



# R PACKAGE VALIDATION WORKSHOP

Elements of  
Validation

# Welcome

- {valtools} infrastructure
- Elements of Validation
- Reports



# THE SETUP

# {valtools} infrastructure

- report .RMD
- validation folder
  - validation components
  - validation.yml
    - Working directory
    - Output directory
    - Users
    - Ordering of validation child files
  - [optional] change log
- [as needed] .gitignore, .here

```
· | - validation
· | · | - requirements/
· | · | - test_cases/
· | · | - test_code/
· | · | - validation.yml
· | · | - change_log.md
· | · | - .gitignore
· | ·
· | - validation.Rmd
```



# THE ELEMENTS

# Validation Components

“confirmation by examination and provision of objective evidence that software specifications conform to user needs and intended uses, and that the particular requirements implemented through software can be consistently fulfilled.”

# Validation Components – what needs to be authored?

- Requirements
- Test Cases
- Test Code

# Validation Reporting – what can be tracked by automating?

- Authorship
- Most recent edit dates
- Testing environment
- Test results



# Requirements - vt\_use\_req()

- Key Arguments
  - name,
  - username
- Name – requirement name
- Username – Person's name writing requirement
- Markdown document with proper roxygen headers
  - Editor
  - editDate
  - riskAssessment

# Test Cases - vt\_use\_test\_case()

- Key Arguments
  - name,
  - username
- Name – requirement name
- Username – Person's name writing test case
- Markdown document with proper roxygen headers
  - Editor
  - editDate
  - coverage

# Test Code - vt\_use\_test\_code()

- Key Arguments
  - name,
  - username
- Name – requirement name
- Username – Person's name writing test code
- Markdown document with proper roxygen headers
  - Editor
  - editDate

# Users

- valtools tracks users and their roles in the validation.yml file
- If a user creates a file not seen before, prompted for their user information
- Manually add a person via **vt\_add\_user\_to\_config()**
  - Need: username, name, role, title

```
> vt_add_user_to_config(username = "ellis_h")
```

Please supply some information for recording users within the package.

Note, that this information can be updated at any time though

```
`vt_add_user_to_config()`
```

Please provide the name of the person associated with the username

`ellis\_h` and press `Enter`.

**Name: Ellis Hughes**

Please provide the title of the person associated with the username

`ellis\_h` and press `Enter`.

**Title: Programmer**

Please provide the role of the person for this validation associated with the username `ellis\_h` and press `Enter`:

**Role: validation Lead**

User `ellis\_h` added to validation config file.

```
>
```

# vt\_use\_\* prompting

```
> vt_use_req("requirement_001")
Item created:requirements/requirement_001.md
User `ehhughes` does not exist in the config file
Add user to config file?
[Y/n]: Y
Please supply some information for recording users within the package.
Note, that this information can be updated at any time though
`vt_add_user_to_config()`

Please provide the name of the person associated with the username
`ehhughes` and press `Enter`.
Name: Ellis Hughes

Please provide the title of the person associated with the username
`ehhughes` and press `Enter`.
Title: Programmer

Please provide the role of the person for this validation associated with
the username `ehhughes` and press `Enter`:
Role: Requirements writer

User `ehhughes` added to validation config file.
Filename(s): requirement_001.md added to validation config file.
* Modify 'C:/Users/ehhughes/Documents/Projects/R PACKAGE VALIDATION FRAMEWORK/test1234/vig
nettes/validation/requirements/requirement_001.md'
> |
```



# COMPLETE

## EXERCISE 4 TASK A



# REPORT



# Creating the validation report Rmd

- `vt_use_report()`
- Creates report from template with suggested contents
- Adds dependencies to package DESCRIPTION
- Two templates included: full validation report and a requirements approval report

# Report basics

- Rmarkdown
- Scrapes information saved across various files
- Produces kables
- Reads in the validation child files and parses them based on order in validation.yml

# Validation Report Helpers

- `vt_path()`
  - `here::here()`, but for validation contents
- `vt_file()`
  - Renders contents based on extensions
    - Rmd parsed
    - Test code executed and results captured
    - Any other file contents are printed

# Child Files

- Validation Report contains pointers for repeated evaluation across different validation environments
- Order of child files can be specified by validation.yml
  - Default is grouped by “requirements”, “test\_cases”, “test\_code” and then alphabetically
  - Control via vt\_get\_child\_files()
- vt\_add\_file\_to\_config() & vt\_drop\_file\_from\_config()
  - Specify validation file (req, test case, test code)
  - Specify which file to add after/before using tidyselect syntax
  - Files added by default to the end by vt\_use\_\*

# Rendering Reports

- `vt_validate_report()` executes the Rmd and saves output
- Serves as “snapshot” in time



# COMPLETE

## EXERCISE 4 TASK B



# REPORT ORDERING

# Validation Reporting Order–

How does valtools know the order to report everything

- validation.yml
- Updated as vt\_use\_\* called
  - On creation call, can set location
  - Edit yml directly or relocate through:
    - vt\_drop\_file\_from\_config()
    - Vt\_add\_file\_to\_config()





# COMPLETE

## EXERCISE 4 TASK C