# Gravity Forms Address Validation with Google Maps API

Author: Devansh Mishra  
Version: 1.0  
Last Updated: October 2025

## Overview

This script enhances Gravity Forms by validating user-entered addresses against Google Maps Geocoding API. It ensures accurate and standardized address entries before form submission, improving data quality and preventing delivery or service issues caused by invalid addresses.

## Features

- Validates Street, City, State, ZIP, and Country using Google Maps API.  
- Supports multiple forms (IDs: 1, 2, 3).  
- Provides user-friendly, field-specific error messages.  
- Includes fuzzy matching for city names (≥85% similarity).  
- Handles both client and server-side validation.  
- Built-in normalization for common city abbreviations (e.g., St → Saint).

## File Placement

Place this code in your theme’s functions.php file or in a custom plugin (recommended for maintainability).

## Code Explanation

### 1. Hook Registration

The filter `gform\_validation` is used to hook into Gravity Forms’ server-side validation process. This allows the script to intercept form submissions and validate addresses before final submission.

add\_filter('gform\_validation', 'validate\_address\_server\_side');

### 2. Main Validation Function

The main function `validate\_address\_server\_side()` processes the validation. It only runs for specific form IDs (1, 2, and 3). For each form, it maps the field descriptions to their corresponding inputs and constructs a full address string to validate with Google Maps API.

function validate\_address\_server\_side($validation\_result) {  
 // Runs when form is submitted, checks address validity using Google API  
}

### 3. API Integration

The script retrieves the Google Maps API key from the WordPress options table (via `paul\_davis\_custom\_settings`). It then calls the Google Geocoding API with the user's address to check validity.

$url = 'https://maps.googleapis.com/maps/api/geocode/json?address=' . urlencode($full\_address) . '&key=' . $api\_key;  
$response = wp\_remote\_get($url);

### 4. Response Validation

If the API response status is not 'OK', the script marks the form as invalid and displays an error message. Otherwise, it compares the user’s input with the verified address components returned by Google.

### 5. Field Matching Logic

Each part of the address (street, city, state, ZIP, country) is compared with Google's data. For cities, it uses a fuzzy match threshold of 85% to tolerate small spelling differences. For the U.S., form ID 3 is automatically assumed valid for 'United States'.

if (in\_array('locality', $component['types'])) {  
 $google\_city = normalize\_city\_name($component['long\_name']);  
 $user\_city = normalize\_city\_name($city);  
 similar\_text($google\_city, $user\_city, $percent);  
 if ($percent >= 85) { $matched['City']['valid'] = true; }  
}

### 6. Helper Functions

Two helper functions improve maintainability and clarity:

function apply\_address\_error($validation\_result, &$form, $message, $type = "Street Address") {  
 // Flags a field as invalid and displays a custom validation message.  
}

function normalize\_city\_name($name) {  
 // Standardizes city names for accurate comparison by removing punctuation and expanding abbreviations.  
}

## Developer Notes

- Ensure your Google Maps API key has Geocoding API enabled.  
- The script expects field descriptions to include the keywords: 'Street Address', 'City', 'State / Province / Region', 'ZIP / Postal Code', and 'Country'.  
- To add more forms, include their IDs in the array: `[1, 2, 3]`.  
- Adjust the fuzzy match threshold in `similar\_text()` if you want stricter or looser validation.

## Testing Instructions

1. Enable WP\_DEBUG and check the debug log for API responses.  
2. Submit a form with a valid address — it should pass.  
3. Submit a form with an incorrect or incomplete address — it should show field-specific validation errors.  
4. Disable or remove the API key to confirm proper error handling ('Missing Google Maps API key.').