

Contacts REST API – Version 3.0

Table of Contents

Introduction.....	2
Requests.....	2
URL.....	2
API Key.....	2
Version.....	2
Content Type.....	3
CORS.....	3
Rate Limiting.....	3
Responses.....	3
Error Response Messages.....	3
Actions.....	4
Login.....	4
Resources.....	5
General.....	5
Contacts.....	7
Addresses.....	10
Assignments.....	11
Dates.....	12
Email Addresses.....	13
Groups.....	14
History Events.....	15
Lead Sources.....	16
Notes.....	17
Phone Numbers.....	18
Members.....	19
Offices.....	21
Error Codes.....	22
Change Control.....	23

Introduction

The Contacts REST API is a way for your application to interact with your contact data. It is a RESTful web service that you can use to query data and add, update or delete data. The API supports JSON based requests and responses.

Requests

All HTTP requests to the API include some common elements.

URL

When your application is granted access to the API you will be given a base URL to use when making requests to the API. Beginning with v3 of this API, this URL must be used with SSL. Any requests made to the API not using *https://* will be rejected with a “403 Forbidden” response.



Example:

`https://contacts.realestatedigital.com`

API Key

Every request to the API must include an API key in the http headers. One key will be provided to each customer after signing the Contact REST API Master Services Agreement. The name of the header is X-RED-ApiKey. If an API key is not provided with the request an HTTP status code of 401 Not Authorized will be returned.



Example:

`X-RED-ApiKey: 119DCFA925FE43A5BB0AFC67CC550B39`

Version

Every request to the API must include a version in the URL. This ensures that your application accesses a stable interface. The version must be included immediately after the host in the URL. If no version is provided in the URL a 404 Not Found HTTP status code will be returned in the response. When a new version is released the previous version will be supported for 90 calendar days. After 180 calendar days, RED reserves the right to decommission the prior version including deactivating and disabling all access to the URLs for the prior version.



Example:

`https://contacts.realestatedigital.com/v3/contacts/119DCFA925FE43A5BB0AFC67CC550B39`

Content Type

Any requests that POSTs or PUTs data to the API may format the data as JSON. The appropriate Content-Type header must be set, accepted values are application/json. If a Content-Type header is not provided, or is not one of the accepted values, a 400 Bad Request HTTP status code will be returned. A request may also specify the content type of the response body using the Accept header. The default response type is application/json.

CORS

Cross-origin resource sharing (CORS) is supported, for browser-based access to the API.

Rate Limiting

RED imposes rate limits on all APIs to ensure maximum performance for all its customers and their developers, and to protect the user experience on customer websites. Rate limits may be different per RED API and are noted in your signed application services agreement.

Responses

Responses to API requests use standard HTTP response codes, with an optional response body. Any error condition (HTTP status codes of 4xx or 5xx) will include an API response message. A successful action request (such as Login) (HTTP status code 200) will also include an API response message. A successful resource request will result in an HTTP status code of 200 OK along with the requested resource in the response body for GET requests, a status of 201 Created for POST requests along with the location of the newly created resource in the Location header, or a status of 204 No Content for successful PUT requests.

Error Response Messages

An error response message includes a JSON object in the response body. This object has two properties, “message” and “errors”. The “message” property gives a description of the general condition, and the “errors” property is an array of specific error messages. Each error message contains a code and a message. If the response is a success, the “errors” property will be empty.



Example:

```
{
  "message": "Invalid contact",
  "errors":[
    { code: 3001, message: "A Primary Email Address is required." },
    { code: 3002, message: "Email addresses must be of the format abc@xyz.tld" }
  ]
}
```

Note: See possible error codes at the bottom of this document.

Actions

Action requests are for API functionality that does not access a specific resource type.

Login

The Login action is to authenticate a user for subsequent requests to a restricted resource. A restricted resource is one that requires acting on behalf of a user. Examples would be getting a specific contact profile, or posting a new email address for a specific contact.

A Login request must be a POST, and include two parameters, “userLoginId” and “password”, both are required. If either parameter is missing a 400 Bad Request response will be returned. If a user is not found with the provided credentials a 401 Not Authorized response will be returned.

A successful Login response will include a cookie in the headers named **.contactsapiauth**. This cookie must be included in requests to restricted resources. It will also include a response body object with a “message” property that includes the location to the contact resource that corresponds to the login credentials posted.



Example:

HTTP Request:

```
Content-Type: application/json
X-RED-ApiKey: 119DCFA925FE43A5BB0AFC67CC550B39

{userLoginId:"username",password:"password"}
```

HTTP Response:

```
Set-Cookie: .contactsapiauth=AE2016641CD8202; expires=Thu, 02-Jul-2013 18:15:05 GMT; path=/

{"message":"https://contacts.realestatedigital.com/v3/contacts"}
```

Resources

Resource requests are for modifying and accessing types of objects within the rDesk CRM application. HTTP verbs are used to determine what type of action to take with the request. See each resource for integration details which are different than below. Model definitions are based on the emerging RESO standard.

General

/v3/[resource name] (GET)

/v3/[resource name]/{resourceKey}/[sub-resource name]{ (GET)

The GET request to the resource with no key supplied will result in an array containing all resources available to the authenticated user. If no resources are found an empty array will be returned.

/v3/[resource name]/{resourceKey} (GET)

/v3/[resource name]/{resourceKey}/[sub-resource name]{sub-resourceKey}/[(GET)

The GET request to the resource with a valid key will return one resource object if the resource being requested exists, with a 200 OK response. If no resource corresponding to the provided key is found, a 404 Not Found response will be returned.

/v3/[resource name] (POST)

/v3/[resource name]/{resourceKey}/[sub-resource name] (POST)

The POST request to the resource will create a new instance of the resource based on model provided in the request body. When the resource is created successfully a 201 Created response will be returned. This response will include the URL of the new resource in the Location header. If the resource already exists, a 409 Conflict response will be returned. If the model fails validation, a 400 Bad Request response will be returned.

/v3/[resource name]/{resourceKey} (PUT)

/v3/[resource name]/{resourceKey}/[sub-resource name]{sub-resourceKey}/ (PUT)

The PUT request to the resource will update an existing resource. When the resource is successfully updated a 204 No Content response will be returned. If the model fails validation, a 400 Bad Request response will be returned. If no resource corresponding to the provided key is found, a 404 Not Found response will be returned.


A recommended approach is to first perform a GET operation with the key, update the fields that your application supports, then PUT the updated model.


/v3/[resource name]/{resourceKey} (DELETE)

/v3/[resource name]/{resourceKey}/[sub-resource name]{sub-resourceKey} (DELETE)

The DELETE request to the resource with a valid key will delete one resource object if the resource specific exists, with a 204 No Content response. If no resource corresponding to the provided key is found, a 404 Not Found response will be returned.

Verb support by resource

 Indicates not implemented yet

Resource	GET	POST	PUT	DELETE
Contacts	✓	✓	✓	
Contacts/Addresses	✓	✓	✓	
Contacts/Assignments	✓		N/A	N/A
Contacts/Dates	✓	N/A	N/A	N/A
Contacts/EmailAddresses	✓	✓	✓	
Contacts/Groups	✓	✓	N/A	
Contacts/HistoryEvents	✓	✓	N/A	N/A
Contacts/LeadSources	✓	✓	N/A	N/A
Contacts/Notes	✓	✓	N/A	
Contact/PhoneNumbers	✓	✓	✓	
Members	✓	N/A	N/A	N/A
Offices	✓	N/A	N/A	N/A

If the resource does not support the verb, a 405 Method Not Allowed response will be returned.

GET/DELETE required fields

Accessing a specific resource via a GET or DELETE verb only requires a key which must be supplied in the URL.

POST/PUT required fields

✓ indicates a field is required for this verb. If one of these fields is missing for a given verb, the model will not pass validation.

✗ indicates a field is not allowed or available for this verb. If one of these fields is present, it will be ignored.

PUT operations

A PUT operation represents the new state of the resource. If a field is missing or blank, it will overwrite any pre-existing value.

Contacts

The Contacts resource allows modifying and accessing contacts within the rDesk CRM application. A contact object may be retrieved, a new contact may be added, or an existing contact profile may be updated. Deleting contacts is not allowed through the API. Getting a contact or updating a contact requires that the request be made in the context of an authenticated contact, by providing the **.contactsapiauth** cookie retrieved from Login action.

Contacts Parameter Definition

Parameter Name	Type	Values	Description
createdAfter	DateTime	UTC	Restricts results to resources created after this datetime stamp
embed	String	Addresses, Assignment, Assignments, EmailAddresses, LeadSources, PhoneNumbers, Notes	Allows sub-resource to be included in the Contact response. Pass as comma-delimited. Assignment can be used to retrieve only the current assignment.

Parameter Name	Type	Values	Description
modifiedAfter	DateTime	UTC	Restricts results to resources modified after this datetime stamp

The Contact model requires a (First Name, Last Name) AND (an Email Address OR a Phone Number). On POST operations, the following sub-resources need to be embedded as an array.

- Email Address and/or Phone Number
- Assignments
- Lead Sources

PUT operations should not include Email Addresses and Phone Numbers arrays. This should be done via the sub-resources.

Contact Model Definition

Field Name	Type	POST	PUT	Description
contactKey	GUID	✗	✓	Passed in URL
acceptedByMember	Boolean	✓	✓	Only applies when ownerType = Member. Previously accepted leads cannot be unaccepted.
company	String			
firstName	String	✓	✓	
fullName	String	✗	✗	Prefix + F + M + L + Suffix
jobTitle	String			
isBuyer	Boolean	✓	✓	
isSeller	Boolean	✓	✓	

Field Name	Type	POST	PUT	Description
lastName	String	✓	✓	
middleName	String			
namePrefix	String			
nameSuffix	String			
nickname	String			
originatingSystemContactKey	String	✗	✗	Contact's RED ID
originatingSystemName	String	✗	✗	"Real Estate Digital"
preferredContactMethod	String			Missing field indicates "No Preference"
preferredPhoneType	String			Missing field indicates "No Preference"
preferredTime	String			Missing field indicates "No Preference"
timestampEntered	DateTime	✗	✗	Creation date (UTC)
timestampModified	DateTime	✗	✗	Last modified date (UTC)

Pagination

Pagination is currently supported for all GET requests on the Contacts resource. Specific pagination options can be supplied in the query string by adding: page, pageSize, and sort parameters.

If no pagination options are supplied, the follow defaults will be used: page=1, pageSize=1000, sort=timestampEntered, i.e. a GET request to the /v3/Contacts endpoint without any supplied parameters will behave as if the GET request was: /v3/Contacts?page=1&pageSize=1000&sort= timestampEntered.

The available sort options are firstName, lastName, timestampEntered, and timestampModified. The default sort direction is Ascending, however the sort parameter can be appended with “ asc” or “ desc” to sort as needed. When results are split across multiple pages, a Link header will be included in the response, with URLs for the first/previous/next/last pages.

Addresses

The Addresses resource allows modifying and accessing addresses for a given contact within the rDesk CRM application. An address object may be retrieved, a new address may be added, and an existing address may be updated or deleted.

Addresses Parameter Definition

Parameter Name	Type	Values	Description
createdAfter	DateTime	UTC	Restricts results to resources created after this datetime stamp
modifiedAfter	DateTime	UTC	Restricts results to resources modified after this datetime stamp

Address Model Definition

Field Name	Type	POST	PUT	Description
addressKey	GUID	✗	✓	Passed in URL
address1	String	✓	✓	
address2	String			
addressType	String	✓	✓	
city	String	✓	✓	
country	String			
postalCode	String	✓	✓	
stateOrProvince	String	✓	✓	
timestampEntered	DateTime	✗	✗	Creation date (UTC)
timestampModified	DateTime	✗	✗	Last modified date (UTC)

/v3/contacts/{id}/addresses (POST)

Multiple address of the same type are acceptable. Therefore, a 409 Conflict response will not be returned.

Assignments

The Assignments resource allows modifying and accessing the current assignment for a given contact within the rDesk CRM application. Assignment objects may be retrieved and a new assignment may be added.

Assignment Parameter Definition

Parameter Name	Type	Values	Description
createdAfter	DateTime	UTC	Restricts results to resources created after this datetime stamp

Assignment Model Definition

Field Name	Type	POST	Description
assignmentKey	GUID	✘	Passed in URL
ownerKey	String	✔	
ownerType	String	✔	Member or Office
autoAccept	Boolean	✔	Indicates whether an Acceptance is required by Assignee
timestampEntered	DateTime	✘	Creation date (UTC)

Dates

The Dates sub-resource allows accessing event dates (e.g. birth date, home anniversary) for a given contact within the rDesk CRM application. A date object may be retrieved.

Date Model Definition

Field Name	Type	POST	PUT	Description
dateKey	GUID	N/A	N/A	
dateType	String	N/A	N/A	Date type (e.g. 'BirthDate', 'HomeAnniversary')
date	DateTime	N/A	N/A	Contact event date (UTC).
timestampEntered	DateTime	N/A	N/A	Creation date (UTC)
timestampModified	DateTime	N/A	N/A	Last modified date (UTC)

/v3/contacts/{id}/dates (GET)

This usage returns a list of dates for this contact.

Email Addresses

The Email Addresses resource allows modifying and accessing email addresses for a given contact within the rDesk CRM application. An email address object may be retrieved, a new email address may be added, and an existing email address may be updated or deleted.

Email Addresses Parameter Definition

Parameter Name	Type	Values	Description
createdAfter	DateTime	UTC	Restricts results to resources created after this datetime stamp
modifiedAfter	DateTime	UTC	Restricts results to resources modified after this datetime stamp

Email Address Model Definition

Field Name	Type	POST	PUT	Description
emailAddressKey	GUID	✗	✓	Passed in URL
emailAddress	String	✓	✓	
emailType	String	✓	✓	Primary, Secondary, Alternate
timestampEntered	DateTime	✗	✗	Creation date (UTC)
timestampModified	DateTime	✗	✗	Last modified date (UTC)
usedForAPM	Boolean			RED reviewing default setting

/v3/contacts/{id}/emailaddresses (POST)

If a resource of the same type already exists, a 409 Conflict response will be returned.

Groups

The Group resource allows attaching a Group to a given Contact. PUTs are not allowed. To change Groups for a Contact, it is expected that POST and DELETE verbs will be used appropriately to set the Groups for a given Contact as desired.

Group Parameter Definition

Parameter Name	Type	Values	Description
createdAfter	DateTime	UTC	Restricts results to resources created after this datetime stamp

Group Model Definition

Field Name	Type	POST	PUT	Description
groupKey	GUID	✘	N/A	Passed in URL
name	String	✔	N/A	
overridesGroupKey	GUID	✘	N/A	The ID of the system group that this group overrides, if any.
timestampEntered	DateTime	✘	N/A	Creation date (UTC)
timestampModified	DateTime	✘	N/A	Last modified date (UTC)

In the CRM Web Application, users may rename groups, including system-level groups. When this happens, the groupKey reported by the API will change to the new ID. The overridesGroupKey field will contain the original system groupKey to facilitate tracking this change.

The groups sub-resource can be used in conjunction with either the contacts or members resources:

/v3/contacts/{id}/groups (GET)

This usage returns a list of groups of which this contact is a member.

/v3/members/{id}/groups (GET)

This usage returns a list of all groups available to the member.

History Events

The History Event resource allows adding a history event record for a given contact within the rDesk CRM application. All history events will have a source of External.

History Event Model Definition

Field Name	Type	POST	PUT	Description
historyEventKey	GUID	N/A	N/A	Passed in URL
details	String	✓	N/A	
timestampEntered	DateTime	✗	N/A	Creation date (UTC)
type	String	✓	N/A	RED reviewing allowed values
initiatorType	String	✓	N/A	RED reviewing allowed values
action	String	✓	N/A	RED reviewing allowed values

Lead Sources

The Lead Sources resource allows accessing the lead sources for a given contact within the rDesk CRM application. A lead source object may be retrieved and a new lead source may be added.

Lead Source Parameter Definition

Parameter Name	Type	Values	Description
createdAfter	DateTime	UTC	Restricts results to resources created after this datetime stamp

Lead Source Model Definition

Field Name	Type	POST	Description
leadSourceKey	GUID	✘	Passed in URL
leadSource	String	✔	Can be used to trigger Updated Leads routing options
subLeadSource	String	✔	Can be used to trigger Updated Leads routing options
isLead	Boolean		If ever isLead = True, the lead will always be visible to assigned Owner and lead admins. Once set to True, it cannot be changed. If isLead is always False, the lead will only be visible to the assigned Member.
timestampEntered	DateTime	✘	Creation date (UTC)

Notes

The Notes resource allows accessing the notes for a given contact within the rDesk CRM application. A note object may be retrieved and a new note may be added.

Note Parameter Definition

Parameter Name	Type	Values	Description
createdAfter	DateTime	UTC	Restricts results to resources created after this datetime stamp

Note Model Definition

Field Name	Type	POST	Description
noteKey	GUID	✗	Passed in URL
note	String	✓	
addedByMember	String	✓	
timestampEntered	DateTime	✗	Creation date (UTC)

Phone Numbers

The Phone Numbers resource allows modifying and accessing phone numbers for a given contact within the rDesk CRM application. A phone number object may be retrieved, a new phone number may be added, and an existing phone number may be updated or deleted.

The rDesk CRM UI only displays phones of type Home, Work, Mobile, and Fax.

Constellation1 reviewing use-case where multiple phone numbers have the same type.

Phone Numbers Parameter Definition

Parameter Name	Type	Values	Description
createdAfter	DateTime	UTC	Restricts results to resources created after this datetime stamp
modifiedAfter	DateTime	UTC	Restricts results to resources modified after this datetime stamp

Phone Number Model Definition

Field Name	Type	POST	PUT	Description
phoneNumberKey	GUID	✗	✓	Passed in URL
phoneNumber	String	✓	✓	
phoneNumberType	String	✓	✓	
timestampEntered	DateTime	✗	✗	Creation date (UTC)
timestampModified	DateTime	✗	✗	Last modified date (UTC)

/v3/contacts/{contactKey}/phonenumbers (POST)

If a resource of the same type already exists, a 409 Conflict response will be returned.

Members

The Members resource allows accessing a list of active users of the rDesk CRM application. A Member object may be retrieved. Adding, updating, and deleting agents is not allowed through the API.

Member Definition

Field Name	Type	Description
memberKey	GUID	Member's RED ID passed in URL
memberAddress1	String	
memberAddress2	String	
memberCity	String	
memberCountry	String	
memberEmail	String	
memberFax	String	
memberFirstName	String	
memberFullName	String	
memberLastName	String	
memberMobilePhone	String	
memberOtherPhone[Type]Number	String	memberOtherPhoneType values will replace [Type]
memberOtherPhoneType	String	
memberPager	String	
memberPostalCode	String	
memberPreferredPhone	String	

Field Name	Type	Description
memberStateOrProvince	String	
officeKey	GUID	Member's Office's RED ID
originatingSystemMemberKey	GUID	ID of Member in Originating System
originatingSystemName	String	Company Name of Source of Roster Data
timestampEntered	DateTime	
timestampModified	DateTime	

Contacts sub-resource of Members (with Pagination support)

/v3/members/{id}/contacts (GET)

The Members resource allows accessing a list of contacts that are currently assigned to a specific member. This Sub-Resource supports pagination in the same manner that is supported on the Contacts resource, i.e. Specific pagination options can be supplied in the query string by adding: page, pageSize, and sort parameters. If no pagination options are supplied, the follow defaults will be used: page=1, pageSize=1000, sort=timestampEntered. For additional information related to pagination, see the section above dedicated to this topic.

Attention v2 Users: If you are currently using this sub-resource to download all contacts for a given agent, you will have to make changes to cycle through all pages of Contacts for those Agents with more than 1000 contacts.

Offices

The Offices resource allows accessing a list of offices within the rDesk CRM application. An Office object may be retrieved. Adding, updating, and deleting offices is not allowed through the API.

Office Model Definition

Field Name	Type	Description
officeKey	GUID	Office's RED ID passed in URL
mainOfficeKey	GUID	Company's RED ID (top-level organization)
officeAddress1	String	
officeAddress2	String	
officeBranchType	String	
officeBrokerKey	GUID	In franchise model, Broker's RED ID. In broker model, same as mainOfficeKey
officeCity	String	
officeEmail	String	
officeFax	String	
officeName	String	
officePhone	String	
officePostalCode	String	
officeStateOrProvince	String	
originatingSystemName	String	Company Name of Source of Roster Data
originatingSystemOfficeKey	GUID	ID of Office in Originating System
timestampEntered	DateTime	

Field Name	Type	Description
timestampModified	DateTime	

Error Codes

Error Response Code	Description	HTTP Status Code
500	System Error	500
<i>1000 series</i>	<i>Application authentication errors</i>	
TBD	Invalid login request	400
TBD	API key invalid	401
TBD	API key disabled	401
TBD	API key not provided	401
<i>2000 series</i>	<i>User authentication errors</i>	
<i>3000 series</i>	<i>Resource errors</i>	
TBD	Resource failed validation	400
TBD	Resource does not exist	404
TBD	Method not allowed for resource	405
TBD	Resource already exists	409
	Contact does not contain	

Change Control

Date	Section/Paragraph	Change
2/27/2015	All	BETA version
3/6/2015	All	Field definitions updated
3/13/2015	All	Required fields updated
3/18/2015	All	Added Assignments & Lead Sources resources, Rate Limiting
3/27/2015	All	Added Notes resource
10/30/2015	All	Added GET support for HistoryEvents
1/17/2016	All	Modified Samples to use v2
2/1/2016	All	Added Groups Resource
4/3/2017	All	Added autoAccept to Assignments Resource
4/3/2017	All	Updated description to match changes made to originatingSystemOfficeKey and originatingSystemMemberKey
5/2/2017	Pagination	Inserted paragraph to describe pagination options and default pagination for Contacts resource
6/8/2017	All	Updated to document to reference v3, as well as the requirement for HTTPS and pagination on the members/{id}/contacts sub-resource