# Markdown to PDF Investigation

As quite a lot of Integrated Centre documentation exists in Markdown format in Refactory, this could make it inconvenient for some researchers to go and find the documentation’s rendered form in Refactory. This documentation should exist in a form outside of Refactory (e.g. on PRN), and to do so, should be in a form better suited to general viewing – PDF.

The tool selected to convert Markdown to PDF must be able to satisfy the following requirements:

1. Maintain relative links to sections within the PDF.
2. Support headers and footers to allow classification banners when on Defence networks.
3. Create one PDF file for N markdown pages including links between markdown pages.

This investigation covers some tools that claim to convert Markdown to PDF.

|  |  |  |
| --- | --- | --- |
| **Tool** | **Pros** | **Cons** |
| MDPDF (NPM) - [https://github.com/BlueHatbR it/mdpdf](https://github.com/BlueHatbRit/mdpdf) npm install mdpdf --save | * Great documentation. * Has a simple programmatic API. Could make tool production easy. * No issues seem to suggest anything is not working. * Maintains relative links. | * No out of the box support for multiple files at once. * Has not been updated in 8 months. * Says it has support for headers and footers, but the implementation seems to be not working right now. * Cannot render mermaid diagrams * Cannot render plantuml diagrams |
| Markdown to PDF (NPM) - <https://github.com/simonhaenisch/md-to-pdf>npm install md-to-pdf --save | * Seems to be actively updated. * Support for headers and footers. * No issues seem to suggest anything is not working. * Providing `shopt -s globstar` has been set, doing a batch of .md files is possible with a glob (i.e. ./\*\*/\*.md). | * Does not maintain section links. * When converting a batch, file-by-file styling is not possible. This means things that require landscape orientation will have to be done manually. * Converts all links to a `localhost:xyz` url. * Cannot render mermaid diagrams * Cannot render plantuml diagrams |

The tools have been tested with some SVGs They seem to render fine, providing they are not too big. Unfortunately, the PDF tools cannot figure out if something would span over the page and cut off, so manual intervention would be needed to render quite a few of these. For example, a lot of the rogue shadow technical plan documents are huge and require a lot of effort for them to not be cut off.

Unfortunately, neither tool has the ability to bulk convert N markdown files into a single PDF document. This would be very useful to have as this would ensure everything stays in a single document and is easy to find. Something like this also introduces the issue of maintaining links to sections and other documents. There could be some annoying complexity here.

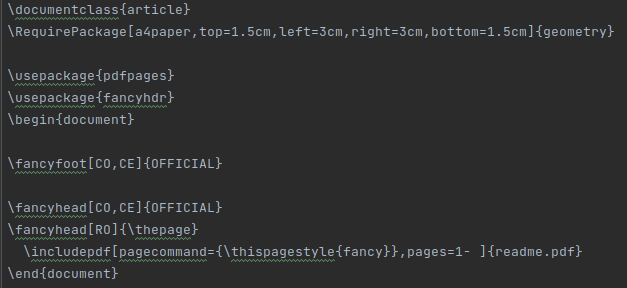
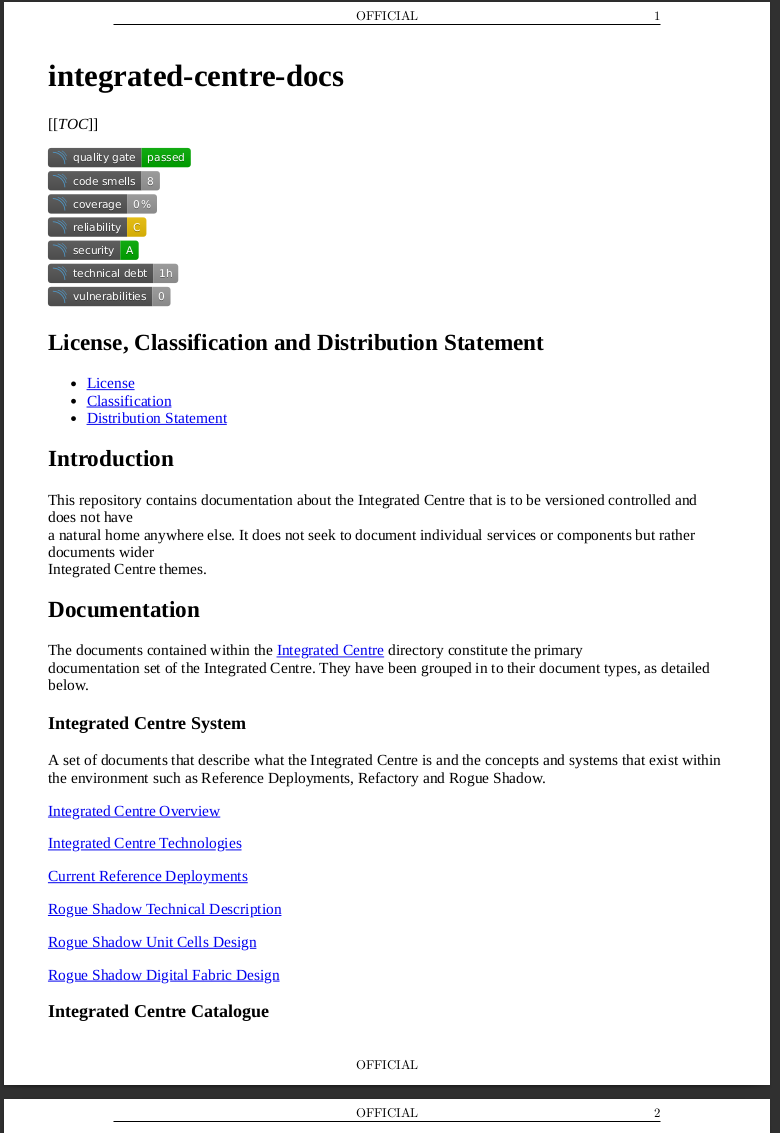
In summary, neither tool perfectly satisfies the requirements. However, both tools have a relatively small codebase, and both have either an MIT licence or an Apache 2.0 licence. This means that the code could be pulled into Refactory and modified to provide the required functionality. The disadvantage of this approach is Integrated Centre inherits the ongoing maintenance.

## Recommendations

* Raise the identified problems as issues on the tools in GitHub.
* Continue to review the ongoing development of both tools.
* Raise a new ticket in Azure Boards (with a link to this investigation document) to create PDF conversion capability after a re-review of current tools.
* Schedule this work when requirement of this capability is required.
* Start investigating creating a tool to meet our needs ourselves.

# Follow On Investigation

The suggestion to use a separate tool to stamp the documents with headers and footers has been suggested. It’s possible to use LaTeX to add bits onto PDFs using `pdflatex` tool. This is done by creating the header and footer in LaTeX and using the `pdfpages` tool to insert the pdf into the latex document, then re-rendering the LaTeX document. An example of the rendered page along with its LaTeX file that was used.



To generate the PDF document, run the following command.

***pdflatex addheader.tex***