# CSV File Importer

## Introduction

A surfboard shop has had some issues with their stock system. They are mainly paper based but need a system to read data from two sets of csv’s to determine which products are in stock and can be sold that day.

The detail of these files can be seen below:

### Product.csv

Fields:

Id – Unused Id Field

Name – Name of the product

Description – Product Description

SKU – Unique product reference code

### WarehouseInventory.csv

Fields:

SKU – Unique product reference code

ProductCount – Number of products available for sale

## Details

A framework for the application has already been created.

This has the following features already built:

* Logger for logging issues/errors
* Dependency injection has been implemented but is only used for the logger so far
* Data models have been created for both the Product and Warehouse inventory data
* Appsettings and config are both present and pre-populated with basic data

## Tasks to complete

1. Add functionality to import csv data (this can either be from a set file location or the user could be prompted for a string file location, or this could be included in a config file!)
2. The columns will need to be validated before loading and presenting to the user.
3. The user should only be presented with products that have a warehouse inventory amount of > 0
4. The user should only see the Product Name, Description and ProductCount fields on successful import
5. Any errors should be added to a file and output to the error location, so should the excel file which contains the error.
6. The error file name should be the name of the excel file plus "-errors" on the end.
7. Both of these files should be placed in an error folder, which is set in the appsettings.json.
8. The format of the error file is up to you, so long as the row number and the column and error reason is listed for each line in the Excel file that has failed validation.
9. The user should also see a visual representation of these error in the form of an error message with some detail pertaining to the error and given the option to re-upload without reloading the application
10. Any files that do not error, need to be placed in the successful folder, again configured in the appsettings.json
11. The product name and des

## Data

Two files are present in the solution underneath the data folder. These are called ‘Products.csv’ and ‘WarehouseInventory.csv’.