

```

/***

EditableBulletList

(optional) values : list

(optional) noValuesLabel : str

(optional) editLabel : str

(optional) addLabel : str

(optional) saveLabel : str

(optional) cancelLabel : str

(optional) publish : str

(optional) class : str

(optional) readOnly : bool

(optional) readOnlyAlert : str

(optional) message : map

***

// TODO (steveb):
// 1) reset the contents of the textboxes when the edit operation is
canceled
// 2) update the bullet list after save is confirmed

// read parameters
var values = $values ?? [ ];
var novalueslabel = $novalueslabel ?? 'There are no values.';
var editlabel = $editlabel ?? 'edit';
var addlabel = $addlabel ?? 'add';
var savelabel = $savelabel ?? 'save';
var cancellabel = $cancellabel ?? 'cancel';
var channel = $publish ?? 'default';
var class = $class;
var readonly = $readonly ?? user.anonymous;
var readonlyalert = $readonlyalert ?? 'You must login to edit these
values.';
var message = $message ?? { };

// render controls

```

```

<div id=(@div) class=(class)>

  // check if the edit button should be enabled
  if(readonly) {
    <input type="button" value=(editlabel) id=(@edit) ctor="
      when($this.click) alert({{ readonlyalert }});
    ">
  } else {
    <input type="button" value=(editlabel) id=(@edit) ctor="
      when($this.click) @toggle();
    ">
  }

  // render current values as a bulleted list
  <div id=(@displayspan)>
    if(#values == 0) {
      <div id=(@resizeonedit)>
        novalueslabel
      </div>
    } else {
      <ul>
        foreach(var value in values) {
          <li> value </li>
        }
      </ul>
    }
  </div>

  // render the edit form
  <form id=(@changeform)>
    if(#values != 0) {
      foreach(var value in values) {
        <textarea>
          value
        </textarea>
      }
    } else {
      <textarea />
    }

    // separate the input boxes from the controls
    <div class="cleardiv"> &nbsp; </div>

    // render the add button
    <input type="button" id=(@add) value=(addlabel) ctor="
      when($this.click) $(''#' + {{ @changeform }} + '
textarea:last').after('<textarea></textarea>');
    ">

    // render the save button

```

```

&nbsp;;
<input type="button" id=(@save) value=(savelabel) ctor="
  when($this.click) {
    var newvalues = [ ];
    $('#'+ {{ @changeform }} + ' textarea').each(function() {
      var val = $(this).val();
      if(val != '') newvalues.push(val);
    });
    var msg = {{ message }};
    msg.values = newvalues;
    msg.completed = function() { @toggle(); };
    Deki.publish({{ channel }}, msg);
  }
"/>

// render the cancel button
&nbsp;;
<input type="button" id=(@cancel) value=(cancellabel) ctor="
  when($this.click) @toggle();
"/>
</form>
</div>

// script
<script type="text/jem">
when(@toggle) {
  $('#'+ {{ @changeform }} + ',#' + {{ @displayspan }} + ',#' + {{
@edit }}).slideToggle();
  var resizeonedit = $('#'+ {{ @resizeonedit }});
  if(resizeonedit.css('padding-top') == '0px') {
    resizeonedit.css('padding-top', '1.6em');
  } else {
    resizeonedit.css('padding-top', '0px');
  }
}
"</script>

// styles
<style type="text/css">
#" .. @resizeonedit .. " {
  padding-top: 1.6em;
}
#" .. @changeform .. " {
  display: none;
}
#" .. @changeform .. " textarea {
  width: 50%;
  float: left;
  margin-bottom: 5px;
}

```

```
#" .. @div .. " .cleardiv {  
  height:1px;  
  clear:both;  
}  
"</style>
```