

# HSN & SAC Master Data API Service Documentation

---

## Overview

This Go application is an API service for uploading and storing HSN (Harmonized System of Nomenclature) and SAC (Services Accounting Code) master data from Excel files into a PostgreSQL database. It uses the Echo web framework and the pgx PostgreSQL driver.

## Technologies Used

- Go: Programming language
- Echo: Web framework for building HTTP APIs
- pgx: PostgreSQL driver for Go
- excelize: Go library for reading Excel files
- PostgreSQL: Relational database

## Features

- Upload Excel files containing HSN and SAC data
- Parse and insert data into PostgreSQL
- Simple root endpoint with instructions

## Installation

- 1. Clone the repository:  
git clone https://github.com/surya20003/PES.git  
cd surya20003/PES
- 2. Set up PostgreSQL with a database named hsn-db and tables hsn\_mstr and sac\_mstr:  

```
CREATE TABLE hsn_mstr (  
  hsn_code TEXT PRIMARY KEY,  
  description TEXT,  
  gst_rate TEXT  
);  
  
CREATE TABLE sac_mstr (  
  sac_code TEXT PRIMARY KEY,  
  description TEXT,  
  gst_rate TEXT  
);
```

- 3. Install dependencies:  
go mod tidy
- 4. Run the server:  
go run  
hsn\_code.go

## Configuration

Database credentials are hardcoded in the application. You can change them in the constants section of hsn\_code.go:

```
const (  
    host    = "localhost"  
    port    = 5432  
    user    = "postgres"  
    password = "*****"  
    dbname  = "hsn-db"  
)
```

## API Endpoints

GET /

Returns a welcome message and information about the upload endpoint.

Response:

```
{  
    "message": "Welcome to the HSN & SAC Master Data API Service!",  
    "upload_endpoint": "/upload_excel",  
    "method": "POST (multipart/form-data with file field)"  
}
```

POST /upload\_excel

Uploads an Excel file with two sheets: hsn\_mstr and sac\_mstr. Parses and inserts the data into the respective PostgreSQL tables.

Request: multipart/form-data with a file field.

Response:

```
{  
    "message": "File uploaded and data inserted into DB successfully",  
    "file": "HSN_SAC.xlsx"  
}
```

Error Response:

```
{  
  "error": "Error message details"  
}
```

## Excel File Format

- hsn\_mstr: Columns - HSN Code, Description, GST Rate
- sac\_mstr: Columns - SAC Code, Description, GST Rate

## Deployment

- Use environment variables instead of hardcoded credentials
- Deploy using Docker or Kubernetes
- Use a reverse proxy like Nginx

## Logging & Error Handling

- Logs are printed to the console using Go's log package
- Echo middleware is used for request logging and recovery from panics

## Troubleshooting

- Cannot connect to database: Check the DSN and PostgreSQL server status
- Excel upload fails: Ensure the file is a valid .xlsx with correct sheet names
- Data not inserted: Check for constraint violations or missing columns