# PrecipGen PAR Design Document

By following this design document, the development of the PrecipGen PAR application can proceed in a structured and organized manner, ensuring a high-quality tool that meets the needs of its users.

# Overview

PrecipGen PAR is designed to generate a set of statistical parameters from an analysis of a historical time series of daily precipitation downloaded from the [Global Historical Climatology Network daily (GHCNd)](https://www.ncei.noaa.gov/products/land-based-station/global-historical-climatology-network-daily) database online. Results can be viewed in an output CSV file.

This document serves as a comprehensive guide for the development of the PrecipGen PAR application, outlining the functionalities required.

## Scope

The scope of this project includes:

1. Fetch and load data from the GHCN-d
2. Validate time series
3. Calculate parameters
4. Export results

# 4. Functional Requirements

The functional requirements describe how the user can interact with the program.

1. Execute the program from a command line terminal using this example command:
   * python pgpar USW00024233 “english”
2. Fetch and Load Data
   * **Load Data**: Get user choice of metric or standard (English) then click OK and perform the function that loads time series data from GHCNd into Python DataFrame and save any metadata to a dictionary called self.metadata. Update station summary on main window:
     + Metadata (station ID and name, units)
     + Start and end date of record
     + Percent of dataset that contains missing values
   * **Calculate PrecipGen Parameters**: Generates the following data.
     + 3 tables each containing 4 columns and 12 rows for each calendar month for:
       - Mean, SD, PWW, and PWD, calculated using all years on record
       - Same as above for 20 driest years
       - Same as above for 20 wettest years
     + 3 scalar outputs:
       - Average correlation coefficient for Mean and PWW
       - Average correlation coefficient for PWW and PWD
       - Autocorrelation for Annual totals
     + When complete, show a message dialog saying, “Calculation is Complete!”
   * **Settings** (allow user to adjust using a new window with widgets):
     + Start date (if different from the start date of the dataset)
     + End date (if different from the start date)
     + Autocorrelation lag (default is 1 year)
3. Validate Time Series
4. “Results” menu
   * **Sample Daily**: pick random month and year then plot all days in new plot window
   * **Annual Totals**: Show line chart of annual values
   * **PrcpGen Parameters**: Opens a new window displaying the calculated results where columns are parameters and the rows are the parameters.
5. Help
   * **About**: Displays information about the application.
   * **Help**: Open a website: www.dynsim.online

# 7. Functions

## Load Time Series

Function that takes data from source and loads it into a DataFrame and also parses metadata to store in python dictionary

# 8. Data Structures

## Model configuration

Data type: Python dictionary

Stores application settings: ghcn data file path

## Time Series

Data type: Pandas DataFrame

Stores the loaded time series data.

## Results

Data type: pandas DataFrame and scalar variables - Stores the calculated parameters.

# 9. Error Handling

* File Not Found: Display an error message if the selected file does not exist.
* Invalid File Format: Display an error message if the selected CSV file does not have the required columns.
* Save As: Prompt the user to save any unsaved changes before performing actions that could result in data loss.

# Testing

* + - site ID = “GHCND:USW00024233”
    - Site Name = “SEATTLE TACOMA AIRPORT, WA US”