Zend_Ascii - Ben Scholzen

<ac:macro ac:name="unmigrated-inline-wiki-markup"><ac:plain-text-body><![CDATA[

<ac:macro ac:name="unmigrated-inline-wiki-markup"><ac:plain-text-body><![CDATA[

Zend Framework: Zend_Ascii Component Proposal

Proposed Component Name	Zend_Ascii
Developer Notes	http://framework.zend.com/wiki/display/ZFDEV/Zend_Ascii
Proposers	Ben Scholzen Matthew Weier O'Phinney (Zend Liaison)
Revision	1.0 - 19 January 2007: Initial proposal (wiki revision: 8)

Table of Contents

1. Overview
2. References
3. Component Requirements, Constraints, and Acceptance Criteria
4. Dependencies on Other Framework Components
5. Theory of Operation
6. Milestones / Tasks
7. Class Index
8. Use Cases
9. Class Skeletons

1. Overview

Zend_Ascii is a component to help a programmer creating elements out of characters. The current class only supports tables yet. You can define colspans, border styles and aligns for each column.

2. References

3. Component Requirements, Constraints, and Acceptance Criteria

- This component **will** allow to define different border styles per table column and row
- This component **will** allow setting the align in each column
- This component **will** allow creating dropshadows to elements
- This component **will only** work with UTF-8 encoded output

4. Dependencies on Other Framework Components

- Zend_Exception

5. Theory of Operation

Zend_Ascii will be instantiated and Zend_Ascii_Elements can then be appended to it. Those elements can be Tables for example. After all elements are appended to Zend_Ascii, the output string can be rendered.

6. Milestones / Tasks

- Milestone 1: [DONE] Creating base class
- Milestone 2: [DONE] Creating table support
- Milestone 3: [DONE] Checking current state into http://zend_ascii.svn.dasprids.de/
- Milestone 4: Proposal acceptance and import of current prototype to incubator SVN
- Milestone 5: Unit tests exist and development.
- Milestone 6: Documentation

7. Class Index

- Zend_Ascii
- Zend_Ascii_Exception
- Zend_Ascii_Element_Interface
- Zend_Ascii_Element_Table
- Zend_Ascii_Element_Table_Column
- Zend_Ascii_Element_Table_Row

8. Use Cases

UC-01

Creating a simple table

```
$ascii = new Zend_Ascii();

$table = new Zend_Ascii_Element_Table(array(20,
                                            array(30, Zend_Ascii::BORDER_THICK),
                                            30),
                                      Zend_Ascii::BORDER_THIN);

$row = Zend_Ascii_Element_Table::createRow();
$row->addColumn(Zend_Ascii_Element_Table::createColumn('Let us write a lot of fisne
text in this column'));
$row->addColumn(Zend_Ascii_Element_Table::createColumn('Let us write a lot of fine
text in this column', Zend_Ascii_Element_Table_Column::ALIGN_CENTER));
$row->addColumn(Zend_Ascii_Element_Table::createColumn('Let us write a lot of fine
text in this column', Zend_Ascii_Element_Table_Column::ALIGN_RIGHT));
$table->addRow($row);

$row = Zend_Ascii_Element_Table::createRow(Zend_Ascii::BORDER_THIN);
$row->addColumn(Zend_Ascii_Element_Table::createColumn('Let us write a lot of fine
text in this column', null, 2));
$row->addColumn(Zend_Ascii_Element_Table::createColumn('Let us write a lot of fine
text in this column'));
$table->addRow($row);

$row = Zend_Ascii_Element_Table::createRow(Zend_Ascii::BORDER_THICK);
$row->addColumn(Zend_Ascii_Element_Table::createColumn('Let us write a lot of fine
text in this column', null, 2));
$row->addColumn(Zend_Ascii_Element_Table::createColumn('Let us write a lot of fine
text in this column'));
$table->addRow($row);

$row = $table->createRow();
$row->addColumn(Zend_Ascii_Element_Table::createColumn('Let us write a lot of fine
text in this column', Zend_Ascii_Element_Table_Column::ALIGN_CENTER, 3));
$table->addRow($row);

$ascii->append($table);
```

9. Class Skeletons

```
class Zend_Ascii
{
    /**
     * Zend_Ascii is a helper class to create elements in simple ascii styles.
     *
     * @param string $defaultText
     */
    public function __construct($defaultText = '')
    {}

    /**
```

```
     * Append an element to the ascii object
     *
     * @param Zend_Ascii_Element $element
     */
    public function append(Zend_Ascii_Element_Interface $element)
    {}

    /**
     * Get the text of this ascii element
     *
     * @return string
     */
    public function render()
    {}

    /**
     * Convert am unicode character code to a character
     *
     * @param int $code
     * @return string
     */
    public static function unichr($code)
    {}

    /**
     * Get a character from the pre-defined list
     *
     * @return string
     */
    public static function getBorderChar($border, $style, $crossStyle = null)
    {}
}

class Zend_Ascii_Exception extends Zend_Exception
{}

interface Zend_Ascii_Element_Interface
{
    /**
     * Render the element to a string
     *
     * @return string
     */
    public function render();
}

class Zend_Ascii_Element_Table implements Zend_Ascii_Element_Interface
{
    /**
     * Init the table element.
     *
     * $columns is an array, which contains all column widths.
     *
     * @param array $columns
     * @param int $borderStyle
     */
    public function __construct(array $columns, $borderStyle = null)
    {}
```

```
    /**
     * Add a row the the talbe
     *
     * @param Zend_Ascii_Element_Table_Row $row
     */
    public function addRow(Zend_Ascii_Element_Table_Row $row)
    {}

    /**
     * Create a column
     *
     * @param string $content
     * @param int $align
     * @param int $colspan
     */
    static public function createColumn($content = null, $align = null, $colspan =
null)
    {}

    /**
     * Create a row
     *
     * @param int $bottomBorderStyle
     */
    static public function createRow($bottomBorderStyle = null)
    {}

    /**
     * Render the element to a string
     *
     * @return string
     */
    public function render()
    {}
}

class Zend_Ascii_Element_Table_Column
{
    /**
     * Set the content
     *
     * @param string $content
     */
    public function setContent($content)
    {}

    /**
     * Set the align
     *
     * @param int $align
     */
    public function setAlign($align)
    {}

    /**
     * Set the colspan
     *
     * @param int $colspan
     */
```

```
    public function setColspan($colspan)
    {}

    /**
     * Get the colspan
     *
     * @param int $colspan
     */
    public function getColspan()
    {}

    /**
     * Render this cell
     *
     * @param int $colWidth
     * @return string
     */
    public function render($colWidth)
    {}
}

class Zend_Ascii_Element_Table_Row
{
    /**
     * Add a column to the row
     *
     * @param Zend_Ascii_Element_Table_Column $column
     */
    public function addColumn(Zend_Ascii_Element_Table_Column $column)
    {}

    /**
     * Set the style of the bottom border.
     *
     * This will be ignored, if this is the last row of the table
     *
     * @param int $style
     */
    public function setBottomBorderStyle($borderStyle)
    {}

    /**
     * Get the style of the bottom border.
     *
     * @return int
     */
    public function getBottomBoderStyle()
    {}

    /**
     * Render the row
     *
     * @param array $columnStyles
     * @return string
     */
    public function render(array $columnStyles)
    {}

    /**
```

```
     * Get all columns
     *
     * @param array $columnStyles
     * @return array
     */
    public function getColumns($columnStyles)
```

```
        {}
  }
```

]]></ac:plain-text-body></ac:macro>
]]></ac:plain-text-body></ac:macro>