

# Descripción general e inicio rápido

## Descripción de la propuesta

La herramienta *~!gurí\_* es un flujo de trabajo y un conjunto de herramientas que facilitan una automatización del proceso de generación de documentos finales para revistas científicas a partir de documentos obtenidos en la etapa de ‘corrección de pruebas’. La herramienta se basa en el uso de [Pandoc](#) como herramienta de conversión entre formatos, a la cual se incorpora un conjunto de filtros *Lua* y plantillas personalizadas, así como un flujo de trabajo, que permite en su conjunto la creación de los documentos finales. La propuesta parte del hecho de que muchas revistas utilizan como base de sus flujos de trabajos documentos *docx*.

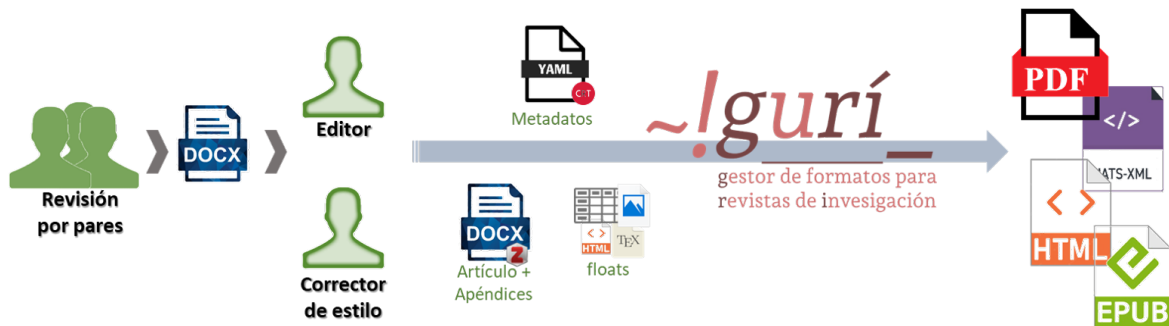


Figura 1: Esquema general

La propuesta permite una fuerte adaptación de la apariencia de los formatos finales, a partir de la utilización de plantillas personalizadas.

En términos resumidos, la propuesta propone esquematizar y separar los principales elementos que componen un artículo científico. Para ello, para cada artículo, se debe generar un archivo *docx* (generado mediante una plantilla y con las citas generadas mediante Zotero) y un archivo *yaml* con los metadatos del artículo. Opcionalmente, puede brindarse otros archivos con información de la [taxonomía CRediT](#), elementos flotantes (figuras y tablas) y apéndices. A partir de estos archivos iniciales se genera un archivo intermedio *markdown*, que se utiliza para generar los documentos finales (en formato *pdf*, *html* y *xml*).

## Descripción de las dependencias de software

Esta herramienta se basa en el uso de [Pandoc](#) (versión 3.1.10 o superior) como herramienta de conversión entre formatos. Si no tiene instalado este software deberá hacerlo directamente de la página oficial.

También es necesario instalar alguna versión de [R](#) (recomendable versión 4.3 o superior), que es el encargado de coordinar el proceso de conversiones. Además, es necesario instalar los paquetes `tidyverse`, `rmarkdown`, `readxl`, `tinytex`, `crayon`. Estos paquetes son instalados automáticamente (si no están ya instalados), mediante la función `GURI_install()`.

Es necesario tener instalada alguna distribución de [LaTeX](#). A las personas con poca experiencia en el uso de [LaTeX](#), recomendamos utilizar la distribución [tinytex](#), por su integración con R y la facilidad para manejar las dependencia de paquetes. Es recomendable tener instalados los siguientes paquetes de [LaTeX](#) que son utilizados por la plantilla de Pandoc: [amsfonts](#), [amsmath](#), [lm](#), [unicode-math](#), [iftex](#), [listings](#), [fancyvrb](#), [longtable](#), [booktabs](#), [graphicx](#), [hyperref](#), [xcolor](#), [soul](#), [geometry](#), [setspace](#), [babel](#), [xeCJK](#), [fontspec](#), [selsolig](#), [mathspec](#), [biblatex](#), [bibtex](#), [biber](#), [upquote](#), [microtype](#), [csquotes](#), [natbib](#), [bookmark](#), [footnotehyper](#), [footnote](#), [xurl](#), [parskip](#) y [svg](#). A su vez, la adaptación de la plantilla utiliza los siguientes paquetes: [adjustbox](#), [fontawesome5](#), [caption](#), [ccicons](#), [relsize](#), [truncate](#), [lastpage](#) y [koma-script](#). Si no tiene instalada la distribución `tinytex` y estos paquetes, pueden ser instalados automáticamente con la función `GURI_install(install_tinytex = T)`.

## Inicio rápido

Para realizar una prueba rápida de `~!guri_` puede utilizar el ejemplo disponible en la carpeta `./example/`.

En primer lugar deberá clonar el repositorio (o descargarlo manualmente).

```
git clone https://github.com/estedeahora/guri/
```

Dentro del directorio de `~!guri_` debe abrir el proyecto de R mediante el archivo `guri.Rproj`. Si es la primera vez que usa `~!guri_` es posible que previamente sea conveniente ejecutar manualmente el siguiente código para instalar las librerías de R necesarias, así como la distribución de `Tinytex`:

```
source("scripts/GURI_00_fx.R", encoding = "UTF-8")
GURI_install()
```

Luego abra el script `GURI_make-files.R` ubicado dentro de la carpeta `./scripts/` y ejecute el mismo siguiendo las indicaciones (para una versión de prueba no debería modificar nada). Como resultado, dentro de la carpeta de cada artículo (en la carpeta `./example/num1/`) encontrará un directorio `_output` con los archivos finales resultantes.