Geogebra Plugin

I created a plugin to embed Geogebra files since the current Geogebra plugins are all too old and don't work. See here for general Geogebra embedding instructions.

This is my first plugin and a bit of a cheat: I basically took the Color plugin and modified it until it did what I needed. However, there's still a lot of code that I don't really understand and a few features that are missing.

Here's how I installed it:

- Add the following three files in the wiki folder: `./lib/plugins/ggb/`
- `manager.dat`

```
installed=Sun, 08 Nov 2020 08:20:00 -0800
```

- `plugin.info.txt`

```
base    ggb
author  Patrick Truchon
email   patoo@rbox.me
date    2020-11-08
name    geogebra6 syntax plugin
desc    Include GeoGebra6 files into Dokuwiki
```

- `syntax.php`

```php
<?php
/**
 * Plugin ggb: Embeds Geogebra files into Dokuwiki.
 *
 * @license     GPL 3 (http://www.gnu.org/licenses/gpl.html)
 * @author      Patrick Truchon <patoo@rbox.me>
 * See: https://wiki.geogebra.org/en/Reference:GeoGebra_Apps_Embedding
 */

// must be run within DokuWiki
if(!defined('DOKU_INC')) die();
if(!defined('DOKU_PLUGIN')) define('DOKU_PLUGIN',DOKU_INC.'lib/plugins/');
require_once(DOKU_PLUGIN.'syntax.php');


/**
 * All DokuWiki plugins to extend the parser/rendering mechanism
 * need to inherit from this class
 */
class syntax_plugin_ggb extends DokuWiki_Syntax_Plugin {

    function getType(){ return 'substition'; }
```

```php
    function getSort(){ return 306; }
    function connectTo($mode) {
$this->Lexer->addSpecialPattern('{{ggb>.*?}}',$mode,'plugin_ggb'); }


    /**
     * Handle the match
     */
    function handle($match, $state, $pos, Doku_Handler $handler){
        $match = html_entity_decode(substr($match, 6, -2));
        list($ggbfile, $index) = explode('|',$match,2);
        if (preg_match('/(.*) ([0-9]+,[0-9]+)$/',trim($ggbfile),$matches)) {
            $ggbfile = $matches[1];
        if (strpos($matches[2],',') !== false) {
            list($w, $h) = explode(',',$matches[2],2);
            }
        else {
            $w = '800';
            $h = $matches[2];
            }
            }
        else {
            $w = '800';
            $h = '400';
            }

        if (!isset($index)) $index = '';
        return array($ggbfile, hsc(trim($index)), hsc(trim($w)), hsc(trim($h)));
        }

    /**
     * Create output
     */
    function render($mode, Doku_Renderer $renderer, $data) {
        list($ggbfile, $index, $w, $h) = $data;
        if($mode == 'xhtml'){
            $slash = strrpos($ggbfile, "/");
            if($slash > 0){
                $slash = $slash + 1;
                }
            $ggbid = substr($ggbfile, $slash, -4);
            $renderer->doc .= '<script
src="https://www.geogebra.org/apps/deployggb.js"></script>';
            $renderer->doc .= '<div id="'.$ggbid.'"></div>';
            $renderer->doc .= '<script>';
            $renderer->doc .= 'window.addEventListener("load", function() {';
            $renderer->doc .= 'new GGBApplet({"width": '.$w.', "height": '.$h.',
';
            $renderer->doc .= '"showToolBar": false, "showAlgebraInput": false,';
            $renderer->doc .= '"showMenuBar": false, "filename": "/_media/';
            $renderer->doc .= "$ggbfile\"},true).inject($ggbid) })";
            $renderer->doc .= '</script>';
```

```php
            $renderer->doc .= 'Download <a
    href="/_media/'.$ggbfile.'">'.$ggbid.'.ggb</a>';
            return true;
        }
        return false;
    }
}
?>
```

The syntax is:

```
{{ggb>path/to/geogebrafile.ggb 740,300}}
```

Note that at the moment, if a path is used, it must use / instead of :. The default dimensions are 800×400 if they are omitted.

Here's an example from one of Ham Basics pages showing how different waves add and what SWR looks like:

Wave Addition

When two waves overlap, they add up together at every point. Here, the blue and green waves are generated and add up together to form the red wave. You can move the blue and green waves and see the result. To convince yourself that the red wave is really the sum of the blue and green waves, look at points A, B, and C. You can move the blue or green waves by sliding their phase (φ and Φ) around. You'll see that point C is always the sum of A and B.

Download waveaddition.ggb

Where do the blue and green waves need to be so that...

- the red wave is the biggest?
- the red wave is cancelled out?[1]

If you press the play button on the bottom left corner, you'll see the blue wave travel to the right and the green wave travel to the left. The red wave oscillates up and down but doesn't travel anywhere. This is called a *standing wave* (SWR). While the animation is running, slowly decrease the amplitude of the reflected wave (V_B) and you'll see that the red wave (which is what actually makes it to the antenna), starts moving to the right.

[1]
Fun fact: This is how 🔊 noise cancelling headphones work. The headset has a microphone that picks up the noise, inverts the waves, and plays them back in the ear piece. The combination of the real life noise and the inverted noise being played in the speaker cancel out (somewhat).