README.md 2023-10-13



### Varacell

▶ Table of Contents

# **About The Project**

A simple and very performant templating engine for working on Excel files using the file itself as a template

- Define rows to start writing records to in each sheet, in the event you have long headers
- Define multiple templates in a convenient manner
- Use a single sheet to define the various content you wish to generate, even across multiple sheets

(back to top)

#### **Built With**

- Go
- Excelize
- Cobra

(back to top)

# **Getting Started**

If you are not compiling the application, all you need is the binary and a spreadsheet to get started! If you are going to be making modifications to the application, you will need a GoLang developers environment and access to the internet to download the modules

#### Installation

The regular Go compile can be used if you are only intending to run the application on your architecture

```
go build -o varacell .
```

### Compiling for targeted architecture

```
# Mac
GOOS=darwin GOARCH=amd64 go build -o ./varacell .
# Windows
```

README.md 2023-10-13

```
GOOS=windows GOARCH=amd64 go build -o ./varacell.exe .

# Linux
GOOS=linux GOARCH=386 go build -o ./varcell .
```

(back to top)

# Usage

The source workbook should have the following sheets (can be configured through Viper) present, see the included example for a working example using default settings

#### Sheet Control

Determines the initial row state to start writing to

- Sheet Name Name of the sheet to setup control for
- Starting Row Row to start writing data from



### **Template**

Define some reusable templates. Each template should start with an identifier name and finish with END using the first column

- Template Identifier for the template which will be used in the generator section
- Sheet Spreadsheet to output the content to
- Columns Determines the length of each row to simplify empty box handling
- Content Columns and rows to be rendered (including the template tags) when generated
  - Dynamic variables should be defined using the following syntax {{ .VARIABLE\_NAME }}.
     More info on variable name in the generators section

Template	Sheet	Columns	Content			
Template 1	Form Status Config	37	449	SSR	Template-SSR-WSI	
			449	SSR	Template-SSR-WSI	
			449	SSR	Template-SSR-WSI	
			449	SSR	Template-SSR-WSI	
			449	SSR	Template-SSR-WSI	
			449	SSR	Template-SSR-WSI	
			449	SSR	Template-SSR-WSI	
			449	SSR	Template-SSR-WSI	
			449	SSR	Template-SSR-WSI	
			449	SSR	Template-SSR-WSI	
			449	SSR	Template-SSR-WSI	
			449	SSR	Template-SSR-WSI	
			449	SSR	Template-SSR-WSI	
END					•	
Prefix	Form Status Config	4	449		{{ .sharon }}	Checkli &
			449		{{ .sharon }}	Checkli &

### Things To Generate

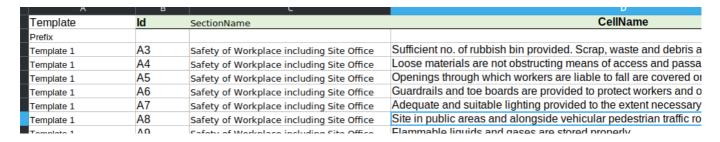
Items that are to be generated.

README.md 2023-10-13

• Template - Template to use

The program will automatically treat any further columns as template variables (identified by the value given in the first row).

- Empty values are allowed and will simply be left empty at runtime
- Variable names should be single words



(back to top)

## Maintainers

An up-to-date list of people involved with development / support of the project

Simon Ball - simonball@varadise.cloud

(back to top)