

Creating a documentation site for users with AsciiDoc and Antora

Alexander Schwartz, Principal IT Consultant
FOSDEM Tool The Docs Dev Room, 2021-02-07

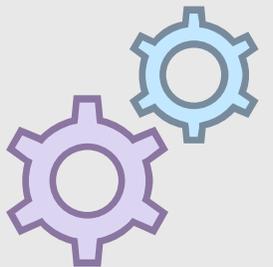
Empower your users with documentation so they get their work done



Keep your content up-to-date



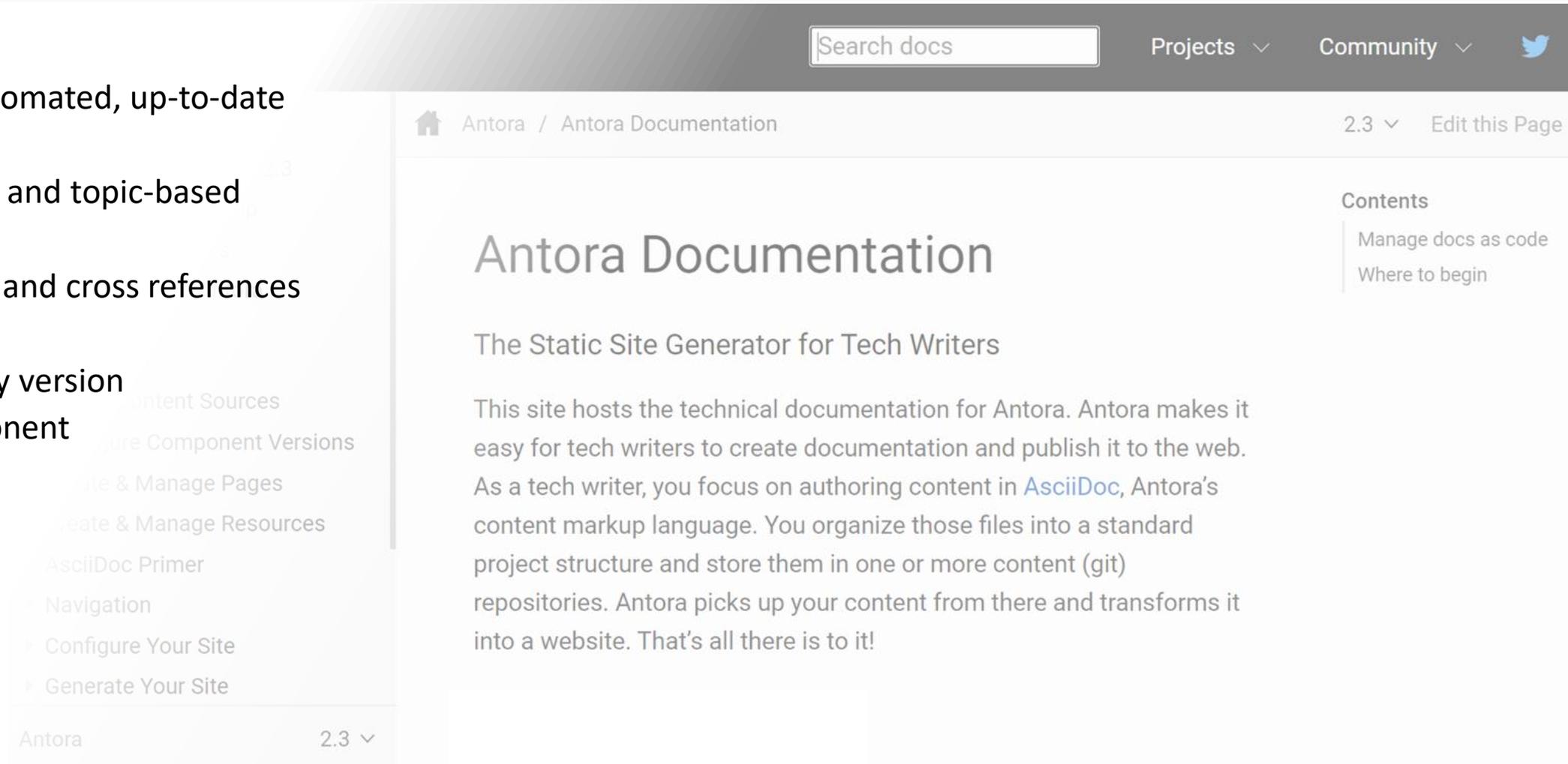
Keep your content consistent

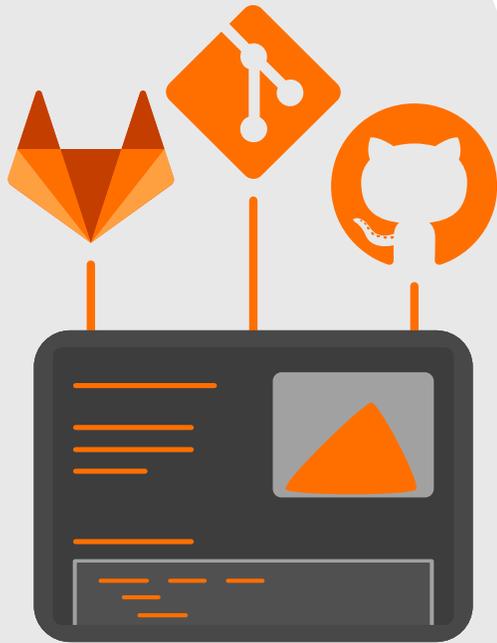


Publish new features for software and documentation in sync

Empower your users with documentation so they get their work done

- Online, automated, up-to-date
- Searchable and topic-based
- Navigation and cross references
- Grouped by version and component





- Structures documentation into components and modules
- Collects AsciiDoc content from multiple Git repositories and branches
- Converts AsciiDoc content to HTML output
- Merges HTML output with UI theme to create a static site

Key benefits:

- Fast conversion built on top of Node.js and minimal dependencies
- Modular and extendible

AsciiDoc is the language, AsciiDoctor is a toolchain

AsciiDoc

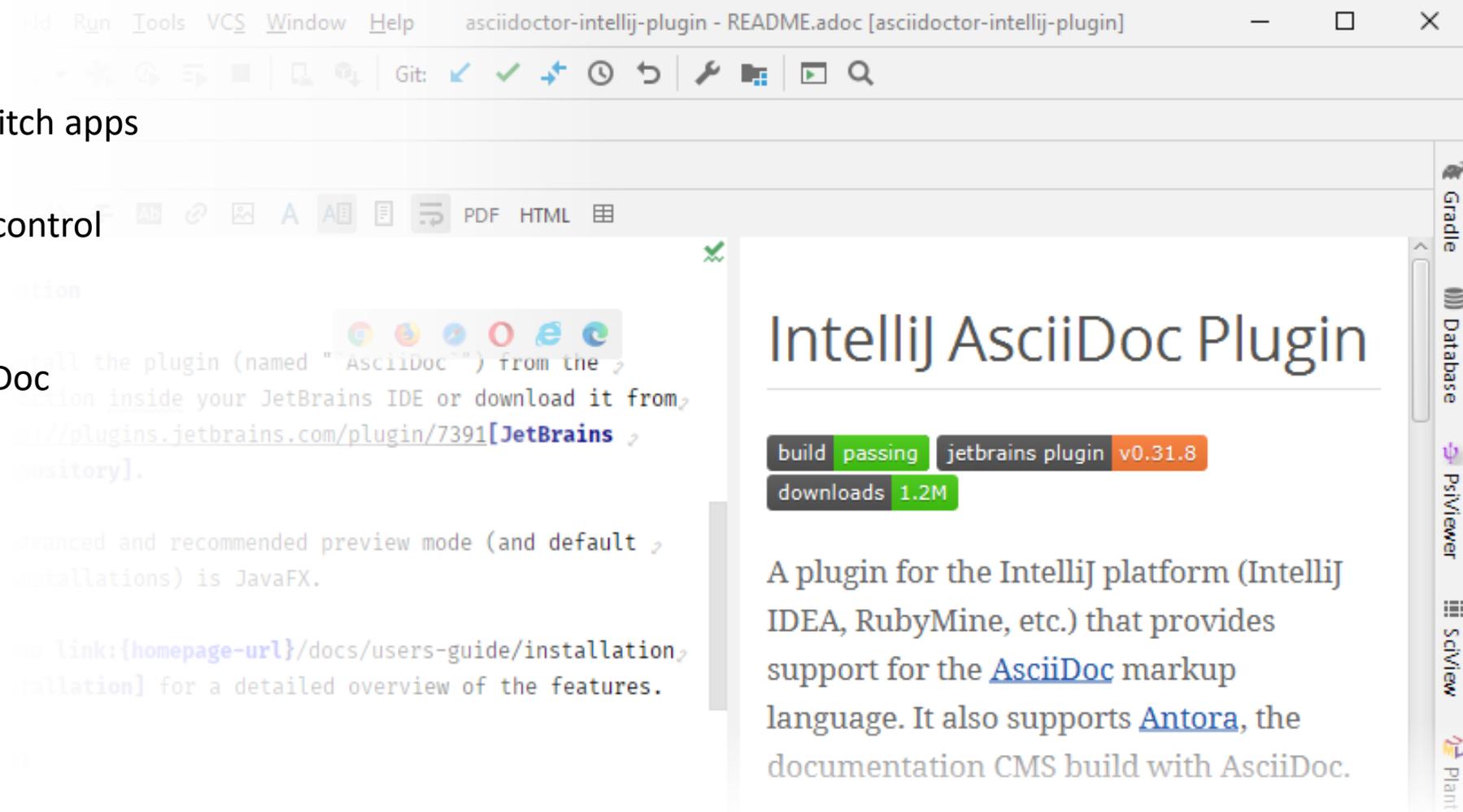
- friction-less writing using plain text
- feature-rich syntax for technical documentation
- 15+ years alive and kicking
- standardization started in 2020 at Eclipse Foundation

AsciiDoctor

- toolchain to create HTML and other outputs from AsciiDoc
- OpenSource
- runs on Java, Ruby and JavaScript runtimes
- powers Antora's AsciiDoc to HTML conversion



- stay focused and don't switch apps
- collaborate using version control
- IntelliJ AsciiDoc plugin provides Antora and AsciiDoc support



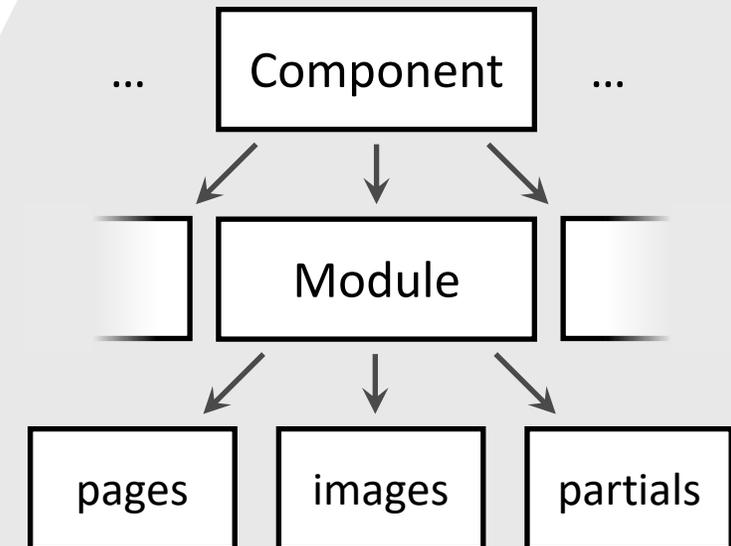
List of editors:

<https://asciidoctor.org/docs/editing-asciidoc-with-live-preview/>

- Multiple independent components
- Each component is identified by name and version
- Each component can have multiple modules
- Each module has pre-defined families for pages, images, examples, attachments and partials (content snippets to be included in multiple places)

Advanced feature:

A component can be distributed across multiple folders, branches and repositories

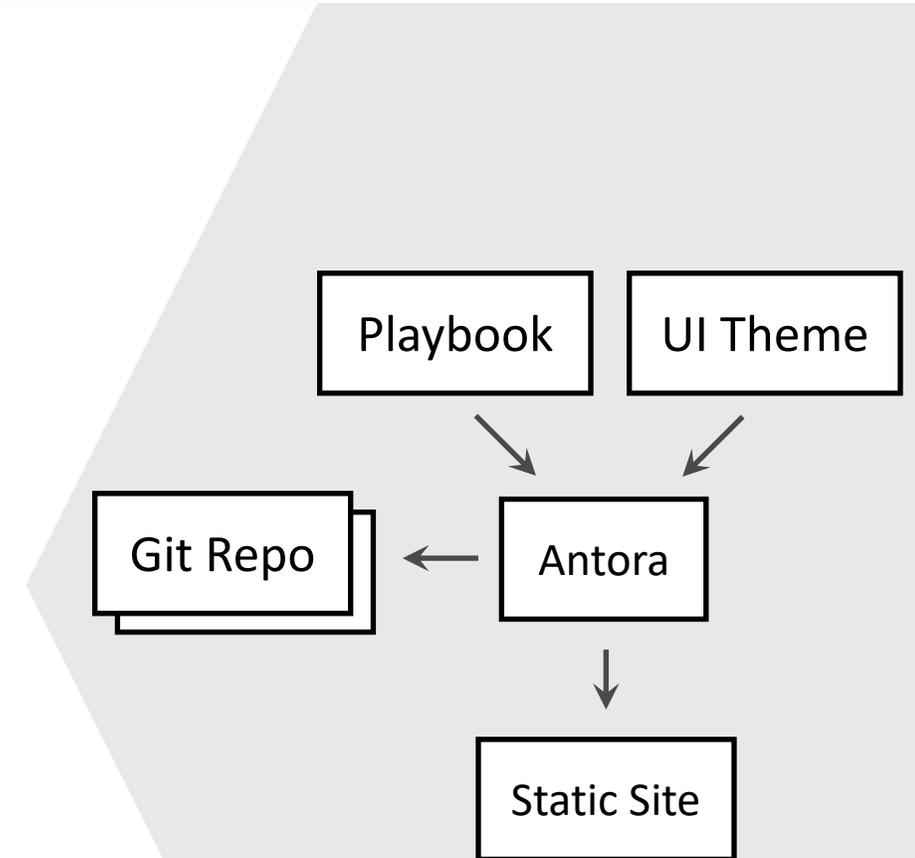


What the user does:

1. Run Antora

What Antora does:

1. Load Antora playbook with list of repositories and branches
2. Clone Git repositories
3. Create HTML output from AsciiDoc
4. Merge with UI theme and UI customizations
5. Optional: Create search index
6. Create output folder with static site



Role	Professional Skills	Technical Skills
Author	Structuring and Writing	<ul style="list-style-type: none">• Git (to collaborate with other authors)• AsciiDoc (to create content)

Role	Professional Skills	Technical Skills
Author	Structuring and Writing	<ul style="list-style-type: none">• Git (to collaborate with other authors)• AsciiDoc (to create content)
Docu Ops	Automation and Infrastructure	<ul style="list-style-type: none">• Git (for initial setup)• YAML (for configuration)• package.json (for version management)• Continuous Integration (for automation)

Role	Professional Skills	Technical Skills
Author	Structuring and Writing	<ul style="list-style-type: none">• Git (to collaborate with other authors)• AsciiDoc (to create content)
Docu Ops	Automation and Infrastructure	<ul style="list-style-type: none">• Git (for initial setup)• YAML (for configuration)• package.json (for version management)• Continuous Integration (for automation)
Web Developer	Web Site Creation	<ul style="list-style-type: none">• HTML/CSS (to customize Antora theme)

Writer:

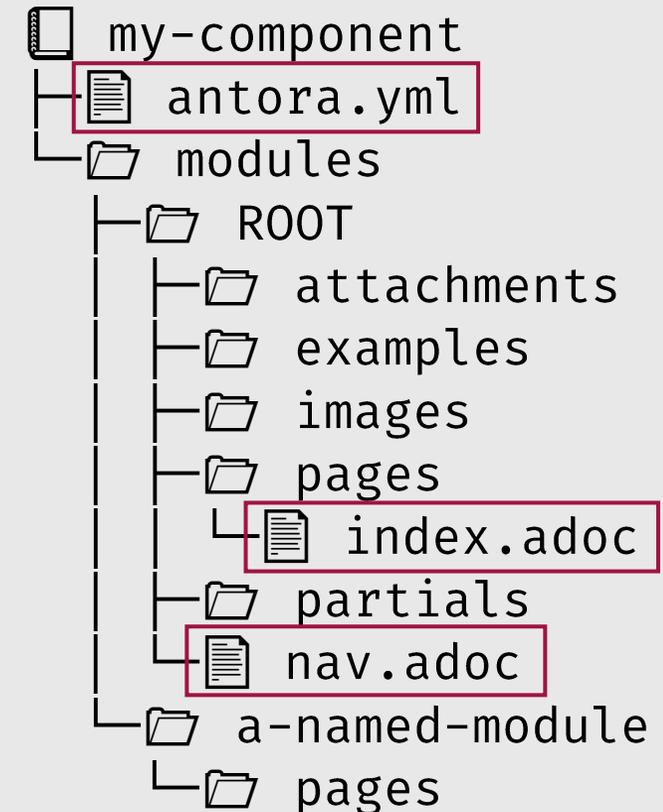
1. Create first component folder structure
2. Create *antora.yml* and give the component a name and version
3. Create *index.adoc* in *pages*
4. Create navigation entries in *nav.adoc* and add it to *antora.yml*

Docu Ops:

1. Install Antora (either via npm CLI or *package.json* file)
2. Create playbook to generate local preview site for authors
3. Create playbook to publish the site on the Internet

Web Developer:

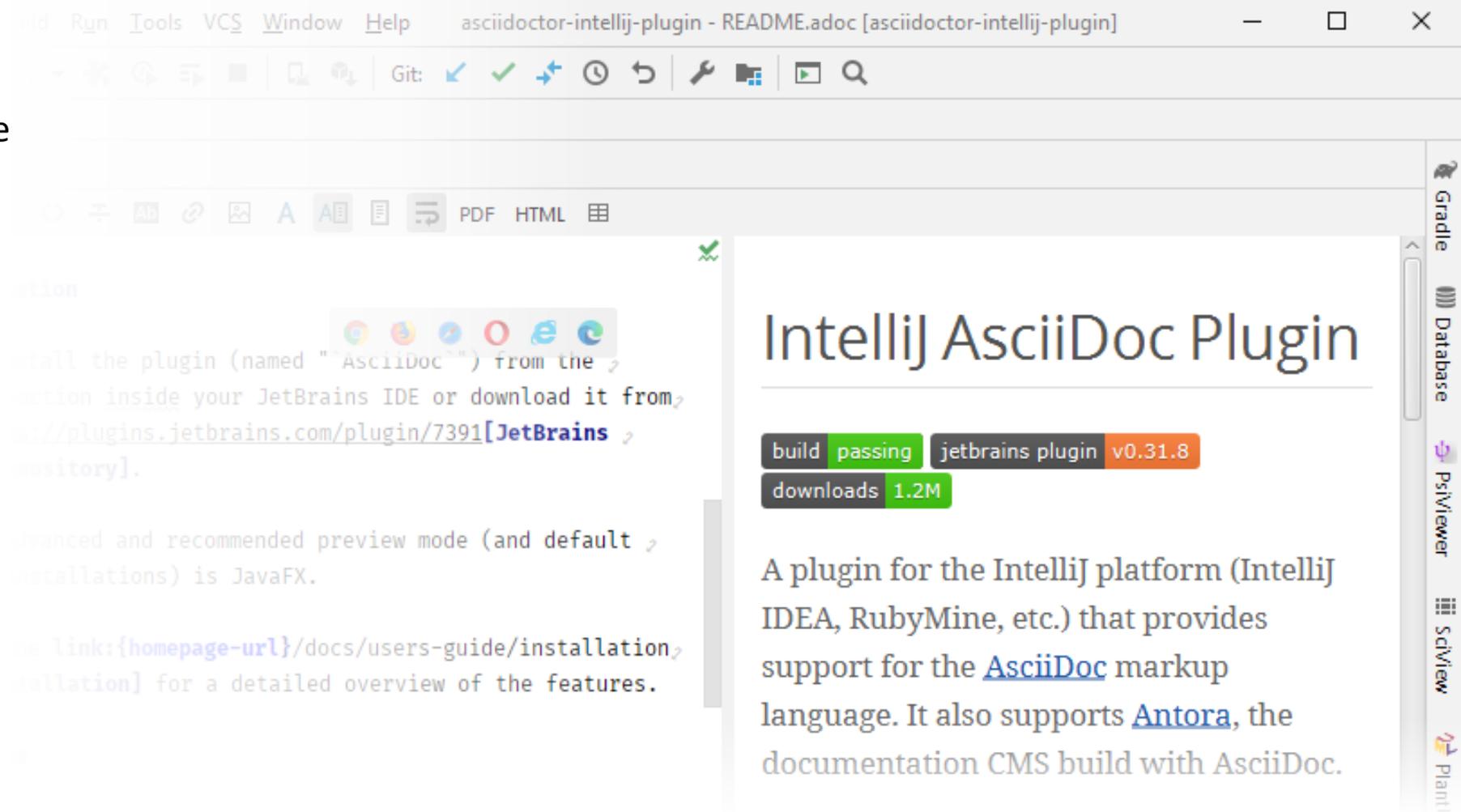
1. Copy the existing UI theme (or create your own)
2. Add UI customizations via supplemental files



Quick Start Guide:

<https://docs.antora.org/antora/2.3/install-and-run-quickstart/>

See how this works using the IntelliJ AsciiDoc plugin!





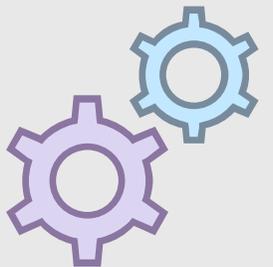
Keep your content up-to-date

using collaboration as a team using version control



Keep your content consistent

by handling component versions and de-duplication of content



Publish new features for software and documentation in sync

using CI/CD tools to automate publishing

Asciidoctor

<https://asciidoctor.org/>

IntelliJ AsciiDoc Plugin

<https://intellij-asciidoc-plugin.ahus1.de/>

PlantUML

<https://plantuml.com/>

<https://real-world-plantuml.com/>

Antora

<https://antora.org/>

Videos, slides and examples

<https://www.ahus1.de/post/asciidoctor-intro-and-deep-dive>

<https://www.ahus1.de/post/cdc-antora-live>

Asciidoctor Kroki extension

<https://github.com/Mogztter/asciidoctor-kroki>



Contact



Alexander Schwartz
Principal IT Consultant

+49 171 5625767
alexander.schwartz@msg.group



msg systems ag
Robert-Bürkle-Straße 1
85737 Ismaning
Germany

value – inspired by people