



# Template:Gallery

[Bilder laden]

```
(function() {

    /* Letzten uninitialisierten Wrapper finden */
    var allWraps = document.getElementsByClassName('galleryWrap');
    var wrap = null;
    var i;
    for (i = 0; i < allWraps.length; i++) {
        if (!allWraps[i].getAttribute('data-ginit')) { wrap = allWraps[i]
        }
    }
    if (!wrap) { return; }
    wrap.setAttribute('data-ginit', '1');

    function q(cls) {
        var els = wrap.getElementsByClassName(cls);
        return (els && els.length) ? els[0] : null;
    }

    var loadBtn    = q('galleryLoadBtn');
    var prevBtn    = q('galleryPrevBtn');
    var nextBtn    = q('galleryNextBtn');
    var playBtn    = q('galleryPlayBtn');
    var stopBtn    = q('galleryStopBtn');
    var counter    = q('galleryCounter');
    var statusEl   = q('galleryStatus');
    var stage      = q('galleryStage');
    var mainImg    = q('galleryMainImg');
    var imgName    = q('galleryImgName');
    var thumbRow   = q('galleryThumbRow');
    var imgDesc    = q('galleryImgDesc');

    if (!loadBtn || !stage || !mainImg) { return; }

    var images     = new Array();
    var currentIdx = 0;
    var slideTimer = null;
```

```

var DELAY      = 3000;
var MAX_DEPTH  = 5;
var IMG_EXTS   = 'jpg jpeg png gif bmp webp svg';
var AUTO_LOAD  = true; /* true = Bilder beim Seitenaufruf automatisch

/* BFS-Zustand */
var bfsQueue   = new Array();
var bfsIdx     = 0;
var allPageIds = new Array();
var pageUrlMap = new Object(); /* pageId -> Wiki-UI-URL */

function setText(el, txt) { if (el) { el.textContent = txt; } }
function setStatus(msg, color) { setText(statusEl, msg); if (statusEl)

function getPageId() {
    var pid = 0;
    if (typeof Deki !== 'undefined') {
        if (Deki.PageId) { pid = Deki.PageId; }
        else if (Deki.Page && Deki.Page.Id) { pid = Deki.Page.Id; }
    }
    return pid;
}

function forceHttps(url) {
    if (url && url.indexOf('http://') === 0) {
        return 'https://' + url.substring(7);
    }
    return url;
}

/* -----
PHASE 1: BFS - alle Unterseiten bis MAX_DEPTH sammeln
(sequenziell, eine Anfrage nach der anderen)
----- */
function loadImages() {
    var pageId = getPageId();
    if (!pageId) { setStatus('Page-ID nicht gefunden!', '#ff4444'); }
    loadBtn.disabled = true;
    images           = new Array();
    allPageIds       = new Array();
    bfsQueue         = new Array();
    bfsIdx           = 0;
    pageUrlMap       = new Object();
    pageUrlMap[pageId] = window.location.href.split('?')[0];

```

```

    allPageIds.push(pageId);
    bfsQueue.push(new Object());
    bfsQueue[0].id    = pageId;
    bfsQueue[0].depth = 0;

    setStatus('Sammle Unterseiten (Tiefe 0/' + MAX_DEPTH + ')...', '#
    processBFS();
}

function processBFS() {
    /* Alle Queue-Eintraege abgearbeitet → Bilder laden */
    if (bfsIdx >= bfsQueue.length) {
        setStatus('Lade Bilder von ' + allPageIds.length + ' Seite(n)
        fetchAllFiles();
        return;
    }

    var item = bfsQueue[bfsIdx];
    bfsIdx++;

    /* Maximale Tiefe erreicht → keine weiteren Kinder holen */
    if (item.depth >= MAX_DEPTH) {
        processBFS();
        return;
    }

    var xhr = new XMLHttpRequest();
    xhr.open('GET', '/@api/deki/pages/' + item.id + '/subpages?dream
    xhr.onload = function() {
        if (xhr.status === 200) {
            var parser = new DOMParser();
            var xml     = parser.parseFromString(xhr.responseText, 'te
            var nodes  = xml.getElementsByTagName('page.subpage');
            var j;
            for (j = 0; j < nodes.length; j++) {
                var childId = nodes[j].getAttribute('id');
                if (childId) {
                    allPageIds.push(childId);
                    var uiNodes = nodes[j].getElementsByTagName('uri
                    pageUrlMap[childId] = (uiNodes.length) ? forceHtt
                    var child    = new Object();
                    child.id     = childId;
                    child.depth  = item.depth + 1;
                    bfsQueue.push(child);
                }
            }
        }
    }
}

```

```

    }
    setStatus(
        'Unterseiten: ' + (allPageIds.length - 1) +
        ' gefunden (Tiefe ' + (item.depth + 1) + '/' + MAX_DEPTH +
        '#7ecfff'
    );
}
processBFS();
};
xhr.onerror = function() { processBFS(); };
xhr.send();
}

/* -----
   PHASE 2: Dateien aller gesammelten Seiten parallel laden
   ----- */
function fetchAllFiles() {
    var total      = allPageIds.length;
    var remaining = total;
    if (total === 0) { finalize(); return; }

    var j;
    for (j = 0; j < total; j++) {
        (function(pid) {
            var xhr = new XMLHttpRequest();
            xhr.open('GET', '/@api/deki/pages/' + pid + '/files?dream
            xhr.onload = function() {
                if (xhr.status === 200) { parseAndAddFiles(xhr.responseText, pid);
                remaining--;
                setStatus(
                    'Seiten: ' + (total - remaining) + '/' + total +
                    '#7ecfff'
                );
                if (remaining === 0) { finalize(); }
            };
            xhr.onerror = function() {
                remaining--;
                if (remaining === 0) { finalize(); }
            };
            xhr.send();
        })(allPageIds[j]);
    }
}

function parseAndAddFiles(responseText, pid) {

```

```

var parser = new DOMParser();
var xml    = parser.parseFromString(responseText, 'text/xml');
var nodes  = xml.getElementsByTagName('file');
var j;
for (j = 0; j < nodes.length; j++) {
    var cNodes = nodes[j].getElementsByTagName('contents');
    var nNodes = nodes[j].getElementsByTagName('filename');
    var dNodes = nodes[j].getElementsByTagName('description');
    if (!cNodes.length) { continue; }
    var url    = forceHttps(cNodes[0].getAttribute('href') || '');
    var name   = nNodes.length ? nNodes[0].textContent : url;
    var ext    = name.split('.').pop().toLowerCase();
    if (IMG_EXTS.indexOf(ext) === -1) { continue; }
    /* Duplikate per URL verhindern */
    var k;
    var dup = false;
    for (k = 0; k < images.length; k++) {
        if (images[k].url === url) { dup = true; break; }
    }
    if (dup) { continue; }
    var desc = (dNodes.length && dNodes[0].textContent) ? dNodes[0].textContent : '';
    var item = new Object();
    item.url    = url;
    item.name   = name;
    item.desc   = desc;
    item.pageUrl = (pageUrlMap[pid]) ? pageUrlMap[pid] : '';
    images.push(item);
}
}

/* -----
   PHASE 3: Gallery anzeigen
   ----- */
function finalize() {
    loadBtn.disabled = false;
    if (images.length === 0) { setStatus('Keine Bilder gefunden.', '#f00');
    setStatus('');
    loadBtn.style.display = 'none';
    stage.style.display  = 'block';
    if (thumbRow) { thumbRow.style.display = 'flex'; }
    if (prevBtn)  { prevBtn.style.display = 'inline-block'; }
    if (nextBtn)  { nextBtn.style.display = 'inline-block'; }
    if (playBtn)  { playBtn.style.display = 'inline-block'; }
    buildThumbs();
    showImage(0);
}

```

```

}

function showImage(idx) {
    currentIdx = idx;
    if (mainImg) {
        mainImg.src = ''; mainImg.src = images[idx].url;
        mainImg.style.cursor = 'pointer';
        mainImg.onclick = function() { window.open(images[idx].pageURL); }
    }
    setText(imgName, images[idx].name);
    setText(imgDesc, images[idx].desc ? images[idx].desc : 'Keine Beschreibung');
    setText(counter, (idx + 1) + ' / ' + images.length);
    if (thumbRow) {
        var thumbs = thumbRow.getElementsByTagName('img');
        var j;
        for (j = 0; j < thumbs.length; j++) {
            thumbs[j].className = j === idx ? 'gThumb gThumbActive' : 'gThumb';
        }
    }
}

function buildThumbs() {
    if (!thumbRow) { return; }
    thumbRow.innerHTML = '';
    var j;
    for (j = 0; j < images.length; j++) {
        var t = document.createElement('img');
        t.className = 'gThumb';
        t.src = images[j].url;
        t.alt = images[j].name;
        t.title = images[j].name;
        (function(idx) { t.onclick = function() { showImage(idx); }; })(j);
        thumbRow.appendChild(t);
    }
}

function showNext() { showImage((currentIdx + 1) % images.length); }
function showPrev() { showImage((currentIdx - 1 + images.length) % images.length); }

function startSlideshow() {
    if (playBtn) { playBtn.style.display = 'none'; }
    if (stopBtn) { stopBtn.style.display = 'inline-block'; }
    slideTimer = setInterval(function() { showNext(); }, DELAY);
}

```

```

function stopSlideshow() {
    if (slideTimer) { clearInterval(slideTimer); slideTimer = null; }
    if (stopBtn) { stopBtn.style.display = 'none'; }
    if (playBtn) { playBtn.style.display = 'inline-block'; }
}

if (loadBtn) { loadBtn.onclick = function() { loadImages(); }; }
if (prevBtn) { prevBtn.onclick = function() { showPrev(); }; }
if (nextBtn) { nextBtn.onclick = function() { showNext(); }; }
if (playBtn) { playBtn.onclick = function() { startSlideshow(); }; }
if (stopBtn) { stopBtn.onclick = function() { stopSlideshow(); }; }

if (AUTO_LOAD) { loadImages(); }

document.onkeydown = function(e) {
    if (!stage || stage.style.display === 'none') { return; }
    var k = e.keyCode;
    if (k === 37) { showPrev(); }
    else if (k === 39) { showNext(); }
    else if (k === 32) { if (slideTimer) { stopSlideshow(); } else {
};

})();

NoElements {}

```