WYSIWYG and Markdown

This is a massive endeavor and we need external funding to make it happen. Fundamentally, work on this project can start once we have secured enough sponsorship. Please reach out to me if you can contribute coding time or money to hire developers. Two organizations have committed to participate financially so far, but they need re-assurances that their investment will lead to reliable WYSIWYG, which can only happen with more sponsorship. Also, the more funds we have the faster we can through all the list 😁

Thanks!

This is a project to try to attain the following objectives. This is a massive project that will have to be done in phases.

1. A commonly used text syntax like Markdown supported natively in Tiki. Better than PluginMarkdown
   - The CommonMark spec is extensive: https://spec.commonmark.org/
   - We use Markdown for bold, bullets, etc. and we keep our current syntax for Tiki plugins
     {PLUGINNAME()} ... {PLUGINNAME}
   - We do what failed here: http://www.wikicreole.org/wiki/TikiWikiCMSGroupware WikiCreole was an attempt to get a common wiki syntax which is more or less what Markdown is.
   - To have a realistic transition, each text area will have a choice of various syntaxes, so each Tiki instance can move at their own pace. Each text area could have options such as
     - Tiki syntax
     - Markdown + Tiki plugins
     - HTML only
     - HTML + Tiki syntax + Tiki plugins
     - Whatever other mode existed in the past
   - Phased approach while we work on conversion tools
     - Optional, default off
     - Optional, default on
     - In 5-10 years, we can decide if we remove Tiki syntax or keep forever for read-only of legacy content
   - Conversion tools
     - Live (supervised by user) and via command line
     - Need a tool to compare HTML end result

2. A reliable WYSIWYG editor that handles markdown with Tiki plugin syntax and avoids
   https://wysiwygproblems.com/
   - https://medium.com/maatwebsite/lets-see-wyg-with-these-editors-e5720e91a626
   - https://github.com/JefMari/awesome-wysiwyg
   - Along with syntax highlighting for Markdown.
   - Need a way to import HTML (copy-paste), ideally, with some level of conversion to text syntax.
     Ex.: https://github.com/thephpleague/html-to-markdown
   - Need a way to deal with HTML that some Tiki instances may have used
     - PluginWYSIWYG and the WYSIWYG editor in wiki pages (one of the modes is to save as HTML)
A good time for item6191-Replace-Emoticons-with-Emoji

3. Merge Tiki plugins and Tiki modules into Gadgets (or Widgets) that can live in a any text area or elsewhere on a page, and move to Vue JS: https://gitlab.com/jonnybradley/tiki/-/tree/project-fluffy-2
   • Can be done in a second step but overall plan needs to make sure it will be a good solution

4. Realtime collaborative editing
   • Can be done in a second step but overall plan needs to make sure it will be a good solution
   • Could be done with TogetherJS or with another method
   • Matrix:
     https://matrix.org/blog/2021/12/22/the-mega-matrix-holiday-special-2021#applications-beyond-chat

5. Page Builder and Form Builder
   • Can be done in a second step but overall plan needs to make sure it will be a good solution

6. Interop with external systems so we could store some wiki pages in Git, as requested: here
   • Can be done in a second step but overall plan needs to make sure it will be a good solution

Steps

1. Scope high level plan done
2. Compare implementation options as per How to pick a software library (ongoing)
3. Gather community agreement done (there is consensus this is desirable)
4. Gather resources (We have two sponsors so far and we will need more sponsorship)
5. Assemble team (around Jonny and Victor) and do it!

Questions

• How do we deal with just HTML content? (which CKEditor is good at)
  • Discussion from the SummerNote project explaining why it's difficult for an HTML WYSIWYG editor to support Markdown
• Do we keep current PHP lib? or go with another? https://packagist.org/?query=markdown (we need flexibility to add some syntax like wiki plugins and to interact with the WYSIWYG)
• How should we go about converting from Tiki syntax to CommonMark?
  • One option is https://pandoc.org/try/
  • We could envisage a tool to compare Tiki Syntax -> HTML vs Tiki Syntax -> CommonMark -> HTML

Options

1.1.1. ProseMirror with Markdown
https://prosemirror.net/examples/markdown/

1.1.2. TOAST UI Editor

• "The main characteristic of TOAST UI Editor is that it is a markdown based editor that support WYSIWYG editor simultaneously."
For major changes in the latest version:

- Test here: https://nhn.github.io/tui.editor/latest/tutorial-example03-editor-with-wysiwyg-mode
- About major changes in latest version: https://github.com/nhn/tui.editor/releases/tag/v3.0.0
  - Important part: "In order to deal with this issue, we used the Prosemirror, a development tool used to build a WYSIWYG editor, for TOAST UI Editor 3.0, and we were able to unify the internal dependencies of the editors into one. The unified dependency allowed us to follow a single, internal structure, and we were able to remove the past dependencies including CodeMirror, squire, and to-mark."

- MIT License
- Based on ProseMirror
- Has an image editor
- Available here: https://asset-packagist.org/package/bower-asset/tui-editor
- This model is the most reliable way to have both a text syntax and WYSIWYG, but for legacy HTML content, we'd have to continue with CKEditor

1.1.3. MeditorJS
- https://github.com/cobalt-os-team/MeditorJS

1.1.4. Stacks-Editor
- https://github.com/StackExchange/Stacks-Editor
- Based on ProseMirror

1.1.5. Milkdown
- https://milkdown.dev/
- https://github.com/Saul-Mirone/milkdown
- Based on ProseMirror

1.1.6. TipTap
- https://github.com/ueberdosis/tiptap
- Based on ProseMirror

Ideas

- Idea from Jonny: use attributes so each object can have a different setting. So old comments could be Tiki syntax, and new comments could be Markdown.

Related links

- https://discuss.prosemirror.net/t/offline-peer-to-peer-collaborative-editing-using-yjs/2488
- https://blog.kevinjahns.de/are-crdts-suitable-for-shared-editing/
- https://www.tiny.cloud/blog/real-time-collaboration-ot-vs-crdt/
- People want Markdown: https://forum.virtualmin.com/t/links-to-documentation-are-broke/111602