned on in the School Settings.

8.25.1 Using the Review System

The profile and document review interface is a single page which cycles through users one at a time. The users are ordered in order of submission date, first in first out.

Users are reviewed one at a time and each user is reviewed as a whole including their profile and all documents. The profile and documents can be navigated on the left hand side with the content on the right. Each item is reviewed individually.
Skipping Users

Users can be skipped. Skipping a user puts them at the end of the queue.

Approving All and Future

Once a user is trusted to upload acceptable document and profile content, they may be approved for all future content. With this option they will no longer require approval before using documents.

Providing Feedback

Feedback can be added to profiles and documents using either the messaging feature or commenting feature on documents. At this point there is no inline comments for document review.

8.25.2 Using the API

In addition to setting a user's profile or document review status on the interface, the statuses may also be set through the API and user sync. This allows, for example, to set a certain subset of users as automatically approved.

8.26 RSS Feeds

The recommended option for easily and safely exporting data from Handshake is through RSS feeds. RSS feeds are live, readonly feeds of events, jobs or career fairs. Each RSS Feed has a private unique link for loading the data.

8.26.1 Creating an RSS Feed

RSS Feeds include two steps. The first step is specifying the name and category, and the second step is scoping the data you want in the feed. Once the feed is set up, the final page can be used to generate the URL.

8.26.2 Using RSS Feed

RSS Feeds can be used for different integrations. You may want to consult IT to help set up and make use of your feed. Note that through the nature of RSS Feeds, data is made public without requiring authentication, and care should be given in handling the external facing RSS Feed links.

Note: Examples may include a list of upcoming events, “hot jobs” using labelling, or newly posted jobs.

8.27 Student Permissions

Numerous features in Handshake can be limited to a subset of students using labelling. Applying student permissions requirements is for the whole feature set. When students are restricted to a particular feature, they will be told so, shown a custom message, and given a way to message staff for more information.
8.27.1 Using Labels for Permissions

When setting up the student permissions you will be asked to choose a label that students either must or must not have in order to access the feature. There are no restrictions on the type of label you choose.

8.27.2 Student Reception

A custom message can be specified to show to students when they are given the permission denied message. The message should be informative about why they are not allowed and what the student can do to gain access. They will also be shown a list of staff members to message for more information.

Note: We will only show the student staff members that are marked as points of contact.

8.28 User Links

There are four different types of users in Handshake: administrators, employers, students and mentors. You may build an account for each type of user using the same email address. When you do this, each of your accounts will be shown in your user dropdown button which you can easily switch to at any time.

8.29 Users and Students

Users in Handshake represent an account that can be logged into and used to access the interface. There are four types of users in Handshake: Administrators, Students, Mentors and Recruiters.

8.29.1 Student Registration

In most cases, student accounts will be imported and ready for students to claim. When student accounts are not imported, students must request access to join their school. There are two ways for students to connect with their school.

Register Using School Domain

If the student registers for an account using the correct domain (for example, schoolname.joinhandshake.com) and they register with their .edu email address, they will be automatically connected with their school upon confirming their email.

If they register on the correct domain, but don’t use their .edu email address, a ‘Student Request’ will be created for approval by school administrators with the correct permissions. Only administrators with the permission to approve at time of request will see the request.

Adding an Existing Student

Existing students can be added directly by email address from the ‘Manage Students’ page as long as the student hasn’t already joined a school.
8.29.2 Profiles

Each user account has a profile that represents who they are. Profiles can be described as a digital version of a resume. Profile content is used to easily search for students or mentors.

Building a Profile

Users can add education, work experience, projects, extra curriculars, courses, skills, organizations and external links to their profile. Building a profile is done on the user’s profile page.

Parsing Resume to Auto Build

An option for quickly building an initial profile is through the Resume Parser. The Resume Parser will parse an uploaded resume and automatically allow you to add the information to your profile.

8.29.3 Primary Educations

In order to determine what jobs, interviews, events, career fairs, articles, career plans and pins that a student qualifies for we look to their Primary Education. Every student has one Primary Education on their profile which is denoted with a label. Handshake will use the GPA’s, majors, minors, and colleges from this education for screening criteria.

8.29.4 Locking fields on students profile

Student profile fields may be locked which will disallow edits by students. This is helpful when you are importing data from your student information center and would like those fields to remain locked until the next student sync. Fields that can be locked are located in the school settings under Student Preferences. Locked fields may still be edited by staff.

8.29.5 Archived Users

Users are not deleted in Handshake; instead, they are archived. Archiving is a safer solution, and behaves the same as if they were deleted. Archived users are still searchable and viewable, but will be denoted as ‘Archived’.

8.29.6 Student Record

A basic student record is available from the user profile page. The student record shows appointment history with notes from each appointment.

8.30 Users vs. Contacts

Handshake has both ‘Users’ and ‘Contacts’. Although similar in that they both have common attributes such as name, email and contact information, they play two different roles in the system. Users are registered accounts with the ability to login and use the Handshake interface. Contacts are simply a record of information for CRM functionalities.
8.30.1 Why?

When creating different items in Handshake such as jobs, events, career fairs, career fair registrations, interviews and more it is necessary to have contact information for that item. By allowing you to use contacts for that information instead of users, it is much easier to add the information needed and does not require creating a login account for that person.

In addition, contacts can be used for a contact relationship system. For more details on how to use contacts, see the contacts page. By using contacts instead of users for contact relation management, the data is controlled by you and is not lost if the user, for example, decides to delete their account.

8.30.2 Inviting Contacts to be Users

Having contacts in Handshake for specific persons is great, but sometimes it is also desired to have them register as a user to take advantage of the Handshake interface. For more information on how to invite contacts to Handshake, see the contacts page and employers page.