olin-api Documentation

Release 0.1

Leon Lam, Dakota Nelson

May 02, 2017

Contents:

CHAPTER 1

Application Module

Contents

Introduction

The application component allows applications to register themselves, at which point they are issued an application token. The application token allows access to application-scoped resources.

Glossary

Application Token

Issued when a user registers an application. Requires verification of the application's contact email address before the token is marked valid. Allows for accessing application-level resources. Required in order to request an auth token.

Validation Token

Used for email validation. Sent in a link to the specified email when an auth token is requested, or to the application contact email when an application token is requested.

Endpoints

`GET /app/all`

List all currently registered applications

Status Codes

- 200 OK – Success

`POST /app/register`

Allows a user to register a client application with the Olin API. This will grant them an application token which can later be used to either access client-specific resources or request an authorization token that will grant the application access to the API on a user's behalf.

Users must agree to the Olin API Honor Code at https://github.com/DakotaNelson/olin-api/blob/master/HONOR-CODE.md

Parameters

- **`contact`** (*`str`*) – An email address to use as a designated contact for the application.
- **`name`** (*`str`*) – The name of the application.
- **`honorcode`** (*`bool`*) – A boolean value indicating whether or not the requester has read and agrees to the Olin API Honor Code.
- **`description`** (*`str`*) – A description of the application.
- **`homepage`** (*`str`*) – A URL pointing to the application's website.

Status Codes

- 200 OK – Success
- 400 Bad Request – A JSON request parameter is missing or incorrect, or an app with the name and contact email already exist.

`GET /app/validate/`(*token*)

Given a validation token (what is sent in an email to the token requester's email address), check that it is good, then mark the corresponding app as valid

Status Codes

- 200 OK – Success
- 400 Bad Request – The validation token provided is either invalid or expired.

CHAPTER 2

Authentication Module

Contents

Introduction

The authentication component allows applications to authenticate that a user does in fact own an email address that they claim. Once this flow is completed, an auth token is issued, which allows the application to access resources on that user's behalf.

Glossary

Application Token

Issued when a user registers an application. Requires verification of the application's contact email address before the token is marked valid. Allows for accessing application-level resources. Required in order to request an auth token.

Auth Token

Issued when a client application requests one. Scoped to a single email address, which is verified before the token is marked valid. Allows for requesting resources on behalf of the provided email address.

Validation Token

Used for email validation. Sent in a link to the specified email when an auth token is requested, or to the application contact email when an application token is requested.

Endpoints

`POST /auth`

Given an authentication token, returns a person profile OR a message stating the token is invalid

Parameters

- **`token`** (*`str`*) – The authentication token to check.

`POST /auth/token`

Returns an access token against an email address which will be valid once the specified email address is verified (by clicking a link in an email sent to it)

If called multiple times, will simply return the same auth token.

Parameters

- **`email`** (*`str`*) – Email address to validate and return an access token for.
- **`apptoken`** (*`str`*) – An application token verifying that the calling application has registered itself, provided a contact, agreed to the Olin API Honor code, etc.

`DELETE /auth/token`

Deletes an access token record, rendering the associated token invalid and allowing for re-issuing a token.

Parameters

- **`email`** (*`str`*) – Email for the token to delete.
- **`apptoken`** (*`str`*) – An application token verifying that the calling application has registered itself, provided a contact, agreed to the Olin API Honor code, etc.

`GET /auth/token/validate/`(*token*)

Given a validation token (what is sent in an email to the token requester's email address), check that it is good, then mark the corresponding token as valid

CHAPTER 3

People Module

PUT /people/

Edits one or more user objects. Search params identical to get(). The nature of the edit is based on the json component of the request.

POST /people/

If requests.json's fields do not match those defined in the Person model, this fails. See documentation for src.document_models.Person for more details.

GET /people/

Returns one or more user objects.

Parameters

- **email** (*str*) – User's email address
- **fName** (*str*) – First name of the user
- **lName** (*str*) – Last name of the user
- **comYearMIN** (*int*) – Minimum community year of the user (inclusive)
- **comYearMAX** (*int*) – Maximum community year of the user (inclusive)

CHAPTER 4

Introduction

Welcome to the Olin-API docs!

Olin-api seeks to provide programmatic access to a wide variety of Olin College digital resources. It is meant to provide the foundation for a variety of student-built applications to benefit the community.

The API is organized as a series of modules, each of which provides a unique service to those using it.

CHAPTER 5

Application Module

This module allows applications to register themselves with the API in order to access further API resources. It also allows clients to list applications which have been registered with the API.

CHAPTER 6

Authentication Module

This module provides authentication services to client applications. Specifically, it is able to verify whether or not any given user is in control of an email address they claim. This allows applications to verify that their users are, for example, in control of an `@students.olin.edu` email address and therefore allowed to access community-only resources.

CHAPTER 7

People Module

This module allows for CRUD operations on a database of Olin community members.

HTTP Routing Table