

Package: AAGIQuartoExtra (via r-universe)

May 9, 2026

Title Facilitates the Creation of AAGI Quarto Documents

Version 0.9.0

Description Assists in the creation of AAGI Quarto documents by providing a set of functions that generates the document structure and style. The package is designed to work with Quarto which is an open-source document format that supports R Markdown and Jupyter Notebooks. The package provides functions to create the document structure, add sections, figures, tables, and references. It also provides functions to download the AAGI document style, including logos, fonts, and colours.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

URL <https://github.com/AAGI-AUS/AAGIQuartoExtra>,
<https://AAGI-AUS.github.io/AAGIQuartoExtra/>

BugReports <https://github.com/AAGI-AUS/AAGIQuartoExtra/issues>

Imports fs, remotes, rstudioapi

Suggests renv, reticulate, systemfonts

Config/Needs/website quarto

Config/pak/sysreqs cmake git make libuv1-dev

Repository <https://aagi-aus.r-universe.dev>

Date/Publication 2026-04-09 03:48:20 UTC

RemoteUrl <https://github.com/AAGI-AUS/AAGIQuartoExtra>

RemoteRef HEAD

RemoteSha aee0833ca2ae5414f7d79f01be0917cbaa09f83e

Contents

aagi_project	2
check_fonts	3
check_proxima_nova_installed	3
create_aagi_ext	3
display_font_setup_instructions	4
download_fonts	4
install_aagi_ext	5
new_aagi_document	6
setup_docs	6
setup_folders	7
setup_github_pat	8
Index	9

aagi_project	<i>Create a new AAGI project</i>
--------------	----------------------------------

Description

Create a new AAGI project with the following structure:

Usage

```
aagi_project(path, ...)
```

Arguments

path	Path to save the new project
...	Additional arguments

Details

- A new directory with the name of the project
- A `_quarto.yml` file with the title of the project
- A `index.qmd` file with the title of the project
- A `references.bib` file
- A `pyenv` directory if `python_venv` is `TRUE`
- A `renv` directory if `with_renv` is `TRUE`

Value

A new AAGI project

check_fonts	<i>Check if the Proxima Nova font is installed</i>
-------------	--

Description

This function checks if the Proxima Nova font is installed on the user's system. If the font is not installed, it provides instructions on how to set up the font for use in the Quarto project.

Usage

```
check_fonts()
```

check_proxima_nova_installed	<i>Internal: Check if the Proxima Nova font is installed</i>
------------------------------	--

Description

This function checks if the Proxima Nova font is installed on the user's system.

Usage

```
check_proxima_nova_installed()
```

Value

TRUE if the Proxima Nova font is installed, FALSE otherwise

create_aagi_ext	<i>Create new AAGI Quarto project from template</i>
-----------------	---

Description

Install bundled Quarto extensions into current working directory and create new qmd using skeleton documents. This function extends a function written by Thomas Mock: <https://github.com/jthomasmock/octavo/blob/master/> and by Spencer Schien: https://spencerschien.info/post/r_for_nonprofits/quarto_template/. This is equivalent to the quarto command `quarto use` with a `--template` flag.

Usage

```
create_aagi_ext(
  file_name = NULL,
  ext_name = "aagi-report",
  university = "AU",
  path = "."
)
```

Arguments

file_name	Name of new qmd file and sub-directory to be created
ext_name	String indicating which extension to install
university	String indicating which university partner to use
path	Path to save new qmd file

Value

a message if extension was successfully copied over

display_font_setup_instructions

Internal: Function to display instructions for setting up the font

Description

This function displays instructions for setting up the Proxima Nova font for use in the Quarto project.

Usage

```
display_font_setup_instructions()
```

download_fonts

Download Proxima Nova fonts from the AAGI private repository

Description

This function downloads the Proxima Nova font files from the private AAGI repository and extracts them to the specified destination directory.

Usage

```
download_fonts(
  font_repo = "jafernandez01/font-resources",
  font_dir = "fonts/",
  dest_dir = "./_extensions/aagi-report",
  pat = Sys.getenv("GITHUB_PAT")
)
```

Arguments

font_repo	The GitHub repository containing the font files
font_dir	The directory within the repository containing the font files
dest_dir	The destination directory to extract the font files
pat	The GitHub Personal Access Token (PAT) to access the private repository

Value

TRUE if the fonts are downloaded and extracted successfully, FALSE otherwise

install_aagi_ext	<i>Install or Update AAGI Quarto Extension</i>
------------------	--

Description

This function installs a specified AAGI Quarto extension into the current working directory or updates an existing installation to the version included in the package. The extension will be installed to the '_extensions' folder in the current directory.

Usage

```
install_aagi_ext(ext_name = "aagi-report", force = FALSE)
```

Arguments

ext_name	String indicating which extension to install. Must be one of "aagi-report", "aagi-short-report", "aagi-presentation", or "all" to install/update all bundled extensions.
force	Logical indicating whether to force installation even if the extension already exists. Default is FALSE, which will prompt the user when an existing extension is found.

Value

Invisibly returns TRUE if the extension was installed or updated, FALSE otherwise.

Examples

```
## Not run:  
# Install the AAGI report extension  
install_aagi_ext("aagi-report")  
  
# Force update the AAGI short report extension  
install_aagi_ext("aagi-short-report", force = TRUE)  
  
## End(Not run)
```

new_aagi_document	<i>Create new Quarto document from template</i>
-------------------	---

Description

Create a new AAGI Quarto document. If the extension is not already installed, it will be copied over to the `_extensions` folder in the current working directory. This function extends a function written by Thomas Mock: https://github.com/jthomasmock/octavo/blob/master/R/use_quarto_ext.R and by Spencer Schien: https://spencerschien.info/post/r_for_nonprofits/quarto_template/

Usage

```
new_aagi_document(
  file_name = NULL,
  ext_name = "aagi-report",
  university = "UQ",
  update = FALSE
)
```

Arguments

file_name	Name of new qmd file and sub-directory to be created
ext_name	String indicating which extension to install
university	String indicating which university partner to use
update	Logical indicating whether to update the extension if it already exists

setup_docs	<i>Rename the index.qmd File and Update _quarto.yml for a Quarto Project</i>
------------	--

Description

This function renames the `index.qmd` file to a new name based on the project name and output type. It also updates the `_quarto.yml` file to specify the output type and directory for the project. This function should be run in the root directory of a Quarto project and after the `setup_folders()` function.

Usage

```
setup_docs(output_type = "Report")
```

Arguments

output_type	Character. The type of output to be generated (e.g., "Report", "Slides", "Manuscript").
-------------	---

Details

The function extracts the project name from the directory name and uses it to rename the `index.qmd` file. The new file name format is `01_projectname_outputtype.qmd`. The `_quarto.yml` file is updated with the output type and directory for the project.

setup_folders

Create a Standard Folder Structure for a Quarto Project

Description

This function creates a standardized folder structure for a Quarto project. The structure includes folders for data, code, figures, documents, outputs, source files, and admin tasks.

Usage

```
setup_folders(project_path = ".")
```

Arguments

`project_path` Character. The root path of the Quarto project where the folders will be created.

Details

- **data**: Contains the raw data files used in the project. These files should not be altered and are ideally read-only.
- **code**: Contains the scripts or code files used to run the project.
- **figs**: Contains plots, images, tables, or figures created and saved by your code. It should be possible to delete and regenerate this folder with the scripts in the project.
- **doc**: Contains any manuscripts or interim summaries produced during the project.
- **output**: Contains non-figure objects created by the scripts, such as processed data or logs.
- **src**: Contains any files you may want to `source()` in your scripts, such as `.R` files containing helper functions.
- **admin**: Contains administrative files related to the project (e.g., notes, project planning documents, and metadata).

Note

Folder structure inspired by Kathryn Destasio's best practices for R projects: https://kdestasio.github.io/post/r_best_practices/

setup_github_pat *Helper function to setup GITHUB_PAT interactively*

Description

This function provides instructions to the user on how to create a GitHub Personal Access Token (PAT) and set it up for use in the current R session.

Usage

```
setup_github_pat()
```

Value

TRUE if the GITHUB_PAT is successfully set up, FALSE otherwise

Index

aagi_project, 2

check_fonts, 3

check_proxima_nova_installed, 3

create_aagi_ext, 3

display_font_setup_instructions, 4

download_fonts, 4

install_aagi_ext, 5

new_aagi_document, 6

setup_docs, 6

setup_folders, 7

setup_github_pat, 8