# COD Services API

# Bulk P-Code

## Contacts

Tom Haythornthwaite, United Nations, Office for the Coordination of Humanitarian Affairs, haythornthwaite@un.org

Anita Russo, University of Georgia, Carl Vinson Institute of Government, Information Technology

Outreach Services, [arusso@uga.edu](mailto:arusso@uga.edu)

## History

First draft: 16 June 2021

Contents

[COD Services API 1](#_Toc107570234)

[Bulk P-Code 1](#_Toc107570235)

[Contacts 1](#_Toc107570236)

[History 1](#_Toc107570237)

[Summary 2](#_Toc107570238)

[Instructions 2](#_Toc107570239)

[Examples 4](#_Toc107570240)

[Assumptions 4](#_Toc107570241)

[Appendix 6](#_Toc107570242)

[Credits 6](#_Toc107570243)

## Summary

This document describes the form to update an Excel spreadsheet with Common Operational Dataset (COD) administrative boundary theme P-Code and names values given the spreadsheet’s latitude and longitude columns. The feature is available through the COD Services API. The COD Services API is meant to be updated frequently to reflect the current state COD availability and service deployments through the ITOS vetting processes. Therefore, the output results are based on this routine maintenance. To understand the time and other aspects of the output, the API provides the details through the request as shown in the help documentation here: <https://apps.itos.uga.edu/CODV2API/Help/Api/GET-api-v1-themes-cod-ab-locations-pCode> with an example here: https://apps.itos.uga.edu/CODV2API/api/v1/themes/cod-ab/locations/moz. Please read on for more information on how to use the feature and examples.

Guest or elevated access is required while the feature is in beta test and a login page ensures that only registered users may access the form to provide the spreadsheet and obtain results. To have an account created for you, contact either contact listed in the contacts section of this document.

## Instructions

1. Visit the website here: <https://apps.itos.uga.edu/CODV2API>

Graphical user interface, application

Description automatically generated

1. Click the login option circled in red to log in.

The resulting page allows for credential entry. Guest users may also access the feature, so click on the “register” link to register as guest or follow the steps below if you already have an account in the system.

Graphical user interface, text, application, email

Description automatically generated

Figure Login screen for the COD Services API

1. Ensure the disclaimer checkbox is checked. Enter User name and Password in the form and click Log in. The means to reset user name and password are available in the links below the form, if these are not readily available.
2. Once Logged in, the BulkPCoder link is shown on the top menu of the site and also can be accessed directly from here: <https://apps.itos.uga.edu/CODV2API/CODInputForm/BulkPCoder>
3. The means to provide the spreadsheet is through the form.

Graphical user interface, text, application, chat or text message

Description automatically generated

1. Click the Choose File button and use the dialog to navigate to a valid spreadsheet to process. An example of an input spreadsheet is shown in the Examples section listed later in this document.
2. Note the filename of the spreadsheet will be shown in the form after selecting the file and dismissing the dialog.
3. Click the Get P-codes button.
4. The browser in use state should show where files containing more than 1000 records to process is provided. For less records, the download may occur momentarily. The button shows as highlighted in this case.

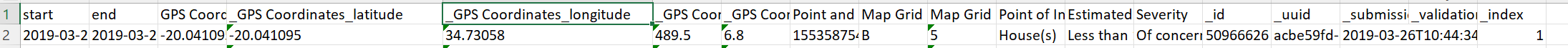
Graphical user interface, text, application, chat or text message

Description automatically generated

1. The output will be downloaded, and you should see your browser indicate the file is available in downloads on your computer.

## Examples

Input spreadsheet: The requirements for a valid input spreadsheet is that there should be two or more columns. Two columns should contain geographic coordinates for latitude and longitude with the latitude and longitude terms provided as at least part of the name of the column in the first row.



The above image shows the spreadsheet contents for a valid spreadsheet and the geographic coordinates are in “GPS\_Coordinates\_latitude” and “GPS\_Coordinates\_longitude” columns with the term “latitude” and “longitude” as part of the names. The coordinates are expected in decimal degrees geographic format. If coordinates are in pattern of data other than numeric decimal degrees for example:

A picture containing application

Description automatically generated

The input sheet may still be processed with output returned for those records that are valid and null reported for those records that cannot be matched to a P-Coded location for example:

Table

Description automatically generated

A file named pcoded is the result. An example is shown below. No changes are made to the contents of the original file except the following:

1. A new row to indicate new column names are added with the new column names appended to the end of the pre-existing columns of the original spreadsheet.
2. The new columns and data pertinent to the latitude and longitude are also appended as rows in these new columns.

Output results:

Table

Description automatically generated

## Assumptions

The input file meets requirements. If not, a message appears:



If data cannot be matched. Null values are given. So, there are cases where a coordinate that is in decimal degrees falls outside the administrative boundary, for example:

Map

Description automatically generated

A good practice may be to check the services for the COD by loading the service in a map or application to determine the coincident service data for the coordinate.

There are no checks to validate if the coordinates are reversed or missing decimal points or negative signs. Whatever data is present is used to query the service. It is assumed the user has validated the spreadsheet and is confident of the coordinate values present.

A number of tools may be used to check the results. Google maps may be a start where entering the comma separated coordinates may give insight as to the location:

Map

Description automatically generated

## Appendix

#### Resources

The COD Services API

The GitHub Repository for the COD Services API:

<https://github.com/UGA-ITOSHumanitarianGIS/CODV2API/>

The COD Services endpoints available through the HDX site: data.humdata.org

## Credits

|  |  |
| --- | --- |
| A picture containing plate  Description automatically generated | http://alpha.itos.uga.edu/USAID.MapCODLocationsPreviewer/Content/Images/usaid.png |