

# Underwriter Microservice API

## Overview

This API provides operations to either accept or deny an application for an account as well as Applicant and Member creation and management.

## Paths

### **GET** /applicants/{id}

Get Applicant by ID

#### Parameters

Type	Name	Description	Schema
path	id <i>required</i>		integer (int64)

#### Responses

Code	Description	Links
404	Applicant does not exist.  <i>Content</i> /	No Links
200	Applicant was found.  <i>Content</i> /	No Links

### **PUT** /applicants/{id}

Update an Applicant by ID

#### Parameters

Type	Name	Description	Schema
path	id <i>required</i>		integer (int64)

#### Responses

Code	Description	Links
400	New values were not valid.	No Links

Code	Description	Links
404	Applicant to update was not found.	No Links
204	Applicant was successfully updated.	No Links

## **DELETE** /applicants/{id}

Delete an Applicant by ID

### Parameters

Type	Name	Description	Schema
path	id <i>required</i>		integer (int64)

### Responses

Code	Description	Links
204	Applicant was successfully deleted.	No Links
404	Applicant to delete was not found.	No Links

## **GET** /applications

Get a paginated response of all applicants

### Parameters

Type	Name	Description	Schema
query	search <i>optional</i>		string
query	pageable <i>required</i>		<a href="#">Pageable</a>

### Responses

Code	Description	Links
200	Successfully retrieved paginated applications. The content array may be empty which means no applications exist.  <i>Content</i> /	No Links

## **POST** /applications

Apply for a membership.

### Responses

Code	Description	Links
201	Application was successfully created and either approved, denied, or set to pending.  <i>Content</i> /	No Links
400	Application could not be processed for some reason.  <i>Content</i> /	No Links
502	The application notification email was not sent.  <i>Content</i> /	No Links
409	There was a conflict with creating one or more of the applicants. There is a conflict with the specified unique columns.  <i>Content</i> /	No Links
404	Creating an application with existing applicants and one or more of the existing applicants do not exist.  <i>Content</i> /	No Links

## GET /applicants

Get all Applicants (Paginated)

### Parameters

Type	Name	Description	Schema
query	search <i>optional</i>		string
query	pageable <i>required</i>		<a href="#">Pageable</a>

### Responses

Code	Description	Links
200	Retrieve a populated or empty page of applicants.  <i>Content</i> /	No Links

## POST /applicants

Manually create an Applicant

### Responses

Code	Description	Links
409	Applicant data conflicts with other applicant data.  <i>Content</i> /	No Links
404	Applicant data is not valid. (Bad request)  <i>Content</i> /	No Links
201	Applicant successfully created.  <i>Content</i> /	No Links

## GET /health

Health check endpoint

### Responses

Code	Description	Links
200	Service is healthy.	No Links
404	The service is probably not running.	No Links

## GET /applications/{id}

Get an application by ID

### Parameters

Type	Name	Description	Schema
path	id <i>required</i>		integer (int64)

### Responses

Code	Description	Links
404	Application does not exist.  <i>Content</i> /	No Links

Code	Description	Links
200	Application was found.  <i>Content</i> <i>/</i>	No Links

# Components

## Schemas

### UpdateApplicant

#### Properties

Name	Description	Schema
firstName <i>optional</i>		string
middleName <i>optional</i>		string
lastName <i>optional</i>		string
dateOfBirth <i>optional</i>		string (date)
gender <i>optional</i>		enum (MALE,FEMALE,OTHER,UNSPECIFIED)
email <i>optional</i>		string
phone <i>optional</i>		string
socialSecurity <i>optional</i>		string
driversLicense <i>optional</i>		string
income <i>optional</i>	<b>Minimum:</b> 0	integer (int32)
address <i>optional</i>		string
city <i>optional</i>		string
state <i>optional</i>		string
zipcode <i>optional</i>		string

Name	Description	Schema
mailingAddress <i>optional</i>		string
mailingCity <i>optional</i>		string
mailingState <i>optional</i>		string
mailingZipcode <i>optional</i>		string

## ApplyRequest

### Properties

Name	Description	Schema
applicationType <i>required</i>		enum (CHECKING,SAVINGS,CHECKING_AND_SAVINGS,CREDIT_CARD,LOAN)
noNewApplications <i>optional</i>		boolean
applicantIds <i>optional</i>	<i>unique items</i>	< integer > array
applicants <i>optional</i>	<i>unique items</i> <b>Maximum Items:</b> 3 <b>Minimum Items:</b> 0	< <a href="#">CreateApplicant</a> > array
applicationAmount <i>optional</i>	<b>Minimum:</b> 0	integer (int32)
cardOfferId <i>optional</i>		integer (int32)
depositAccountNumber <i>optional</i>		string

## CreateApplicant

### Properties

Name	Description	Schema
firstName <i>required</i>		string

Name	Description	Schema
middleName <i>optional</i>		string
lastName <i>required</i>		string
dateOfBirth <i>required</i>		string (date)
gender <i>required</i>		enum (MALE,FEMALE,OTHER,UNSPECIFIED)
email <i>required</i>		string
phone <i>required</i>		string
socialSecurity <i>required</i>		string
driversLicense <i>required</i>		string
income <i>required</i>	<b>Minimum: 0</b>	integer (int32)
address <i>required</i>		string
city <i>required</i>		string
state <i>required</i>		string
zipcode <i>required</i>		string
mailingAddresses <i>required</i>		string
mailingCity <i>required</i>		string
mailingState <i>required</i>		string
mailingZipcode <i>required</i>		string

## ApplicantResponse

### Properties

Name	Description	Schema
id <i>optional</i>		integer (int64)

Name	Description	Schema
firstName <i>optional</i>		string
middleName <i>optional</i>		string
lastName <i>optional</i>		string
dateOfBirth <i>optional</i>		string (date)
gender <i>optional</i>		string
email <i>optional</i>		string
phone <i>optional</i>		string
socialSecurity <i>optional</i>		string
driversLicense <i>optional</i>		string
income <i>optional</i>		integer (int32)
address <i>optional</i>		string
city <i>optional</i>		string
state <i>optional</i>		string
zipcode <i>optional</i>		string
mailingAddress <i>optional</i>		string
mailingCity <i>optional</i>		string
mailingState <i>optional</i>		string
mailingZipcode <i>optional</i>		string
lastModifiedAt <i>optional</i>		string (date-time)
createdAt <i>optional</i>		string (date-time)



## ApplyAccountResponse

### Properties

Name	Description	Schema
accountNumber <i>optional</i>		string
accountType <i>optional</i>		string

## ApplyMemberResponse

### Properties

Name	Description	Schema
membershipId <i>optional</i>		string
name <i>optional</i>		string

## ApplyResponse

### Properties

Name	Description	Schema
id <i>optional</i>		integer (int64)
applicationType <i>optional</i>		enum (CHECKING,SAVINGS,CHECKING_AND_SAVINGS,CREDIT_CARD,LOAN)
loanType <i>optional</i>		enum (PERSONAL,AUTO,HOME,BUSINESS,SECURE)
applyAmount <i>optional</i>		integer (int32)
applicants <i>optional</i>		< <a href="#">ApplicantResponse</a> > array
status <i>optional</i>		enum (APPROVED,DENIED,PENDING)

Name	Description	Schema
reasons <i>optional</i>		< string > array
accountsCreated <i>optional</i>		boolean
createdAccounts <i>optional</i>	<i>unique items</i>	< <a href="#">ApplyAccountResponse</a> > array
membersCreated <i>optional</i>		boolean
createdMembers <i>optional</i>		< <a href="#">ApplyMemberResponse</a> > array

## Pageable

### Properties

Name	Description	Schema
offset <i>optional</i>		integer (int64)
sort <i>optional</i>		<a href="#">Sort</a>
unpaged <i>optional</i>		boolean
paged <i>optional</i>		boolean
pageNumber <i>optional</i>		integer (int32)
pageSize <i>optional</i>		integer (int32)

## Sort

### Properties

Name	Description	Schema
unsorted <i>optional</i>		boolean
sorted <i>optional</i>		boolean
empty <i>optional</i>		boolean

## ApplicationResponse

### Properties

Name	Description	Schema
id <i>optional</i>		integer (int64)
applicationType <i>optional</i>		string
applicationStatus <i>optional</i>		string
primaryApplicant <i>optional</i>		<a href="#">ApplicantResponse</a>
applicants <i>optional</i>	<i>unique items</i>	< <a href="#">ApplicantResponse</a> > array

## PageApplicationResponse

### Properties

Name	Description	Schema
totalPages <i>optional</i>		integer (int32)
totalElements <i>optional</i>		integer (int64)
size <i>optional</i>		integer (int32)
content <i>optional</i>		< <a href="#">ApplicationResponse</a> > array
number <i>optional</i>		integer (int32)
sort <i>optional</i>		<a href="#">Sort</a>
first <i>optional</i>		boolean
last <i>optional</i>		boolean
numberOfElements <i>optional</i>		integer (int32)
pageable <i>optional</i>		<a href="#">Pageable</a>
empty <i>optional</i>		boolean

## PageApplicantResponse

### Properties

Name	Description	Schema
totalPages <i>optional</i>		integer (int32)
totalElements <i>optional</i>		integer (int64)
size <i>optional</i>		integer (int32)
content <i>optional</i>		< <a href="#">ApplicantResponse</a> > array
number <i>optional</i>		integer (int32)
sort <i>optional</i>		<a href="#">Sort</a>
first <i>optional</i>		boolean
last <i>optional</i>		boolean
numberOfElements <i>optional</i>		integer (int32)
pageable <i>optional</i>		<a href="#">Pageable</a>
empty <i>optional</i>		boolean