



R PACKAGE VALIDATION WORKSHOP

Internal Package
Validation

Welcome

- Internal Packages
- Validation Approaches for Internal Packages



INTERNAL PACKAGES

{valtools} Packages

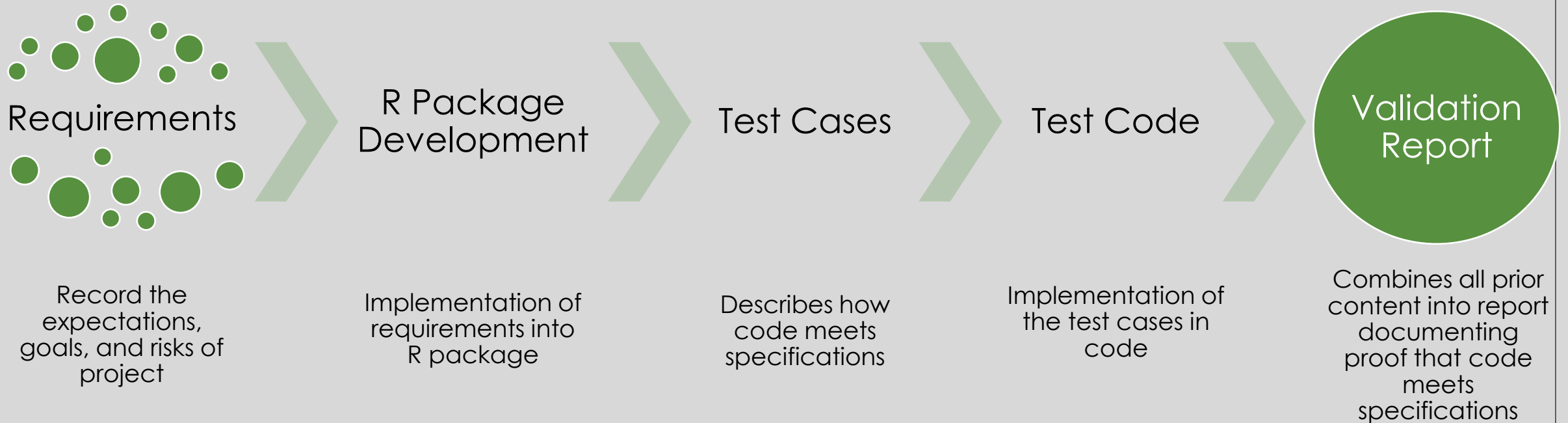
- vt_create_package()
 - Package folder
 - Valid package name
 - Creates package infrastructure
 - Creates valtools validation infrastructure

{valtools} Packages

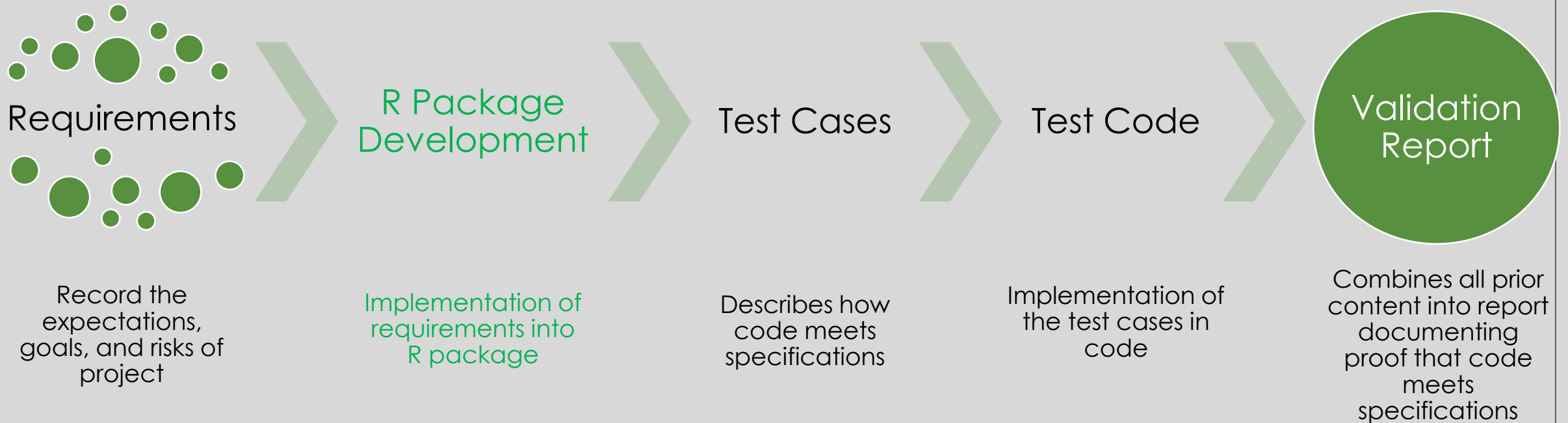
- Vignettes folder (default)
 - Validation Elements
 - validation.yml
- Vignette serves as validation report

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

Package Development

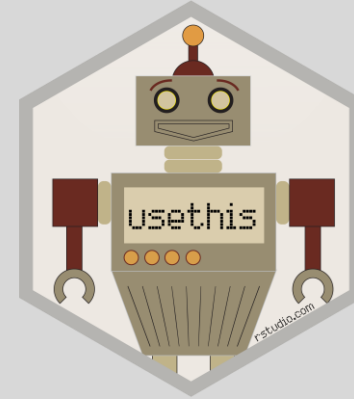


Package Development



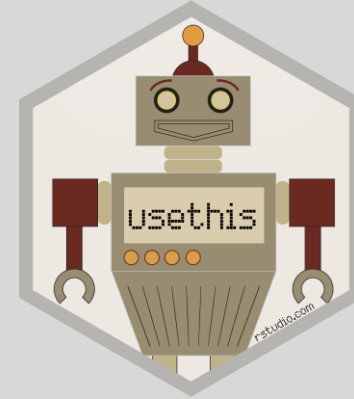
Package Development

- Standard Package Development Process
 - usethis
 - devtools
 - roxygen2
 - rmarkdown
 - testthat



Package Development

- Standard Package Development Process
 - usethis
 - devtools
 - roxygen2
 - rmarkdown
 - testthat
- valtools set at an imported package
 - Support specialized roxygen tags
 - editor
 - editdate
 - Validation Vignette Compilation





SHARING YOUR PACKAGE VALIDATION



SOURCE

vt_validate_source()

- For testing/release of source code
- Temporary installation to test on system



BUILD

vt_validate_build()

- Builds source code, includes all components of validation into tarball
- For release of source code with controlled versions
- Users have access to install as necessary



INSTALL

vt_validate_install()

- Builds tarball and Installs
- Runs validation report on installed system



RE-V VALIDATION

vt_validate_installed_package()

- ONLY works for the following
 - vt_validate_install()
 - tarball created by `vt_validate_build()`
- Run when environment has changed, but installation has not