JavaScript

```
/**
* This Google Apps Script acts as a backend to receive data from the Passion-to-Profession Finder web app
* and write it to a Google Sheet.
*
* Instructions:
* 1. Go to Google Sheets (sheets.google.com) and create a new blank spreadsheet.
* Name it something like "Passion Finder Submissions".
* 2. In this new spreadsheet, go to Extensions > Apps Script. This will open a new Apps Script project.
* 3. Replace any existing code in the Apps Script editor with the code below.
* 4. IMPORTANT:
* - Update 'YOUR_SPREADSHEET_ID_HERE' with the actual ID of your Google Sheet.
* The ID is found in the Google Sheet URL:
* https://docs.google.com/spreadsheets/d/YOUR_SPREADSHEET_ID_HERE/edit
* - Update 'Submissions' if you want to use a different sheet name (default is 'Sheet1').
* 5. Save the Apps Script project (File > Save project). Give it a name like "PassionFinderBackend".
* 6. Deploy the script as a web app:
* - Click "Deploy" (top right) > "New deployment".
* - Select "Web app" as the type.
* - For "Execute as", choose "Me".
```

```
* - For "Who has access", choose "Anyone". (This is crucial for your frontend to reach it).
* - Click "Deploy".
* - You might be asked to authorize permissions. Review and allow them.
* - Copy the "Web app URL" that appears. This is the URL you will paste into your frontend HTML code.
* (Replace 'YOUR_GOOGLE_APPS_SCRIPT_WEB_APP_URL_HERE' in the JavaScript code).
* 7. Set up initial headers in your Google Sheet (Row 1 of the 'Submissions' sheet):
* Timestamp, Parent Name, Parent Email, Teen Interests Input, Suggested Careers (JSON), General Advice
*/

function doPost(e) {
 // Set CORS headers to allow requests from any origin.
 // This is essential for the web app to receive data from your frontend.
 var headers = {
   'Access-Control-Allow-Origin': '*',
   'Access-Control-Allow-Methods': 'POST, GET, OPTIONS',
   'Access-Control-Allow-Headers': 'Content-Type'
 };

 if (e.method == 'OPTIONS') {
   // Respond to pre-flight requests
```

```
   return
ContentService.createTextOutput(JSON.stringify({})).setHeaders(heade
rs).setMimeType(ContentService.MimeType.JSON);
 }

 try {
   // Parse the incoming JSON data from the frontend
   var requestData = JSON.parse(e.postData.contents);

   // Replace with your actual Spreadsheet ID
   var SPREADSHEET_ID = 'YOUR_SPREADSHEET_ID_HERE'; // <---
IMPORTANT: Update this!
   var SHEET_NAME = 'Submissions'; // <--- You can change this if
your sheet has a different name

   var ss = SpreadsheetApp.openById(SPREADSHEET_ID);
   var sheet = ss.getSheetByName(SHEET_NAME);

   // If the sheet doesn't exist, create it and add headers
   if (!sheet) {
     sheet = ss.insertSheet(SHEET_NAME);
     var headersRow = [
       "Timestamp",
       "Parent Name",
       "Parent Email",
       "Teen Interests Input",
       "Suggested Careers (JSON)",
       "General Advice"
     ];
```

```
    sheet.appendRow(headersRow);
  }

  // Extract data from the request
  var timestamp = new Date();
  var parentName = requestData.parentName || '';
  var parentEmail = requestData.parentEmail || '';
  var teenInterestsInput = requestData.teenInterestsInput || '';

  // Stringify the Gemini response to store it as JSON string in a
single cell
  var suggestedCareersJson =
JSON.stringify(requestData.geminiResponse.suggestedCareers || []);
  var generalAdvice = requestData.geminiResponse.generalAdvice ||
'';

  // Prepare the row to append to the sheet
  var rowData = [
    timestamp,
    parentName,
    parentEmail,
    teenInterestsInput,
    suggestedCareersJson,
    generalAdvice
  ];

  // Append the data as a new row
  sheet.appendRow(rowData);
```

```
    // Return a success response
    return ContentService.createTextOutput(JSON.stringify({ status:
"success", message: "Data saved successfully!" }))
      .setHeaders(headers)
      .setMimeType(ContentService.MimeType.JSON);

  } catch (error) {
    // Log the error for debugging
    Logger.log("Error in doPost: " + error.toString());

    // Return an error response
    return ContentService.createTextOutput(JSON.stringify({ status:
"error", message: error.message }))
      .setHeaders(headers)
      .setMimeType(ContentService.MimeType.JSON);
  }
}
```